THE UNIVERSITY *of* EDINBURGH

This thesis has been submitted in fulfilment of the requirements for a postgraduate degree (e. g. PhD, MPhil, DClinPsychol) at the University of Edinburgh. Please note the following terms and conditions of use:

- This work is protected by copyright and other intellectual property rights, which are retained by the thesis author, unless otherwise stated.
- A copy can be downloaded for personal non-commercial research or study, without prior permission or charge.
- This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the author.
- The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author.
- When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given.

An optimal approach to the design of experiments for the automatic characterisation of biosystems

David Gomez-Cabeza

THE UNIVERSITY
of EDINBURGH

Doctor of Philosophy

The University of Edinburgh

2023

Declaration

I declare that this thesis was composed solely by myself, that the work contained herein is my own except where explicitly stated otherwise in the text by reference or acknowledgement, and that this work has not been submitted for any other degree or professional qualification.

David Gomez-Cabeza
June 2023

Abstract

The Design-Build-Test-Learn cycle is the main approach of synthetic biology to re-design and create new biological parts and systems, targeting the solution for complex and challenging paramount problems. The applications of the novel designs range from biosensing and bioremediation of water pollutants (e.g. heavy metals) to drug discovery and delivery (e.g. cancer treatment) or biofuel production (e.g. butanol and ethanol), amongst others. Standardisation, predictability and automation are crucial elements for synthetic biology to efficiently attain these objectives. Mathematical modelling is a powerful tool that allows us to understand, predict, and control these systems, as shown in many other disciplines such as particle physics, chemical engineering, epidemiology and economics. Yet, the inherent difficulties of using mathematical models substantially slowed their adoption by the synthetic biology community. Researchers might develop different competing model alternatives in absence of in-depth knowledge of a system, consequently being left with the burden of with having to find the best one. Models also come with unknown and difficult to measure parameters that need to be inferred from experimental data. Moreover, the varying informative content of different experiments hampers the solution of these model selection and parameter identification problems, adding to the scarcity and noisiness of laborious-to-obtain data. The difficulty to solve these non-linear optimisation problems limited the widespread use of advantageous mathematical models in synthetic biology, broadening the gap between computational and experimental scientists. In this work, I present the solutions to the problems of parameter identification, model selection and experimental design, validating them with *in vivo* data. First, I use Bayesian inference to estimate model parameters, relaxing the traditional noise assumptions associated with this problem. I also apply information-theoretic approaches to evaluate the amount of information extracted from experiments (entropy gain). Next, I define methodologies to quantify the informative content of tentative experiments planned for model selection (distance between predictions of competing models) and parameter inference (model prediction uncertainty). Then,

I use the two methods to define efficient platforms for optimal experimental design and use a synthetic gene circuit (the genetic toggle switch) to substantiate the results, computationally and experimentally. I also expand strategies to optimally design experiments for parameter identification to update parameter information and input designs during the execution of these (on-line optimal experimental design) using microfluidics. Finally, I developed an open-source and easy-to-use Julia package, BOMBs.jl, automating all the above functionalities to facilitate their dissemination and use amongst the synthetic biology community.

Lay Summary

Life is a fascinating but complicated phenomenon. We are composed of many small living units known as cells. So are other organisms we can see with the naked eye, such as animals or plants. Yet, some cells can live on their own without forming large organisms. Some examples are the yeast we use to make bread or the bacteria that cause us intestinal infections. These cells interact and process information from their surroundings to survive, grow and reproduce. To do so, they perform a wide variety of functions within them. A big part of these functions involves a crucial group of bio-molecules, the proteins. These molecules can translate information from the environment into something the cells understand. They can also make some biochemical reactions faster or even detect dangerous substances. DNA is another important bio-molecule that contains the genetic information of a cell. This "instruction manual" holds all the information about an organism saying which proteins a cell can produce or how this production is regulated. For decades scientists have been able to take fractions of DNA from a cell and introduce it into different ones. An early example of this genetic editing is the introduction of the gene of a glowing protein from a jellyfish into many other organisms to better study cellular functions. Such is the advancement in this genetic editing process that now scientists can create new cellular functions never found in nature before and introduce them into cells. This discipline is known as synthetic biology, and it is eager to solve complicated problems like efficient bio-fuel production, cleaning polluted waters or curing some types of cancer, amongst many others. Yet, despite many efforts, it is usually quite hard to predict the exact behaviour of such new functions or the potential side effects they will cause to the cells beforehand. This problem limits the speed at which the knowledge of the field advances and its applicability to solve real-life problems.

Mathematical modelling is a powerful tool that we have seen help advance other sciences, such as physics. We can use mathematical equations to describe cell

functions. These models help scientists understand how cells behave in certain conditions. They can predict the behaviour of new functions that we introduce to them. Yet, using these mathematical models still presents challenges that limit their spread in biology. The first problem might come if a scientist is unsure of the biochemical mechanisms behind a specific function. If that is the case, one might come up with different candidate models. At this point, the issue is, how do we choose the correct or at least the best one? This is the problem of model selection. An even more common problem is that when one formulates the equations to describe a cell function, these tend to have a series of unknown variables (numbers) known as parameters. Finding the exact value for these parameters is complicated and sometimes even impractical. However, we can indirectly identify them from experimental data. But how do we perform this indirect identification of those numbers? How confident can we be from the estimations? This is the problem of parameter identification. Ultimately, we need to interrogate cells and collect experimental data to solve both problems. It turns out that some experiments are better than others in solving these two issues. However, these experiments or cell interrogations tend to be highly laborious, time-consuming and expensive. So, what would be the best way to determine how useful an experiment is before performing it to obtain much better results with less cost and effort? This is the problem of experimental design.

In this thesis, I seek to answer the three previous questions to aid the spread of mathematical modelling across biology. For the problem of model selection, I will present a strategy to better design experiments to efficiently solve it. I try to find experimental designs (a specific combination of experimental variables) that make two competing models show behaviours as different as possible for the same cellular function. These experiments will better help uncover the best model and discard the one incapable of predicting the real cellular operation. Since the number of possible experimental options can be unreasonably high, I use optimisation programs to find the best one. We call this procedure optimal experimental design for model selection. To indirectly identify the value of the parameters (known as inference), I will investigate the two main statistical approaches to solve the task (frequentist and Bayesian). Such statistical methods try to find a value for the parameters by comparing the predictions of our mathematical models and the experimental data we collect. The inferred parameters will be the ones that make this difference as little as possible. Yet, this is not a trivial task. Usually, we can identify such parameters only with a

certain confidence. So, instead of obtaining an exact number, we get a range of values with a probability associated. One can select the value with the highest probability and the amount of error linked to that estimate (a common approach in frequentist statistics). On the other hand, one can consider the whole range of values with their probabilities and try to minimise their uncertainty as much as possible in light of new data (a common approach in Bayesian statistics). I will also show a potential strategy to better design experiments using a Bayesian approach for the task in what is called optimal experimental design for model inference. The strategy uses again an optimisation program to select the experiment that makes our models highly unsure about what they have to predict. I will show that this makes the experimental data more informative, easing the parameter inference problem. After showing this strategy computationally, I will also apply it in a real scenario, using bacteria with a synthetic function inside them and a mathematical model describing this. Since we are talking about something extremely tiny, proteins inside bacteria cells, I will use a microscope to observe them. The cells also contain glowing proteins glued to our proteins of interest to indirectly quantify their amount by measuring the amount of light emitted by these tags. To have even better control of what we provide to the cells (this is their "food" or other chemicals), I will trap the cells inside small chambers connected to small channels (everything in the order of a few micrometres) in what is known as microfluidic devices. Next, I will expand our experimental set-up to process the experimental data in real time and automatically design optimal experiments. After showing the advantages of optimally designing experiments for model selection and parameter inference, I wrapped up the majority of the strategies and algorithms presented into a free and easy-to-use program called BOMBs.jl. This program requires minimal coding knowledge, and I wrote it in an efficient programming language of recent increasing popularity called Julia. Hence, together with this thesis, BOMBs.jl aims to ease the spread of all these methodologies and the use of mathematical models across the different fields of Biology.

Acknowledgments

Firstly, I would like to express my deepest gratitude to my supervisor Dr Filippo Menolascina for his invaluable words of wisdom, support and guidance. Thanks for your time and enthusiasm, for improving my critical thinking and most importantly, for having given me the opportunity to grow as a scientist in your group. I will always admire your creativity and curiosity. I am also deeply indebted to Dr Lucia Bandiera, with whom none of this would have been possible. Thanks for your friendship, mentorship, patience and trust. Also, thanks for all the time you have spent teaching me how to be a better scientist. I will always admire your persistence and hardworking attitude.

I would like to give special thanks to Dr Eva Balsa-Canto for the immense help and guidance in most of my computational work related to optimal experimental design. Thanks to you and your work, I have learned very much, and I will always be thankful for it. I am also grateful to my second supervisor, Dr Leonardo Rios Solis, for your enthusiasm and guidance. Thanks also to Dr Diego Oyarzún for your guidance and spot-on advice during my annual reviews. Many thanks also to Dr Teuta Pilizota for letting me use some of your facilities and equipment to make microfluidic devices.

I am extremely grateful to all the past and present members of Dr Filippo Menolascina's group. I have learned a lot from all of you, either from the weekly group meetings or else from our many chats. So, many thanks to Brian Philip, David Henriques, Freddie Starkey, Gregg Sutton, Jamie Gilman, Maria Jimenez Ramos, Rose Rae, Sean Bankier, Trevor Ho, Varun Kothamachu and Zhaozheng Hou (George). All my work has been positively influenced by all of you to some extent. I would also like to acknowledge the IBioE community and all the people around the lab. Thanks for always being nice to each other and willing to help anyone that needs it. And special thanks to Kata, the IBioE lab technician, for all your invaluable help over the years.

Finally, I would like to thank all my friends and family for their support during this period. Thanks for being there, even if it was in the distance. And for never giving up on me even when I fail to keep in touch. Last but not least, special thanks to my parents and my sister, who, despite my oddities, love and support me unconditionally. I am who I am because of you. I have achieved what I have, thanks to you.

Contents

Declaration i

Abstract ii

Lay Summary iv

Acknowledgments vii

List of Figures xxii

List of Tables xxiii

1 Introduction 1

1.1 Research Outputs 7

1.2 Thesis Layout 8

2 Background on biological engineering, predictive mathematical modelling and experimental design: a literature review 11

2.1 Engineering of Biological Systems 11

2.1.1 Synthetic Biology 11

2.1.2 The Design-Build-Test-Learn Cycle 13

2.1.3 Fluorescent Proteins 14

2.1.4 Challenges 14

2.2 Microfluidics and Dynamic Cell Stimulation 17

2.2.1 Microfluidics Background 17

2.2.2 Microfluidic Device Fabrication 18

2.2.3 Microfluidics in Biology 19

2.3 Mathematical Modelling in Biological Sciences 21

2.3.1 Evolution of Mathematical Modelling in Biology 21

2.3.2 Types of Mathematical Models 21

2.3.3 Law of Mass Action and Ordinary Differential Equations in Biology . . . 23
2.3.4 Derivation of a Model Describing a Genetic System with Positive Self-Regulation . . . 25
2.3.5 Derivation of a Model Describing a Genetic System with Negative Feedback and External Induction . . . 29
2.3.6 General Model Formulation and Observation Function . . . 33
2.3.7 Bistability in Biological Systems . . . 34
2.4 Parameter Identification and Model Selection . . . 38
2.4.1 Problems in Model Development . . . 38
2.4.2 The Problem of Parameter Estimation . . . 38
2.4.3 Frequentist Approach to Parameter Estimation . . . 39
2.4.4 Bayesian Approach to Parameter Estimation . . . 41
2.4.5 The Problem of Model Selection . . . 42
2.4.6 Parameter Identifiability and Sensitivity Analyses . . . 44
2.5 Optimal Experimental Design . . . 47
2.5.1 Optimal Experimental Design in Biology . . . 47
2.5.2 Frequentist Approach to Optimal Experimental Design for Parameter Estimation . . . 47
2.5.3 Bayesian Approach to Optimal Experimental Design for Parameter Inference . . . 49
2.5.4 Optimal Experimental Design for Model Selection . . . 50
2.6 Aims and Objectives . . . 52

3 Materials and Methods 53
3.1 Materials . . . 53
3.1.1 Chemicals and Reagents . . . 53
3.1.2 Equipment . . . 55
3.1.3 Software . . . 59
3.1.4 Bacteria Strain . . . 59
3.1.5 Yeast Strain . . . 60
3.1.6 Growth Media . . . 60
3.1.7 External Experimental Data for the Genetic Toggle Switch 61
3.2 Methods . . . 64
3.2.1 Mathematical Models and Simulations . . . 64
3.2.1.1 *E. coli* Genetic Toggle Switch . . . 64
3.2.1.2 *S. cerevisiae* Orthogonal Inducible Promoter . . . 69

3.2.1.3 The Repressilator and Hes1 Models 71
3.2.1.4 Simulations and Event Inputs Implementation . . 73
3.2.2 Model Parameter Identification for ODEs 75
3.2.2.1 Frequentist Parameter Estimation 76
3.2.2.2 Bayesian Inference 77
3.2.2.3 Inferred Posterior Quality Assessment 79
3.2.3 Information theory to assess informative content of experiments 81
3.2.4 Goodness of Fit Metrics 83
3.2.5 Model Prediction Uncertainty Quantification 84
3.2.6 Bi-Stability Analysis for the genetic Toggle Switch 86
3.2.7 Optimal Experimental Design 87
3.2.7.1 Optimal Experimental Design for Model Selection 88
3.2.7.2 Frequentist Optimal Experimental Design for Model Calibration 91
3.2.7.3 Bayesian Optimal Experimental Design for Model Inference 92
3.2.8 *E. coli* Culture Growth Conditions and Maintenance . . . 93
3.2.8.1 Toggle switch plasmid transformation 94
3.2.9 *S. cerevisiae* Culture Growth Conditions and Maintenance 95
3.2.10 Fluorescence Microscopy 95
3.2.11 Microfluidic Experiments for Model-Based Bio-Systems Identification 96
3.2.11.1 Microfluidic Devices Preparation 97
3.2.11.2 Microfluidic Experiments for the Synthetic Toggle Switch – *E. coli* 99
3.2.11.3 Microfluidic Experiments for the Synthetic Inducible Promoter – *S. cerevisiae* 101
3.2.11.4 Calibration of the DAW junction 104
3.2.11.5 Image Processing for Microscopy Images from Microfluidic Experiments 105
3.2.11.6 *In Vivo* On-Line Optimal Experimental Design . 107
3.3 Chapter Conclusions 110

4 Tools for Bayesian Parameter Inference of Models Describing Biological Systems: Uncertainty and Information Quantification

of Experiments **111**
4.1 Bayesian Inference of Deterministic Models for Biological Systems 112
4.1.1 Bayesian Inference of the Repressilator and Hes1 Models . 112
4.1.2 Bayesian Inference of the Orthogonal Inducible Promoter Developed in *S. cerevisiae* 115
4.1.3 Bayesian Inference of the Genetic Toggle Switch Developed in *E. coli* 117
4.1.4 Conclusions 123
4.2 *A Posteriori* Informative Content Analysis of Experiments for Biological Systems 124
4.2.1 A Quantitative Analysis of Parameter Knowledge Improvement 124
4.2.2 An Information Theory Approach for Information Content Analysis of Experiments 127
4.2.3 Conclusions 130
4.3 Prediction Uncertainty of Mathematical Models for Biological Systems and Link to Parameter Knowledge 131
4.3.1 A Model Prediction Uncertainty Comparison Between Frequentist and Bayesian Methods 131
4.3.2 Prior Knowledge Propagation to Model Prediction Uncertainty 134
4.3.3 Conclusions 137
4.4 Chapter Conclusions 139

5 Optimal Experimental Design for Model Selection: Stability Analysis of a Genetic Toggle Switch **140**
5.1 Optimal Experimental Design for Model Selection 141
5.1.1 Traditional Goodness of Fit Model Comparison for the Genetic Toggle Switch 141
5.1.2 Frequentist and Bayesian Optimal Experimental Design for Model Comparison 145
5.1.3 Conclusions 150
5.2 Stability Analysis of the Genetic Toggle Switch developed in *E. coli* 151
5.2.1 Stability Properties of Models for the Genetic Toggle Switch at Different Inducer Concentrations 151
5.2.2 Conclusions 154
5.3 Chapter Conclusions 156

6 Fluorescence Microscopy for Microfluidics in Synthetic Biology: Image Processing and Platform Automation 157

6.1 Microscopy Image Processing for Microfluidic Experiments 158

6.1.1 Calibration of the Dial-A Wave Junction to Assess Media Mixing Ratio in Microfluidic Chips 158

6.1.2 Convolutional Neural Networks (U-Net) for Microscopy Image Segmentation 161

6.1.3 Image Processing Automation for Microfluidic Experiments 164

6.1.4 Conclusions 168

6.2 An Experimental Platform for On-Line Optimal Experimental Design for Model Calibration 170

6.2.1 Architecture of a Platform for On-Line Optimal Experiments 170

6.2.2 *In vivo* Frequentist On-Line Optimal Experimental Design Using the Orthogonal Inducible Promoter Developed in *S. cerevisiae* 173

6.2.3 *In vivo* Bayesian On-Line Optimal Experimental Design Using the Genetic Toggle Switch Developed in *E. coli* . . . 177

6.2.4 Conclusions 181

6.3 Chapter Conclusions 183

7 Bayesian Optimal Experimental Design for Parameter Inference: A Computational and Experimental Study 184

7.1 *In silico* Bayesian Optimal Experimental Design for Parameter Inference Exploiting Prediction Uncertainty 186

7.1.1 Bayesian Optimal Experimental Design for Parameter Inference: Proof of Concept 186

7.1.2 *In silico* Validation of Bayesian Optimal Experimental Design for Parameter Inference Using the Genetic Toggle Switch 191

7.1.3 *In silico* On-line Bayesian Optimal Experimental Design . 198

7.1.4 Conclusions 203

7.2 *In vivo* Implementation of Bayesian Optimal Experimental Design for Parameter Inference Exploiting Prediction Uncertainty 206

7.2.1 *In Vivo* Validation of Bayesian Optimal Experimental Design for Parameter Inference Using the Genetic Toggle Switch 207

7.2.2 *In vivo* Implementation of On-line Bayesian Optimal Experimental Design Using the Genetic Toggle Switch . . 217
7.2.3 Optimal Selection of Steady-State Conditions for the Genetic Toggle Switch 225
7.2.4 Conclusions 236
7.3 Chapter Conclusions 239

8 BOMBs.jl: A Julia Package to Ease Simulations, Parameter Inference and Bayesian Optimal Experimental Design for Model Selection and Calibration 241
8.1 Functionality and Architecture of BOMBs.jl With an Illustrative Case Study Using a Model for the Genetic Toggle Switch 243
8.1.1 Model Definition 243
8.1.2 Simulations and Pseudo-Data Generation 244
8.1.3 Maximum Likelihood Estimation 247
8.1.4 Bayesian Inference 248
8.1.5 Optimal Experimental Design 251
8.2 Chapter Conclusions 254

9 Discussion and Conclusions: Limitations and Future Perspectives 256

A Supplementary Data 268
A.1 Supplementary Figures for Chapter 4 268
A.2 Supplementary Figures for Chapter 5 285
A.3 Supplementary Figures for Chapter 6 291
A.4 Supplementary Figures for Chapter 7 293

Bibliography 327

Glossary 362

List of Figures

1.1 Thesis summary graphical abstract 10

2.1 Parallelism between electrical and biological engineering 12
2.2 Excitation and emission spectra of commonly used fluorescent proteins of different colours 15
2.3 Microfluidics device manufacture procedure schematic using soft lithography 18
2.4 Michaelis-Menten and Hill kinetics 22
2.5 Schematics for the hypothetical biological systems used to derive mathematical models 27
2.6 Dynamics of gene regulatory networks presenting positive and negative feedback 31
2.7 Stability analysis of mathematical models 35
2.8 Comparison between frequentist and Bayesian parameter inference strategies 40
2.9 Simplified exemplification of the model selection problem for biological processes 43
2.10 Optimality criteria effect on the parameter confidence ellipsoid . . 48

3.1 Experimental imaging set-up for microfluidic experiments 97
3.2 Schematics of the microfluidic devices used in this work. 98
3.3 DIC microscopy images from bacteria and yeast microfluidic experiments 102

4.1 Bayesian inference of the oscillatory repressilator and Hes1 systems 113
4.2 Bayesian inference of the inducible orthogonal promoter developed in *S. cerevisiae* 116
4.3 Bayesian inference of three models for the genetic toggle switch developed in *E. coli*. 119

4.4 Predictive capabilities of the different toggle switch models inferred on experimental data. 122
4.5 *A posteriori* experiment informative content analysis for four different *in silico* studies 126
4.6 Posterior entropy approximation to assess the informative content of experiments. 129
4.7 Posterior and prediction uncertainty comparison between Bayesian and frequentist methods 133
4.8 Use of prediction uncertainty to assess the *a priori* informative content of experiments 135

5.1 Dependence of model selection on the considered experimental scheme and metric used 142
5.2 Schematic of the proposed strategy for the design of optimal experimental profiles for model selection 145
5.3 Dependence on the convergence curves for optimal experimental design for model selection on the observable(s) and the number of input perturbation steps selected 146
5.4 Optimal experimental design for model selection identifies an experimental scheme that drives the two competing models for the genetic toggle switch to opposite steady-states 149
5.5 Stability analysis of competing models for the genetic toggle switch in frequentist and Bayesian frameworks 153

6.1 Calibration of the Dial-A-Wave (DAW) junction for dynamic experiments 160
6.2 Convolutional neural networks (CNN) for image segmentation of yeast and bacteria microfluidic experiments 163
6.3 Automated image processing of microfluidic experiments in real-time 166
6.4 *In vivo* on-line optimal experimental design for parameter inference 171
6.5 Frequentist on-line optimal experimental design platform 174
6.6 *In vivo* Frequentist on-line optimal experimental design for the orthogonal inducible promoter developed in *S. cerevisiae* 176
6.7 Bayesian on-line optimal experimental design platform 178
6.8 *In vivo* Bayesian on-line optimal experimental design for the genetic toggle switch developed in *E. coli* 180

7.1 Schematic of the proposed strategy for the design of optimal experimental profiles for parameter inference 187
7.2 Bayesian optimal experimental design prof of concept 189
7.3 Bayesian optimal experimental design improves convergence to the true parameters 193
7.4 Bayesian optimal experimental design reduces parameter uncertainties 195
7.5 Bayesian optimal experimental design aids inference of the stability properties of the genetic toggle switch 197
7.6 On-line OED overcomes heavily constrained experimental setups generating highly informative data 200
7.7 Constant inputs can drive a model for the genetic toggle switch to predict two steady-states simultaneously 202
7.8 Microfluidics experimental data for the genetic toggle switch used for parameter inference 208
7.9 Rationale behind the model-based experimental design update of the first OED 209
7.10 Information gain comparison between randomly designed and optimally designed microfluidic experiments 212
7.11 Validation of the inferred mathematical model for the genetic toggle switch 214
7.12 Bayesian optimal experimental design for parameter inference . . 218
7.13 On-line OED overcomes heavily constrained experimental setups in *in vivo* scenarios 220
7.14 Prediction validation of a mathematical model for the genetic toggle switch inferred using data from an on-line OED 221
7.15 Steady-state model validation analysis for the genetic toggle switch 223
7.16 Steady-state model validation analysis for the genetic toggle switch after MLE using reporter ratios as constraints 228
7.17 Microfluidics data model validation analysis for the genetic toggle switch after MLE using observable ratios as constraints 231
7.18 Steady-state observable ratio constraints effect on parameter posterior uncertainty 233
7.19 Steady-state model validation analysis for the genetic toggle switch after Bayesian inference using reporter ratios as constraints 234
7.20 Model validation analysis for a model describing the genetic toggle switch employing microfluidics data for Bayesian inference using observable ratios as constraints 235

8.1 Bayesian optimisation of models for biosystems (BOMBs) 242
8.2 BOMBs.jl automatically generates scripts for model simulations . 243
8.3 BOMBs.jl streamlines reiterative model simulations with minimum user input 245
8.4 BOMBs.jl expands on model simulation automation to generate pseudo-data traces for *in silico* studies 245
8.5 BOMBs.jl eases and automates fast parameter estimation for biomodels using maximum likelihood estimation (MLE) 247
8.6 BOMBs.jl eases and automates Bayesian inference of models using the statistical programming language Stan 249
8.7 BOMBs.jl eases Bayesian inference of models using the Julia package Turing.jl 251
8.8 BOMBs.jl integrates automated optimal experimental design for model selection 252
8.9 BOMBs.jl integrates automated optimal experimental design for parameter inference 253

S1 Repressilator posterior kernel densities against the parameter values used to generate the pseudo-data for experimental sets 1, 2, 3 and 4269
S2 Repressilator posterior (set 1) predictions, against simulations using the parameter values used to generate pseudo-data and pseudo-data, for all states and all data sets 270
S3 Repressilator posterior (set 2) predictions, against simulations using the parameter values used to generate pseudo-data and pseudo-data, for all states and all data sets 271
S4 Repressilator posterior (set 3) predictions, against simulations using the parameter values used to generate pseudo-data and pseudo-data, for all states and all data sets 272
S5 Repressilator posterior (set 4) predictions, against simulations using the parameter values used to generate pseudo-data and pseudo-data, for all states and all data sets 273
S6 Repressilator posterior (set 5) predictions, against simulations using the parameter values used to generate pseudo-data and pseudo-data, for all states and all data sets 274
S7 Bayesian inference of the Hes1 system considering the total Hes1 cell protein as observable 275

S8 Bayesian inference of the orthogaonal inducible promoter developed in *S. cerevisiae* . . . 276
S9 Marginal distribution box plots for each parameter for the toggle switch model M_3 inferred in each experiment considered from Lugagne *et al.* 2017 . . . 277
S10 Marginal distribution box plots for each parameter for the toggle switch models M_1, M_2 and M_3 inferred in all experiments considered from Lugagne *et al.* 2017 . . . 278
S11 Prior predictions for the three toggle switch models considered for the 10 experiments used from Lugagne *et al.* 2017 . . . 279
S12 Posterior predictions for the three toggle switch models considered for the experiments considered from Lugagne *et al.* 2017 . . . 280
S13 *A posteriori* informative content analysis . . . 281
S14 Quantitative informative content analysis of experiments . . . 282
S15 Predictions for the three toggle switch models considered using 8000 samples from the approximated multivariate normal distribution resultant from the frequentist parameter estimation for the experiments from Lugagne *et al.* 2017 . . . 283
S16 Hes1 model predictions for mRNA, cytosolic Hes1 and nuclear Hes1 284
S17 Bar plot for the minus base-10 logarithm of the Bayesian information criteria (BIC) for the three toggle switch models . . . 286
S18 Frequentist predictions for the three toggle switch models considered for the validation experiments from Lugagne *et al.* 2017 . . . 287
S19 Bayesian predictions for the three toggle switch models considered for the validation experiments from Lugagne *et al.* 2017 . . . 288
S20 Frequentist optimal experimental for model selection design results and competing model simulations for optimisations considering 2, 4, 6 and 8 input perturbation steps and for each combination of the observable(s) considered . . . 289
S21 Bayesian optimal experimental for model selection design results and competing model simulations for optimisations considering 2, 4, 6 and 8 input perturbation steps and for each combination of the observable(s) considered . . . 290
S22 Graphical user interface (GUI) for the online image processing platform . . . 292
S23 Parameter priors and pseudo-data used for the Bayesian OED proof of concept . . . 294

S24 Pseudo-data and intuition-driven inferred posterior predictions used for the Bayesian OED proof of concept 295
S25 Pseudo-data and optimally designed inferred posterior predictions used for the Bayesian OED proof of concept 296
S26 Pseudo-data and sequential optimally designed inferred posterior predictions used for the Bayesian OED proof of concept 297
S27 Pseudo-data and inducer input traces generated using the toggle switch model M_3 using the parameter vector Θ_{TS}^{T1} in the computational study for Bayesian optimal experimental design for parameter inference. 298
S28 Pseudo-data and inducer input traces generated using the toggle switch model M_3 using the parameter vector Θ_{TS}^{T2} in the computational study for Bayesian optimal experimental design for parameter inference. 299
S29 Pseudo-data and inducer input traces generated using the toggle switch model M_3 using the parameter vector Θ_{TS}^{T1} in the computational study for on-line Bayesian optimal experimental design for parameter inference. 300
S30 Posterior sample distances to Θ_{TS}^{T1} for each inferred parameter posterior obtained in the computational study for on-line Bayesian optimal experimental design for parameter inference. 301
S31 Stability analysis of the inferred parameter posteriors from the computational on-line OED for parameter inference study 302
S32 Toggle switch model M_3 prior predictions for the microfluidic experiments performed in this work 303
S33 Posterior distributions for the mathematical model of the genetic toggle switch inferred from microfluidics experimental data 304
S34 Posterior distributions for the mathematical model of the genetic toggle switch sequentially inferred from microfluidics experimental data 305
S35 Toggle switch model M_3 predictions for the microfluidic experiments performed in this work using the parameter posterior obtained after inference using the initial pulse-like experiment 306
S36 Toggle switch model M_3 predictions for the microfluidic experiments performed in this work using the parameter posteriors obtained in iteration 2 307

S37 Toggle switch model M_3 predictions for the microfluidic experiments performed in this work using the parameter posteriors obtained in iteration 3 after multi-experimental inference using uninformative priors 308
S38 Toggle switch model M_3 predictions for the microfluidic experiments performed in this work using the parameter posteriors obtained in iteration 3 after a single-experimental inference using as priors the posteriors from iteration 2 309
S39 Posterior distribution comparison for the mathematical model of the genetic toggle switch inferred from microfluidics experimental data following random, off-line optimal and on-line optimal experimental designs 310
S40 Toggle switch model M_3 predictions for the microfluidic experiments after including an on-line OED 311
S41 Steady-state model validation analysis for the genetic toggle switch at iteration 2 312
S42 Model predictions of microfluidic after parameter MLE using the on-line OED with steady-state information as inference constraints 313
S43 Steady-state model validation analysis for the genetic toggle switch after MLE using randomly designed microfluidic experiments and randomly selected steady-state conditions as constraints 314
S44 Steady-state model validation analysis for the genetic toggle switch after MLE using optimally designed microfluidic experiments and randomly selected steady-state conditions as constraints 315
S45 Steady-state model validation analysis for the genetic toggle switch after MLE using randomly designed microfluidic experiments and optimally selected steady-state conditions as constraints 316
S46 Steady-state model validation analysis for the genetic toggle switch after MLE using optimally designed microfluidic experiments and optimally selected steady-state conditions as constraints 317
S47 nRMSE of the genetic toggle switch model using the best parameter estimate according to MLE including steady-state observable rations as constraints when predicting microfluidic experiments 318
S48 Model predictions of microfluidic experiments after parameter MLE using the initial pulse-like experiment with steady-state information as inference constraints 319

S49 Model predictions of microfluidic experiments after parameter MLE using the first random experiment with steady-state information as inference constraints 320
S50 Model predictions of microfluidic experiments after parameter MLE using the first and second random experiments with steady-state information as inference constraints 321
S51 Model predictions of microfluidic experiments after parameter MLE using the first off-line optimal experiment with steady-state information as inference constraints 322
S52 Model predictions of microfluidic experiments after parameter MLE using the first and second off-line optimal experiments with steady-state information as inference constraints 323
S53 Posterior distribution comparison for a mathematical model of the genetic toggle switch inferred using the on-line OED microfluidics data including randomly or optimally selected steady-state observable ratios as constraints 324
S54 Model predictions of microfluidic experiments after Bayesian inference using the on-line OED with steady-state observable ratio information as inference constraints 325
S55 Model validation analysis for a model describing the genetic toggle switch employing microfluidics data for Bayesian inference without using observable ratios as constraints 326

List of Tables

3.1 Chemicals and reagents 54
3.2 Equipment 56
3.3 Software 59
3.4 Growth medium 60
3.5 Information of the experimental sets retrieved from Lugagne *et al.* 2017 and used in this work 62
3.6 Information of the experimental sets retrieved from Lugagne *et al.* 2017 and used in this work on validation steps 62
3.7 Parameters information for the toggle switch models 67
3.8 Parameters values selected for computational studies for the toggle switch model M_3 68
3.9 Parameters information for the orthogonal inducible promoter . . 70
3.10 Parameters values selected for computational studies for the orthogonal inducible promoter model 71
3.11 Parameters information for the Repressilator and Hes1 models . . 72
3.12 Parameters values selected for computational studies for the repressilator model 73
3.13 Parameters values selected for computational studies for the Hes1 model 73

Chapter 1

Introduction

In recent years, life sciences have been adopting quantitative tools and skills from diverse disciplines, beginning the shift of biology to a more predictive field [1]. The new interdisciplinary approach in some areas of the biological community led to the establishment of fields such as bioinformatics, systems biology and synthetic biology. Bioinformatics combines biology with computational sciences to analyse the large amounts of biological data generated through new technological advancements [2]. Its applications range from genome and proteome sequence analysis to detecting complex gene regulatory networks involved in cancer development [3]. Systems biology follows a holistic approach to studying different molecule interactions and regulations of biological systems while adopting mathematical and statistical tools to understand these [4], [5]. The inherent mathematical approach of systems biology allowed the understanding of essential biological processes such as bistability [6] involved in cell differentiation [7]. Synthetic biology follows engineering approaches to characterise biological parts and predictably assemble them to re-design or create new functions into cells [8]. The bioengineering strategy went from constructing completely new genetic circuits [9], [10] to modifying bacteria for targeted drug delivery [11]. What all these relatively new biological areas have in common is the successful merging of biology with quantitative disciplines (e.g. computer science, physics and engineering) while incorporating predictive tools (e.g. mathematical modelling). The result of the transition is a better understanding of biological systems and how to use or modify them to solve problems.

Technological advances played a crucial role in the transition of biology towards a more predictable and quantitative field, allowing scientists an in-depth interrogation of biological processes. New technologies provided us with fast

genome sequencing [12] used to detect, understand and potentially treat genetic diseases [13]. Fluorescence microscopy [14], [15] allows us to examine biological processes *in vivo* by integrating fluorescent proteins into desired pathways [16], [17], [18]. Recent research even allows using fluorescence microscopy to image different types of RNAs to study cell processes such as translation or detection of RNA guided regulation [19], [20]. New experimental systems based on microfluidics, optogenetics or automated culture systems (e.g. chemostats) allow easy and cost-effective *in vivo* examination of cells over extended time periods with dynamic modulations of experimental variables (e.g. temperature, external genetic inducers, nutrients, etc.) [21], [22], [23], [24]. Researchers can study cell stochasticity thanks to equipment that allows single-cell data extraction (e.g. cytometry, microscopy or optogenetic platforms) [15], [22], [25]. Organ-on-a-chip technologies permit replicating *in vitro* the physiology of healthy and sick tissues and organs in microfluidic devices, easing clinical trials of new drugs and therapies [26]. All these new technologies and the large amounts of new and complex data gathered from biological systems call for adequate approaches to analysing the different processes studied. Consequently, competent accurate predictive tools become necessary to understand these systems [27] and, ultimately, modify or design new ones in a cost-effective and automated manner [28].

Mathematical models pose an indispensable tool to predict and understand the underlying mechanisms of biological systems and hence, their control and re-design. Many scientific fields have been using mathematical models as a core tool to advance their knowledge and understanding [29]. Physicists use models to understand physical phenomena, ranging from describing fluid dynamics [30] to the quantum mechanics of subatomic particles [31]. In chemistry, models are used to describe chemical reactions involved in many industrial processes [32]. Engineers use models to design and build from electrical circuits to infrastructures while minimising potential implementation failures [33]. Some biological disciplines also share a successful history of modelling implementation to understand biological processes, such in ecology [34], enzymology [35] and epidemiology [36]. A few other branches are starting to introduce models in their methodologies to reliably predict, control and design biological functions (i.e. systems and synthetic biology). Immediate results from this adoption are the development of biosensors or metabolic path engineering for efficient antimalarial drug production [37]. Yet, while the advances in technology and computational

tools are incessantly growing, the widespread of predictive mathematical models is still, in other biological areas, slow and tedious.

Mathematical modelling still carries a concatenation of obstacles that hinders inexperienced users from taking full advantage of its potential power in understanding and reliably predicting biological systems. While developing mathematical models is a laborious task, efforts of interdisciplinary groups lead to numerous references on how to build mathematical models [27], [38], [39], [40], [41], [42], [43], [44], repositories of already developed models [45], [46], [47], tools to ease their construction [48], [49] and programs to infer them from experimental data [50]. Yet, there are still common roadblocks to their use. First, how should one identify model parameters and which is the best procedure? Mathematical models come with a varying number of parameters which, usually due to technical or practical limitations, are laborious or unfeasible to be measured experimentally. Hence, parameters are commonly inferred from experimental data, a complex non-linear optimisation problem that requires technical skills to be accurately solved [51]. Additionally, there are different ways to tackle the problem. Generally, frequentist approaches trade simplified noise assumptions (for data and parameters) for manageable and faster optimisations. Instead, Bayesian ones relax these assumptions by updating probabilistic knowledge based on observations but at a much higher computational cost and complexity [52]. Second, how should one choose between apparently equally plausible competing models? This problem, commonly referred to as model selection, points to a traditionally dismissed task in the use of models due to its high complexity [53]. Model selection refers to cases where there are different model formalisations [51] or where one model has different parameterisations [54]. Regardless of the situation, the model selection problem is also a non-linear optimisation problem that, if left unaddressed, can lead to unpredictable or wrong models and the loss of time and resources. Finally, which are the best experiments to solve the previous two problems, if these exist? Biological experiments are generally laborious to perform and constrained by time and economic resources. Yet, different experiments carry varying amounts of information for a particular problem [55], [56]. Hence, researchers are left guideless with the overwhelming problem of identifying good experiments to infer model parameters and potentially select between candidate models.

Optimal experimental design (OED) aims to free experimentalists from the

burden of designing highly informative experiments to solve the problems of model selection and identification, amongst others [57], [58]. The field poses an alternative to traditional experimental design using statistics to find the optimum maximally informative experiment concerning a statistical metric [59]. Optimal designs consider variance/uncertainty, or a different statistical metric, of an element from the model (e.g. prediction uncertainty or parameter variance), incorporate real experimental constraints and find the optimal perturbation scheme, sampling or observable needed to solve a problem [60]. Since its boom in the 1950s [61], researchers have been developing different ways to assess the informative content of experiments for diverse problems [57], [62], [63], [64], [65] , contributing to the growth of the methodology. For this reason, many disciplines, such as chemistry [66], chemical engineering [67], materials science [68], or physics [69], adopted OED as a core strategy for experimental design. Results in these other disciplines highlight the usefulness of OED in solving the problems of parameter estimation and model selection while minimising the limited amount of time and resources allocated to experiments. Indeed, OED is spreading into life sciences [21], [70], [71], [72], [73], with an increasing number of new tools, strategies and implementations to efficiently solve the task [74], [75], [76]. However, at present, OED is mainly constrained to theoretical frameworks with few *in vivo* implementations and validations.

Despite the benefits of OED, the complexity of assessing the informative content of experiments before performing them and solving the non-linear optimisation problem [56], [57], [70], [77], [78], hinders the widespread of the methodology amongst inexperienced researchers in life sciences. Researchers have been using the Fisher information matrix (FIM) in frequentist scenarios to approximate the informative content of experiments intended for parameter estimation. This matrix is derived computing the model state sensitivities with respect to a parameter value, and estimates the amount of information that random variables of a process (i.e. observations) provide about the knowledge of model parameters [57]. Since the FIM provides information about the variances and covariances associated with current parameter estimates, OED strategies use some metric of this (e.g. determinant) to design experimental schemes that, in theory, when carried out, will improve parameter estimation [75]. Yet, while the computation and different potential metrics for the FIM are extensively reported in the literature [75], [79], [80], [81], solving the OED problem requires knowledge of non-linear optimisation techniques [57]. These optimisation

problems are difficult to solve, requiring complex algorithms (i.e. meta-heuristic optimisers) and a combination of technical skills still unpopular in some biological areas. In Bayesian approaches, the dominant issue of OED for model inference resides in the *a priori* quantification of the informative content of experiments. Whilst there are different strategies to approximate information content (e.g. mutual information), their excessive computational complexity limits their practical implementation [56], particularly in optimisation schemes which have to evaluate different experimental designs. Hence, Bayesian approaches to OED are generally limited to theoretical scenes [78], [82]. Optimal experiments directed to improve model selection, in both frequentist and Bayesian approaches, usually fall in the same range of complexities to design. Nonetheless, there are fewer research efforts directed at quantifying the informative content of experiments *a priori* [70]. Studies generally focus on approaches to assess the goodness of fit of models (e.g. Akaike information criterion or model evidence), reducing the problem of model selection to find the model that consistently is capable of predicting experimental data [51], [53], [54].

In this thesis, I propose and validate easily implementable OED strategies for model selection and model inference and provide an open-source tool (BOMBs.jl) to promote their use across biological sciences. I use frequentist and Bayesian approaches to infer the parameters of mathematical models describing biological systems while accounting for time-varying stimuli (Chapter 4, p. 111). Thanks to current technological advancements, I focus most of the work on Bayesian parameter inference. I leverage computationally efficient tools (i.e. Stan [83]) to relieve noise assumptions and constrain inferences on observed experimental data [52], [84]. Analyses undertaken with multiple models point to the same conclusion; different experiments provide different amounts of information about the system under study. For this reason, I contribute to the advancement and establishment of OED for model selection and model inference in life sciences by proposing and analysing a design strategy for each. For model selection (Chapter 5, p. 140), I frame the optimisation problem to find an experimental design (e.g. physico-chemical external stimuli) that maximises the distance between the predictions of two competing models. My results identify the limitations of the goodness of fit metrics and highlight the advantages of OED in exploiting inherent properties of a system (e.g. bistable properties of a genetic toggle switch) to discriminate between equally plausible models. For model inference (Chapter 7, p. 184), I propose a Bayesian sampling-based OED method that exploits model

prediction uncertainty to identify optimum experiments to minimise parameter posterior probability uncertainties. First, I test the validity of the methodology *in silico*. My results show the reproducibility of previously published results using alternative methodologies (i.e. mutual information) [56]. Furthermore, my results show the computational feasibility of the strategy to introduce it in an optimisation scheme to design highly informative experiments to infer parameters of a model for a synthetic genetic toggle switch. To reinforce the results and show the advantages of OED in a biological context, I assess the strategy *in vivo*. I used the genetic toggle switch developed by Lugagne *et al.* 2017 [85] to compare random design of experiments to OED in a microfluidic experimental set-up combined with fluorescence microscopy. I developed a fully automated image processing platform (Chapter 6, p. 157) using state-of-the-art image segmentation algorithms (i.e. convolutional neural networks) to process the large amounts of data generated. Consequently, I combined my work into an experimental platform capable of using experiment subsections to update parameter knowledge and re-design the remaining experiment, further improving parameter inference (on-line OED) [86]. To contribute to the widespread use of mathematical modelling in life sciences, I developed and present BOMBs.jl (Bayesian Optimisation of Models for Biosystems), my easy-to-use Julia-based package to aid work with models and experimental design in biology (Chapter 8, p. 241). The package fully automates the simulation of mathematical models with the minimally required model and experiment scheme information from users. In addition, the code allows the use of models with time-varying inputs and extends simulations to the generation of pseudo-data for *in silico* studies. The package relieves the work associated with inferring model parameters by automating maximum likelihood estimation strategies and Bayesian inference. Bayesian inference relies on sampling-based inference algorithms from Stan [83] and Turing.jl. Finally, I incorporated the methodologies proposed for OED for model selection and model inference into BOMBs.jl. The code automates the optimisation schemes and minimises the efforts of users to design highly informative experiments.

1.1 Research Outputs

1. **A cyber-physical platform for model calibration** Bandiera, L.,Gomez-Cabeza,D., Balsa-Canto, E., & Menolascina, F. *Methods in molecular biology (Clifton, N.J.), 2021 vol. 2229.*
   doi: 10.1007/978-1-0716-1032-9_12

2. **Optimally designed model selection for synthetic biology.** Bandiera, L.†,Gomez-Cabeza,D.†, Gilman, J., Balsa-Canto, E., & Menolascina, F. *ACS Synthetic Biology, 2020, 9, 3134-3144.*
   doi: 10.1021/acssynbio.0c00393

3. **A systematic framework for biomolecular system identification.** Tuza, Z., Bandiera, L., Gomez-Cabeza,D., Stan, G., & Menolascina, F. *In Proceedings of the 58th ieee conference on decision and control, 2019.*

4. **Bayesian model selection in synthetic biology: Factor levels and observation functions.** Bandiera, L.†,Gomez-Cabeza,D.†, Balsa-Canto, E., & Menolascina, F. *8th Conference on Foundations of Systems Biology in Engineering FOSBE, 2019.*
   doi: 10.1016/j.ifacol.2019.12.231

5. **Information contentan alysis reveals desirable aspects of in vivo experiments of a synthetic circuit.** Gomez-Cabeza,D.†, Bandiera, L.†, Balsa-Canto, E., & Menolascina, F. *In 2019 IEEE conference oncomputational intelligence in bioinformatics and computational biology (cibcb).*
   doi: 10.1109/CIBCB.2019.8791449

Authors who equally contributed to a publication are marked with a †.

1.2 Thesis Layout

This work covers a wide range of topics directed at using mathematical models to understand, predict and ultimately engineer biological systems. Fig. 1.1 serves as a graphical abstract for the presented work. Researchers might want to study and understand specific biological processes based on hypotheses or use well-characterised genetic parts to engineer novel biological systems. Mathematical modelling is a powerful tool that aids the previously mentioned tasks in a predictive manner. Yet, some issues arise when using these models. If the system under study is unknown to some degree, researchers might need to develop multiple model candidates and face the problem of model selection. These models usually come with unknown parameters that need to be inferred from experimental data. These two common tasks in developing predictive mathematical models describing biological systems are complicated and major issues that limit the widespread use of models. To solve these two problems, researchers need experimental data, which tends to be scarce and noisy. Furthermore, different experiments carry varying levels of information for a specific task. For this reason, optimal experimental design (OED) poses a promising strategy to generate highly informative data while minimising experimental efforts and resources. This strategy can help experimentalists to determine the optimal observables, sampling times, initial conditions or external inputs to apply in an experiment to receive as much information as possible to solve a specific problem. Yet, the implementation of these strategies might prove difficult for inexperienced users, highlighting the importance of the development of easy-to-use tools. This thesis aims to tackle the different aspects mentioned and is structured as follows:

Chapter 2: provides the fundamental theoretical background needed to contextualise the contents of this thesis. The literature review covers the use of mathematical models in life sciences, frequentist and Bayesian statistics for parameter inference, model selection approaches, experiments informative content quantification, optimal experimental design, and microfluidics combined with fluorescence microscopy to interrogate biological systems.

Chapter 3: encloses the different computational and experimental materials (**Section 3.1**) and methods (**Section 3.2**) used in this thesis.

Chapter 4: covers the Bayesian inference of model parameters of four biological systems (**Section 4.1**), introduces analyses of the informative content

between experiments after being performed (*a posteriori*) to spotlight the informative difference of multiple experimental designs (**Section 4.2**) and seeks to investigate the relationship between model prediction and parameter uncertainties and its potential use in Bayesian optimal experimental design (**Section 4.3**).

Chapter 5: looks into traditional model selection strategies to highlight the advantages of OED in generating highly informative experimental data for the task (**Section 5.1**). The chapter further investigates this results, analysing the stability properties of inferred models for a bistable genetic toggle switch (**Section 5.2**).

Chapter 6: introduces the fluorescence microscopy image processing procedures needed to process microfluidics experimental data (**Section 6.1**) and integrates the previous computational work to build a microfluidics experimental platform capable of inferring parameters from a portion of the processed data and re-design the remaining experiment on-line (**Section 6.2**).

Chapter 7: builds upon the use of prediction uncertainty for Bayesian OED, examining the strategy *in silico* (**Section 7.1**) and implements the Bayesian OED strategy *in vivo* through microfluidics using a genetic toggle switch as a case study (**Section 7.2**).

Chapter 8: describes BOMBs.jl, a Julia package to automate work with models, inference of parameters and optimal experimental design.

Chapter 9: provides a conclusions section of the work presented and builds upon a discussion of the presented results to provide the future directions of mathematical modelling in life sciences.

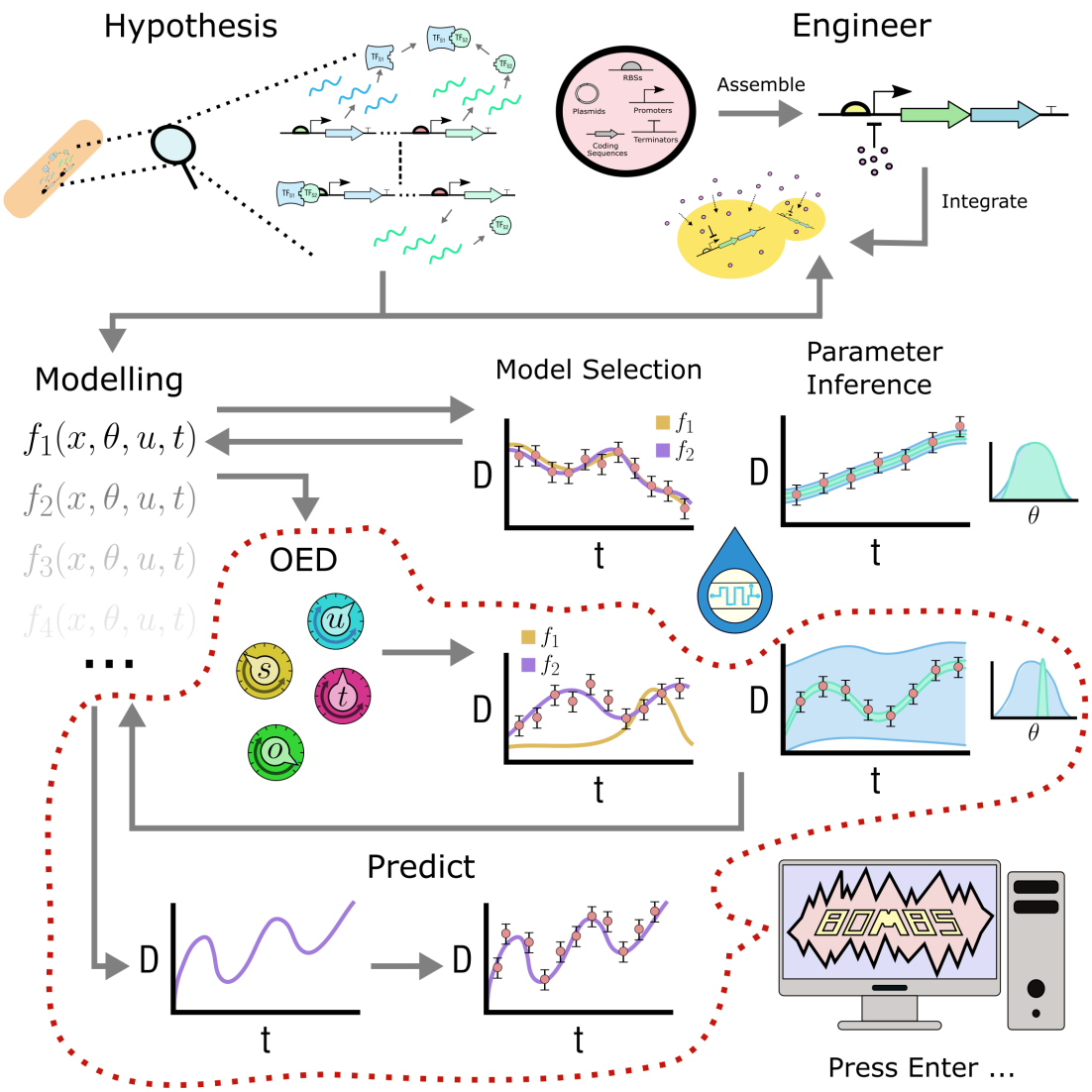


Figure 1.1: Thesis summary graphical abstract. Mathematical modelling can improve the understanding of biological systems, guide the design of new ones and allow their predictive control. In the presence of multiple model candidates, researchers need to identify the best candidate model while still needing to infer the model parameters from experimental data. Optimal experimental design (OED) allows the detection of the most informative experiments intended to solve both problems. This strategy proves to be of higher efficiency minimising the time and resources allocated to experiments. New experimental platforms (e.g. microfluidics) permit dynamic cell stimulation to implement the informative designs. BOMBs.jl is an open-source Julia-based package that aids the automation of model simulations, parameter inference and OED.

Chapter 2

Background on biological engineering, predictive mathematical modelling and experimental design: a literature review

2.1 Engineering of Biological Systems

2.1.1 Synthetic Biology

Modern predictive engineering of biological organisms began in the late 1960s/1970s with the development of recombinant DNA technologies [87], [88]. From restriction endonucleases [89], [90] to more recent strategies using the CRISPR-Cas9 system[i] [91], [92], these technologies allowed the introduction and deletion of genes and other DNA sequences (e.g. promoters) into a wide variety of organisms. The resultant genetically modified organisms had a high impact on our society, with the efficient and sustainable production of high-interest compounds such as antibiotics [93], vaccines [94], hormones [95], drugs [96] or pest-resistant crops [97], amongst many others. Due to the high relevance of the products generated, many efforts are focused on pathway optimisation for high-yield compound production [98], [99], [100], [101], [102], [103]. Yet,

[i]Originally found in prokaryotic organisms as an infection defence mechanism against viruses, a guide RNA drives an enzyme to remove specific DNA sequences.

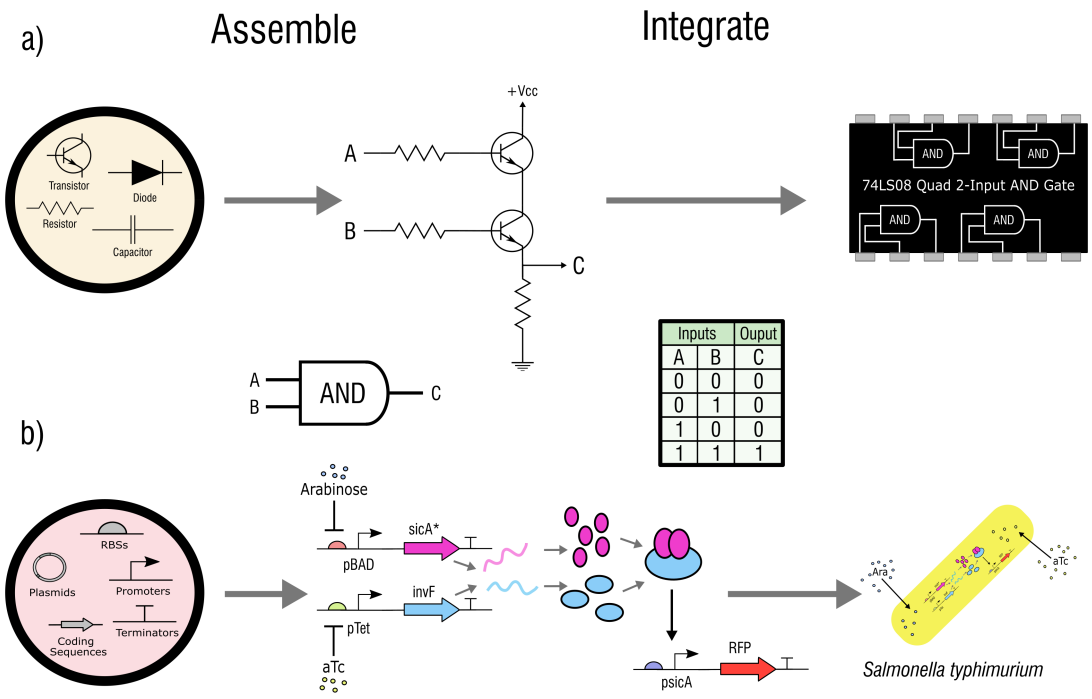


Figure 2.1: Parallelism between electrical and biological engineering. a) Use of electrical components to design a logical AND gate and integrate it into a 74LS08 Quad 2-Input AND Gate. b) Use of genetic parts to design a genetic AND gate in *Salmonella typhimurium* from Moon *et al.* 2012 [104].

this traditional approach to genetic engineering is limited to already existing functions, pathways and compounds from cells.

The first utterly new and functional synthetic gene regulatory networks were engineered in the year 2000 by Elowitz and Leibler (i.e. the oscillatory repressilator, where a mathematical model is studied in Chapter 4 of this thesis) [10] and by Gardner *et al.* (i.e. the genetic toggle switch, system extensively studied throughout this thesis in Chapters 4, 5, 7) [9]. Since then, researchers have developed a wide range of new systems directed to understanding biological processes and solving potential health and environmental problems [105], usually under the wing of synthetic biology [8]. The generation of synthetic oscillators and genetic switches redirected the view on genetic engineering, allowing the introduction of many other completely new functions into cells. Researchers can build new systems into a cell host (chassis) by combining well-characterised parts (e.g. promoters, terminators or coding sequences for fluorescent proteins) in a top-down approach (Fig.

2.1)[ii] [106], [107], [108]. The use and combination of genetic feedback[iii] and feedforward[iv] loops allowed a better understanding of biological processes [109], [110] and information processing in cells [111], [112]. Furthermore, the incorporation of control theory permitted the building of more robust synthetic systems [113], [114], [115], the design of genetic logic gates [28], [116], and the generation of biosensors [117], [118] or the development of targeted drug delivery bacteria [11], [119]. Alternatively, more recent bottom-up approaches[v] use genetic parts to construct new systems of increasing complexity from scratch without a cell host [120]. These allow the generation of cell-free systems[vi] to study biochemical reactions while reducing complexity and removing cellular context [121], [122], [123]. For example, cell-free systems proved a promising path to generate new or improved materials [124], [125] and pollutant or disease sensors [126], [127]. Other synthetic biology bottom-up explorations seek to better understand the complexity of life by building the genome of an organism [128] or understanding the minimal cellular elements and reactions that form life [129].

2.1.2 The Design-Build-Test-Learn Cycle

A core concept in synthetic biology borrowed from engineering is the Design-Build-Test-Learn (DBTL) cycle, which allows an efficient iterative workflow in engineering biological systems [130]. Nonetheless, the characterisation of genetic parts (e.g. plasmids, promoters, terminators or coding sequences) [131] is crucial for the predictive design of functional engineered biological systems [132], [133]. Researchers in the field have directed lots of efforts to generate libraries of tested parts with functions and input/output relationships (e.g. BioBricks) in diverse cellular contexts [134]. Mathematical modelling plays a fundamental role in the adequate characterisation of these parts so these can be assembled in predictive and functional gene circuits [135], [136]. Accurate depiction of these genetic parts and mathematical modelling can become of essential importance in the new era of automated machine-made designs [28], [137]. All these characterisation efforts allow the

[ii] To break down a system into smaller parts to understand it.
[iii] The output of a system is also its input.
[iv] An input directly regulates an intermediate element and both regulate the final element.
[v] To put together characterised pieces to create more complex systems.
[vi] In vitro tool where cellular reactions can occur outside of a cellular context.

integration of these systems into cell hosts, testing them in many traditional (e.g. flow cytometry) or novel (e.g. microfluidics) experimental platforms [130]. Next, computational tools aid the extraction of information from these new synthetic systems and can point to the best next design to achieve a goal [138], [139].

2.1.3 Fluorescent Proteins

Quantitative science needs a measurable readout after an input is applied to the system to assess the functionality of many natural and synthetic biological processes. Fluorescent proteins allow tagging of tissues, cells, cellular compartments and specific molecules for local and temporal studies [140]. These molecules, commonly found in some sea creatures and bacteria, emit light when excited at a determined wavelength (λ) spectrum of higher energy due to loss of energy in the process (Stokes shift), which then can be measured with sensitive laboratory equipment (e.g. plate readers, microscopes or flow cytometers) [18]. Hence, fluorescent proteins are used extensively to assess synthetic circuits for protein labelling or promoter activity [28], [85], [141], [142]. In this work, fluorescent proteins and microscopy are amply used to investigate the dynamics of synthetic promoters and systems (see Chapters 6 and 7). Since the discovery of green fluorescent proteins in the early 1960s [143], a variety of new molecular fluorophores emerged (Fig. 2.2), accommodating different research needs such as multiple fluorescence tagging [144] or using orange-red fluorescent proteins to reduce photo-toxicity in prolonged measurements [145]. Such is the importance of these proteins nowadays that, over the last years, many efforts have been directed at engineering existing fluorescent proteins to be more stable, have shorter maturation times, be more red-shifted or have higher quantum yield[vii] [146], [147].

2.1.4 Challenges

Despite all the potential advances and benefits of synthetic biology in solving health, fuel or pollution problems, there still are limitations on what can we genetically engineer or the implications of introducing a synthetic circuit into a cell host. Combining different components into a system and placing them into a chassis might result in undesired results if these get introduced into a different cellular

[vii] Ratio of absorbed to emitted photons.

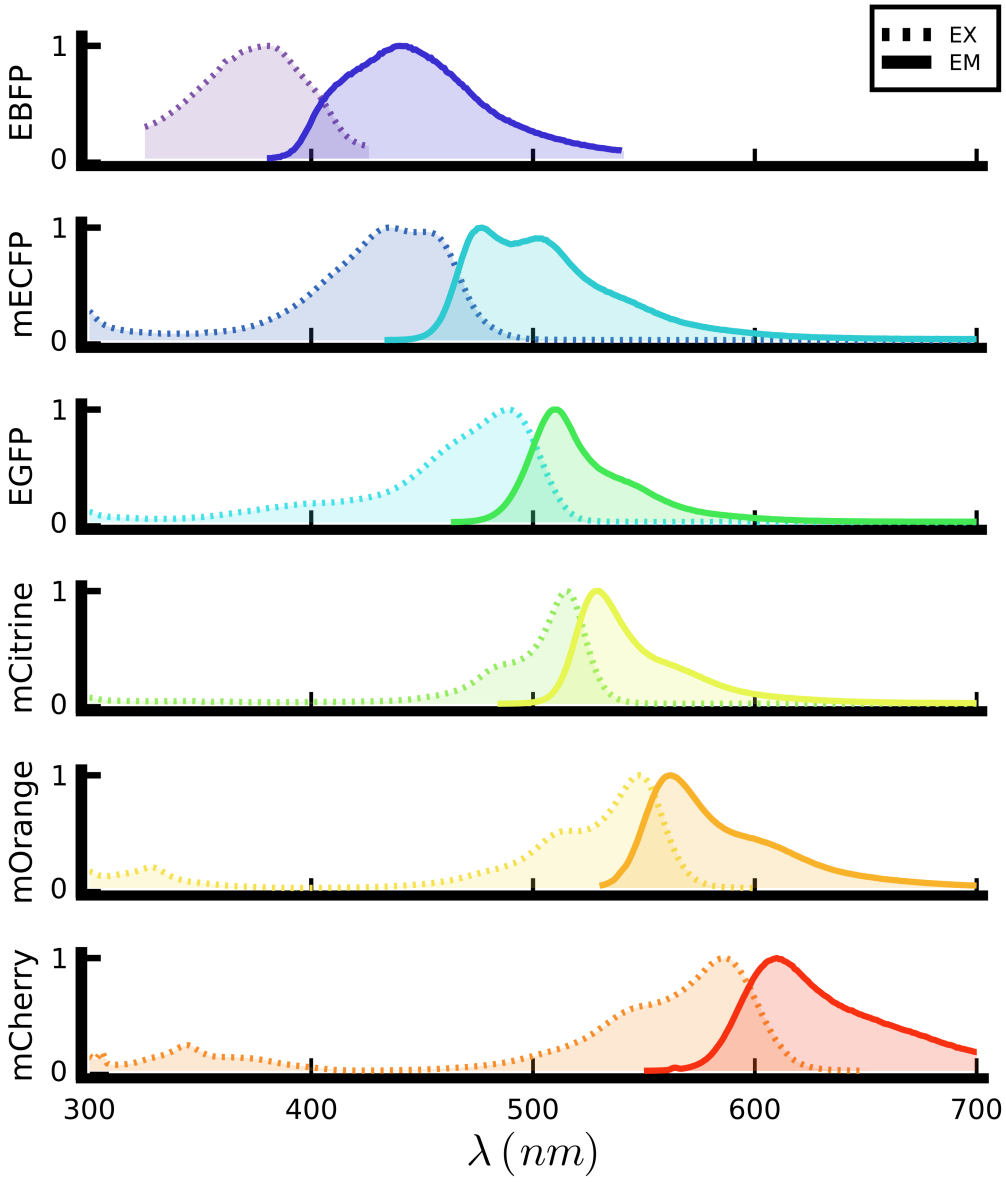


Figure 2.2: Excitation and emission spectra of commonly used fluorescent proteins of different colours. Proportion of excitation (EX) and emission (EM) of an example for blue (EBFP), cyan (mECFP), green (EGFP), yellow(mCitrine), orange (mOrange) and red (mCherry) fluorescent proteins.

context from the one each part got characterised [148], [149]. These inappropriate and unpredictable designs can lead to a smaller dynamic range of the system outputs [150], cross-talk with other cell elements (lack of orthogonality) [151], undesired genetic expression due to inadequate monocistronic[viii] designs [152] or undesired loss of function due to homologous recombination [153] or mutations [154]. On top of that, even with well characterised orthogonal systems, these are introduced into living cells depending on the host machinery (e.g. ribosomes), generating potential resource allocation issues with catastrophic effects on cell growth, division or viability [109], [155], [156]. While cell-free systems can present an attractive solution to these problems, these also present limitations on the scalability of the synthetic designs, stable long-term reactions or regeneration of necessary intermediate molecules [122], [157]. In addition, there are general concerns of unpredicted biosafety hazards of not well characterised systems [158]. For these reasons, it is crucial to consider the scalability of systems, invest in an extended range of well characterised hosts and parts libraries, and the introduction of standard methodologies [105]. To do so in a predictable manner, mathematical modelling, adequate data measurement and processing platforms, and computational tools are of the essence while reducing the number of iterations needed in the DBTL cycle to obtain successful results [21], [159], [160], [161], [162], [163].

[viii] One gene transcribes to one mRNA which translates to a single protein. Polycistronic mRNA is more common in prokaryotes.

2.2 Microfluidics and Dynamic Cell Stimulation

2.2.1 Microfluidics Background

Microfluidic technologies for cell biology use devices that enclose channels containing media that follows laminar flow (low Reynolds numbers[ix]) that can have traps to contain cells. Over the last years, microfluidic systems have grown in popularity amongst biological sciences [164]. These allow investigation of single-cells or small groups of cells over long periods of time while using low reagent volumes and allowing high-throughput analysis and data-gathering [165]. Moreover, microfluidic set-ups permit accurate spatial and temporal modulation of media and external inducers presented to cells within an experiment leveraging laminar flow [166] as demonstrated in Chapters 6 and 7 of this thesis. This precise spatiotemporal control allows dynamic interrogation of natural and synthetic systems in cells, unattainable with more traditional technologies (e.g. plate readers or flow cytometry) [21]. Microfluidic technologies allow studying cellular growth and division [167], gene expression [168] or the temporal behaviour and control of synthetic circuits [85].

Thanks to the micron scale size of microfluidic device channels, the characteristic laminar flow allows single-cell analysis and distribution in specific regions of the chips [169], quick media changes [85] and spatial distribution of media [170]. Laminar flow is achieved thanks to the down-scaled size of the channels in microfluidic chips, which reduces the influence of inertial forces following the Reynolds number Re. This number is defined as the product of the fluid (i.e. media) density ρ, the fluid speed v and the characteristic length L (e.g. hydraulic diameter of a channel) divided by the fluid dynamic viscosity μ (i.e. $Re = \frac{\rho v L}{\mu}$) [171]. Due to the reduction in the characteristic length and fluid speed in microfluidic devices, Re is typically below 2000[x], ensuring laminar flow. Since the diffusion time for a molecule decreases proportionally to the decrease of the characteristic length L, the laminar flow also permits the formation of static and dynamic media gradients [166]. Consequently, in the case where fast media mixing is desired, such as to achieve a specific mixing ratio between two different media sources, microfluidic chips need to integrate mixing microchannels [170].

[ix] Ratio between the inertial force of a fluid to its viscosity. High Reynolds numbers indicate turbulent flows, while low ones indicate laminar flows.

[x] The Reynolds number Re is dimensionless.

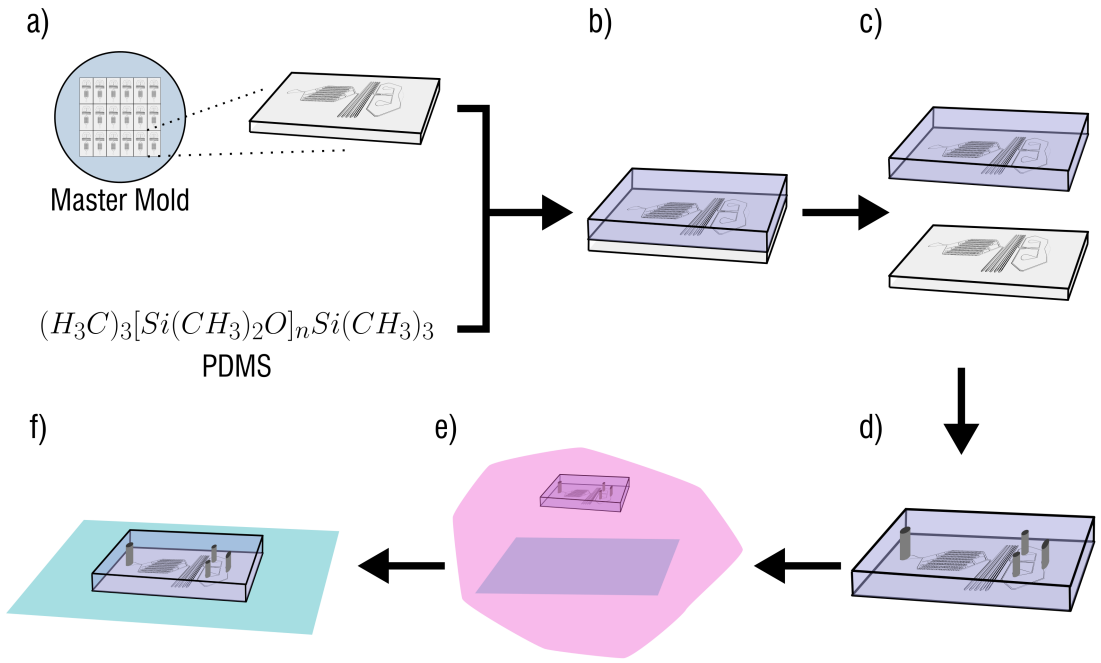


Figure 2.3: Microfluidics device manufacture procedure schematic using soft lithography. a) Pouring of PDMS into a master mold containing multiple photoresist patterns for a determined microfluidics device design. b) Air bubbles are removed, and the master mold with PDMS is placed into an oven at $65°C$ for 3 hours for curing. c) The cured silicon rubber mould is removed from the master mold, and each chip is cut. d) Inlets are drilled with a biopsy punch (or similar) of adequate size. e) After cleaning the devices and glass coverslips, the pairs are plasma treated in a plasma cleaner to activate the surfaces. f) Bonding of individual chips into glass cover slides.

2.2.2 Microfluidic Device Fabrication

Over the years, microfluidic tools have become more accessible thanks to cheap and easy to use methodologies to fabricate microfluidic devices, such as soft lithography of polydimethylsiloxane (PDMS)[xi]. In this type of strategy, a device design (a negative image) is printed using photoresist[xii] layers into a master mold through photolithography[xiii] [170]. Generation of a master mold with

[xi] Silicon-based organic polymer.
[xii] Light sensitive material to generate patterns into a substrate in the micro and nano scale.
[xiii] Process to generate small (micro and nano scale) patterns into a substrate or wafer using light.

photolithography allows custom multilayered patterns, with the resolution of μm, designed with computer-aided design software such as AutoCAD [164]. The drawback of generating such moulds is the need for cleanroom facilities and trained personnel, which might be unavailable for some groups [166]. Nonetheless, a growing number of companies provide the service where users only need to provide the designs to receive the desired master mold [164]. Once the master mold is obtained, this can be reused for the soft lithography process[xiv] (Fig. 2.3), making the generation of microfluidic devices easy, fast and inexpensive [171]. Hence, liquid PDMS (mixed with curing agent[xv]) can be poured into the mold, cured, disassembled when solid, cut, inlet pores drilled and device bonded to a cover slide for use in about a working day (see Section 3.2.11.1 p. 97 of Chapter 3 for more details). Different strategies and materials can be used to build microfluidic devices [171], [172], [173]. Yet, the cheap, easy and fast tractability of PDMS combined with its bio-compatibility, gas permeability, transparency and accessibility makes PDMS soft lithography the preferred and most accessible option for biological studies [164], [174], [175].

2.2.3 Microfluidics in Biology

Microfluidics is commonly combined with microscopy to obtain real-time cell measurements over extended time periods in an automated fashion [21], [24], [164], [165]. Microscopy allows high-resolution imaging of fluorescent components (e.g. fluorescent proteins) and spatiotemporal interrogation of cells enclosed in microfluidic devices [21]. In addition, microfluidic setups permit close temperature and media administration control over time [165]. A microscope stage incubation box or cage grants temperature control and additionally can provide gas concentration control (e.g. CO_2 control for mammalian cell cultures) [166], [176]. External valves, pumps or syringes (leveraging hydrostatic pressure) allow media administration control in real-time [24], [170]. The delivered media can contain external inducers required to induce or repress a function in cells (dynamic stimulation) [85], [177]. This work leverages such conditions to study a genetic toggle switch developed in bacteria (see Chapters 6 and 7). Such is the potential of microfluidics combined with microscopy that can aid the development of cell-free systems [178] and

[xiv] Before the first use a release agent (such as Trimethylchlorosilane or (Tridecafluoro-1,1,2,2-Tetrahydrooctyl)-1-Trichlorosilane) needs to be applied to facilitate the detachment of PDMS.

[xv] PDMS is a two component polymer, constituted by an elastomer and a curing agent.

single-cell genotyping of cell libraries [179]. This strategy also allows the study of cellular environmental responses (e.g. taxis) [180], interactions between different microorganisms [181] or interactions between bacteria and higher-order organisms, such as fungi [182]. Dynamic cell stimulation has been applied to implement control algorithms leveraging gene regulation [177], [183] and investigating and controlling synthetic gene networks [85], [184], [185]. Droplet microfluidics[xvi] aids the automation of the DBTL cycle [186], [187] by allowing DNA assembly [188], genetic transformation and transfection [189], [190] or cell sorting [191], [192]. Finally, microfluidics is also reshaping healthcare, contributing to drug development, personalised medicine, and illness understanding by studying disease models and pharmaceutical drug effects using organ-on-a-chip devices [193], [194], [195], [196].

[xvi] Generation and manipulation of contained droplets in immiscible phases in microfluidic devices.

2.3 Mathematical Modelling in Biological Sciences

2.3.1 Evolution of Mathematical Modelling in Biology

Mathematical modelling is a central tool to understand the processes underneath experimental observations and predict the behaviour of newly designed systems. Two of the earliest, yet still relevant, modern mathematical formalisations of biological processes are the descriptions of 1) enzyme catalysed substrate to product reactions by the Michaelis-Menten model [197], and 2) the cooperative binding of oxygen to hemoglobin by the Hill equation [198] (Fig. 2.4). Since the development of these models at the beginning of the 20th century, mathematical modelling has been steadily adopted by an increasing number of biological disciplines. Some notable examples are ecology (e.g. predator-prey interactions), epidemiology (e.g. malaria dynamics), physiology (e.g. action potential in neurons) or population genetics (e.g. evolutionary dynamics) [199], [200], [201]. Nonetheless, the holistic approach of systems biology made the substantial popularisation of this predictive tool at the beginning of the 21st century, describing more complex and fundamental biological processes [199]. Such growth in using mathematical models allowed a better understanding of core biological processes. Modelling provided a deeper comprehension of cell differentiation [202], bioproduction networks [203], pathogen infection mechanisms [204] and genetic regulation [205], which allowed the new era of predictive design of new synthetic systems [28], [206].

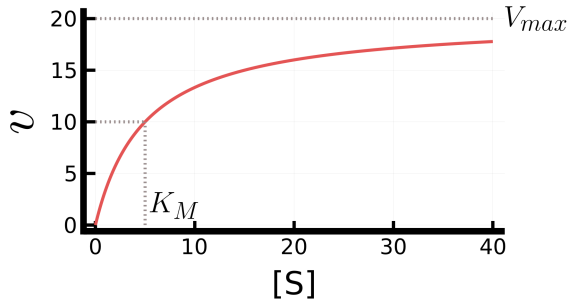
2.3.2 Types of Mathematical Models

Different mathematical formulations exist to abstract the behaviour of biological systems with a more or less, yet necessary, level of detail. A general modelling categorisation discriminates between stochastic and deterministic models to describe dynamical systems [207].

Stochastic models describe systems introducing random (noise) variables in the equations to account for the heterogeneous behaviour of biological processes at the single-cell level [207]. The most common stochastic approach to describing biochemical networks is the use the chemical master equation [208]. The chemical

Michaelis-Menten:

$$v = \frac{V_{max}[S]}{K_M + [S]}$$



Hill:

$$y = \frac{y_{max}[L]^n}{K_d^n + [L]^n}$$

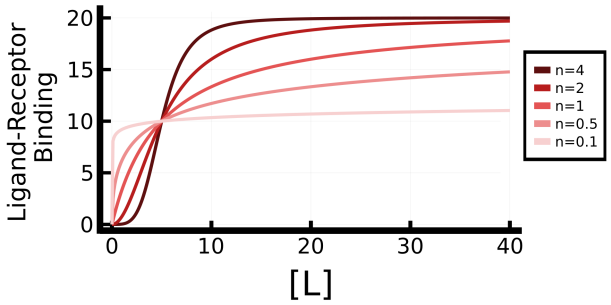


Figure 2.4: Michaelis-Menten and Hill kinetics. General formulations for the Michaelis-Menten (top-left) and activation Hill (bottom-left) functions. Example of reaction rate v (Michaelis-Menten equation) and ligand-receptor binding y (Hill equation) as a function of the concentration of substrate [S] (top-right) and ligand [L] (bottom-right). For this example, the maximum product rates V_{max} and y_{max} were set to 20, and the Michaelis-Menten constant K_M and the activation coefficient K_d were set to 5. I used five different values for the Hill coefficient n (n = [0.1, 0.5, 1, 2, 4]).

master equation is a set of differential equations[xvii] that describe the temporal evolution of probabilistic chemical reactions describing systems that can be found in different states, usually using underlying Markov processes[xviii] [209]. However, while being more faithful representations of some biological processes due to variability caused by a low number of reaction components (intrinsic noise) or environmental interactions (extrinsic noise) [210], stochastic modelling is still constrained by many implementation challenges. The analytical solution of the chemical master equation is often intractable, and even if an approximation can be estimated, this still comes at a high computational cost [211]. Moreover, parameter inference of stochastic models tends to be a complicated and, again, a

[xvii] Equations containing the derivative of one or more unknown functions commonly used to represent the rate of change of states with respect to one or more variables.

[xviii] Random process where potential future events only depend on the current event state.

computationally expensive labour, relying on fitting assumptions that decrease the model prediction capabilities [210]. Hence, researchers should carefully assess if adding stochasticity when developing a model for a biological process is needed [212].

In contrast, deterministic models exclude the random element making the predictions of these entirely determined by the equations, their initial conditions and model parameters [211]. Ordinary differential equations (ODE) and partial differential equations (PDE) are generally used as mathematical descriptions of biological systems using a deterministic framework. ODEs are differential equations that represent the change rate of a state concerning one variable (usually time), while PDEs include more than one independent variable (e.g. temperature or spatial location) [213]. Nonetheless, deterministic modelling comes at the expense of more restrictive assumptions in biological scenarios. The main common ones are using continuous variables to represent discrete entities and the loss of stochasticity of intrinsically stochastic biological systems. These assumptions hold in situations where there are abundant elements in the process (i.e. high population size), such as in population-based models in contrast to single-cell models [214], [215]. Consequently, these assumptions also become a prime consideration when developing models for biological systems [212]. Yet, deterministic modelling (especially employing ODEs) is a common approach in biological sciences [27] [213], [216], [217]. Compared to stochastic models, deterministic differential equations are easier to use, fit and computationally less expensive to simulate [213], [218]. Moreover, thanks to their traditional use in many other disciplines (e.g. physics or engineering), extensive literature and tools focused on solving and simulating these models already exist [211]. Throughout this work, I will focus on deterministic ODEs to study biological systems at a population level or considering scenarios with high molecular populations (see Section 3.2.1, p. 64 of Chapter 3).

2.3.3 Law of Mass Action and Ordinary Differential Equations in Biology

Mathematical models describing biological processes and the interaction between different components can be derived from their chemical reactions in the form of ODEs by following the law of mass action [216]. The law of mass action states

that the rate (usually measured in inverse units of time) at which a given reaction occurs must be proportional to the product of the concentration of the different reactants following stoichiometry[xix] [213] [219]. For example, if a reversible biochemical reaction involves the chemical species A and B to form the compound C, the reaction can be represented as:

$$[\mathrm{A}] + [\mathrm{B}] \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} [\mathrm{C}]$$

where k_1 is the rate of formation of [C] and k_{-1} the rate of dissociation to [A] and [B]. Deterministic models account for concentrations of molecules expressed in number of molecules per unit of fixed cell volume (e.g. moles/mL = mM) rather than number of molecules, as some stochastic models do [220]. We can then define a system of ODEs as:

$$\begin{aligned} \frac{d[\mathrm{A}]}{dt} &= k_{-1}[\mathrm{C}] - k_1[\mathrm{A}][\mathrm{B}] \\ \frac{d[\mathrm{B}]}{dt} &= k_{-1}[\mathrm{C}] - k_1[\mathrm{A}][\mathrm{B}] \\ \frac{d[\mathrm{C}]}{dt} &= k_1[\mathrm{A}][\mathrm{B}] - k_{-1}[\mathrm{C}] \end{aligned} \tag{2.1}$$

where the rate of change of a molecular species over time depends on the production rate and concentration of molecules that form it minus its rate of dissociation and concentration (reverse reaction). This approach also allows the introduction of non-linearities in the models (e.g. Michaelis-Menten equations or Hill function). For instance, the dimerisation of a biochemical molecule D follows the reaction:

$$[\mathrm{D}] + [\mathrm{D}] \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} [\mathrm{D}_2]$$

Following the same procedure as before, we can define the system of ODEs to describe the dimerisation dynamics over time as:

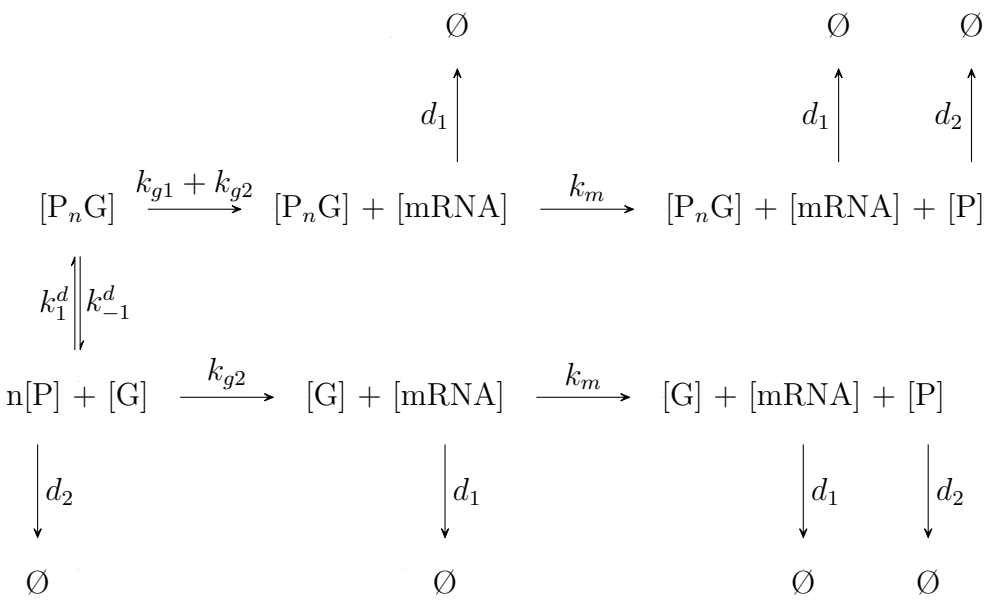
[xix] Relationship between the amount of reactants and products in a chemical reaction so to follow the laws of conservation of mass and energy.

$$\begin{aligned} \frac{d[\mathrm{D}]}{dt} &= 2k_{-1}[\mathrm{D_2}] - 2k_1[\mathrm{D}]^2 \\ \frac{d[\mathrm{D_2}]}{dt} &= k_1[\mathrm{D}]^2 - k_{-1}[\mathrm{D_2}] \end{aligned} \tag{2.2}$$

Note that the terms in the rate equation $\frac{d[\mathrm{D}]}{dt}$ are multiplied by 2, since two molecules are involved in both reactions [213]. This would be equivalent to the equations $\frac{d[\mathrm{A}]}{dt}$ and $\frac{d[\mathrm{B}]}{dt}$ from the previous example. The situation can be generalised by considering an n number of molecules D forming the complex $\mathrm{D_n}$. These scenarios can happen when describing the cooperative binding of molecules or binding of n molecules of a transcription factor into an inducible promoter [213], [216]. A property of these systems is that, after enough time passes, reactions reach an equilibrium point (unless there is an input of energy or mass) where the number of molecules for a given species remains constant because the forward reaction rate equals the backwards one. In this situation, the time derivatives are equal to zero, leaving the system at steady-state. If only some reactions in the system are at equilibrium, this is at a quasi-steady-state [221]. All this derivations and approximations are used to derive all the mathematical models used in this thesis (see Section 3.2.1, p. 64 of Chapter 3)

2.3.4 Derivation of a Model Describing a Genetic System with Positive Self-Regulation

Using the law of mass action as previously shown, we can describe more complex biological systems such as gene regulations following positive or negative feedback. For example, we can represent a positive self-regulation of a gene (positive feedback) as:



The top half reactions represent the transcription and translation of a protein P subject to positive regulation by itself, while the bottom half represents the basal transcription (and translation) for the protein. In the presence of protein P, an n number of this can bound (with constant rate K_1^d) to the gene promoter G, positively regulating its own transcription (with constant rate k_{g1}). In the absence of protein P (or when this unbounds with constant rate K_{-1}^d), the basal transcription (promoter leakiness) occurs at a rate k_{g2}. Following the central dogma of molecular biology, the transcribed mRNA translates to protein P at a constant rate k_m and, at any point, mRNA and protein P degrade at rates d_1 and d_2, respectively [213], [222], [223], [224]. Note that, for representation simplicity purposes, I excluded association/dissociation of the complex [P_nG] in the transcription and translation results (middle and right), but these should also be considered there since they can happen at any point of the process. Fig. 2.5a shows a schematic representation for this hypothetical system. At this point, we can describe the mRNA and protein P dynamics with ODEs as:

$$\begin{aligned} \frac{d[\mathrm{mRNA}]}{dt} &= k_{g2}[\mathrm{G}] + k_{g2}[\mathrm{P_nG}] + k_{g1}[\mathrm{P_nG}] - d_1[\mathrm{mRNA}] \\ \frac{d[\mathrm{P}]}{dt} &= k_m[\mathrm{mRNA}] - d_2[\mathrm{P}] \end{aligned} \tag{2.3}$$

where knowing that the total amount of gene promoter G ([$\mathrm{G_T}$]) is distributed between the bound and unbound states to protein P, [$\mathrm{G_T}$] = [G] + [$\mathrm{P_n}$G]. Hence, the ODE describing the mRNA dynamics can be rearranged as:

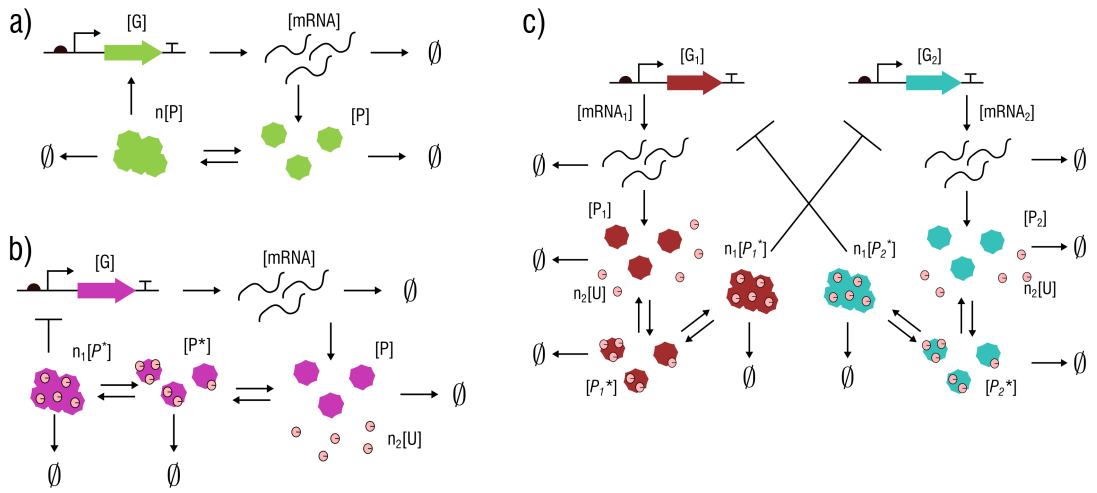


Figure 2.5: Schematics for the hypothetical biological systems used to derive mathematical models. Diagrams representing the proposed genetic systems with positive self regulation (a), with negative feedback and external induction (b) and with double negative feedback and external induction (c) used to derive equations 2.8, 2.14 and 2.17, respectively. The external inducer U is introduced from outside the system instead of being produced by an element from the system. While I choose to represent protein complexes that bind to the gene promoter, this could also be individual proteins binding to the promoter in a sequence.

$$\begin{aligned}\frac{d[\mathrm{mRNA}]}{dt} &= k_{g2}[\mathrm{G}] + k_{g2}[\mathrm{P_nG}] + k_{g1}[\mathrm{P_nG}] - d_1[\mathrm{mRNA}] = \\ &= k_{g2}([\mathrm{G_T}] - [\mathrm{P_nG}]) + k_{g2}[\mathrm{P_nG}] + k_{g1}[\mathrm{P_nG}] - d_1[\mathrm{mRNA}] = \\ &= k_{g2}[\mathrm{G_T}] - k_{g2}[\mathrm{P_nG}] + k_{g2}[\mathrm{P_nG}] + k_{g1}[\mathrm{P_nG}] - d_1[\mathrm{mRNA}] = \\ &= k_{g2}[\mathrm{G_T}] + k_{g1}[\mathrm{P_nG}] - d_1[\mathrm{mRNA}] = \\ &= \alpha + k_{g1}[\mathrm{P_nG}] - d_1[\mathrm{mRNA}]\end{aligned} \tag{2.4}$$

Since $[\mathrm{G_T}]$ remains a constant over time, we can group the term $k_{g2}[\mathrm{G_T}]$ into the parameter α to describe the basal expression (i.e. promoter leakiness) of the system [225]. To deal with the term containing $[\mathrm{P}_n\mathrm{G}]$ and accounting for the time scale separation with the rest of the processes, a common approach is to assume that the association and dissociation of the complex are at steady-state (quasi-steady-state approximation) [211], [226], so:

$$\begin{aligned} \frac{d[\mathrm{P_nG}]}{dt} &= k_1^d[\mathrm{G}][\mathrm{P}]^n - k_{-1}^d[\mathrm{P_nG}] = 0 \\ k_1^d[\mathrm{G}][\mathrm{P}]^n &= k_{-1}^d[\mathrm{P_nG}] \end{aligned} \tag{2.5}$$

If we know that the total amount of the gene promoter G comprises the bound and unbound states ($[\mathrm{G_T}] = [\mathrm{G}] + [\mathrm{P_nG}]$), we can rearrange the previous equality as:

$$\begin{aligned} k_{-1}^d[\mathrm{P_nG}] &= k_1^d[\mathrm{P}]^n([\mathrm{G_T}] - [\mathrm{P_nG}]) = k_1^d[\mathrm{P}]^n[\mathrm{G_T}] - k_1^d[\mathrm{P}]^n[\mathrm{P_nG}] \\ k_1^d[\mathrm{P}]^n[\mathrm{G_T}] &= k_{-1}^d[\mathrm{P_nG}] + k_1^d[\mathrm{P}]^n[\mathrm{P_nG}] = [\mathrm{P_nG}](k_{-1}^d + k_1^d[\mathrm{P}]^n) \\ [\mathrm{P_nG}] &= k_1^d[\mathrm{G_T}]\frac{[\mathrm{P}]^n}{k_{-1}^d + k_1^d[\mathrm{P}]^n} \end{aligned} \tag{2.6}$$

Hence, to obtain the general Hill activation function, we can rearrange the equation describing $[\mathrm{P}_n\mathrm{G}]$ as:

$$\begin{aligned} [\mathrm{P_nG}] = k_1^d[\mathrm{G_T}]\frac{[\mathrm{P}]^n}{k_{-1}^d + k_1^d[\mathrm{P}]^n}\frac{(k_1^d)^{-1}}{(k_1^d)^{-1}} = [\mathrm{G_T}]\frac{[\mathrm{P}]^n}{\frac{k_{-1}^d}{k_1^d} + [\mathrm{P}]^n} = \\ = [\mathrm{G_T}]\frac{[\mathrm{P}]^n}{K_d + [\mathrm{P}]^n} = [\mathrm{G_T}]\frac{[\mathrm{P}]^n}{K_M^n + [\mathrm{P}]^n} \end{aligned} \tag{2.7}$$

where the dissociation constant K_d is defined as $K_d = k_{-1}^d/k_1^d$ and the half occupancy activation coefficient K_M is defined as $K_M = \sqrt[n]{K_d}$ [227]. Then we can rearrange equations 2.3 as:

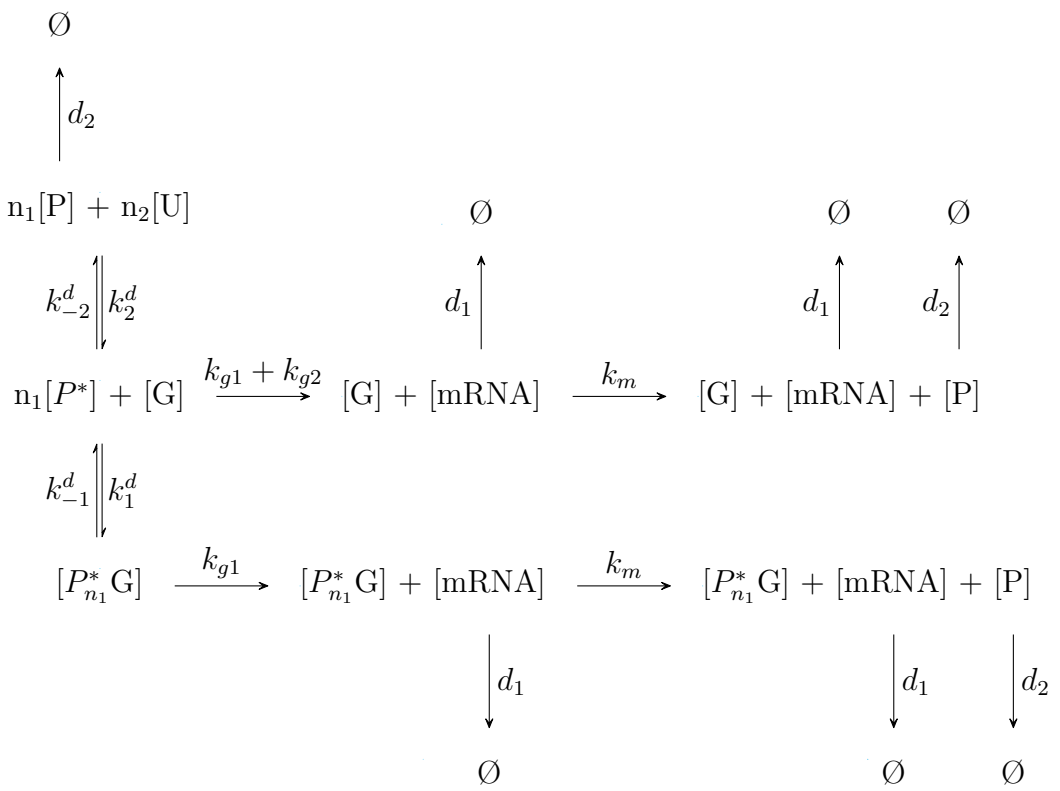
$$\begin{aligned} \frac{d[\mathrm{mRNA}]}{dt} &= \alpha + V_m\frac{[\mathrm{P}]^n}{K_M^n + [\mathrm{P}]^n} - d_1[\mathrm{mRNA}] \\ \frac{d[\mathrm{P}]}{dt} &= k_m[\mathrm{mRNA}] - d_2[\mathrm{P}] \end{aligned} \tag{2.8}$$

where similarly as with the promoter leakiness, the maximum induced transcription rate $V_m = k_{g1}[\mathrm{G_T}]$. This mathematical model describes the mRNA and protein P dynamics for the defined self-inducible promoter following positive

feedback. Fig. 2.6a shows an example for the evolution of the model states over time, showing how the basal transcription and the positive feedback allows the system to evolve to its steady-state independently of the initial conditions. Bandiera *et al.* 2018 [86] followed a similar derivation to obtain their reduced mathematical model describing the synthetic orthogonal inducible promoted designed by Gnügge *et al.* 2016 [228]. This model (Section 3.2.1.2, p. 69 of Chapter 3) is further used in this thesis to study the informative content of experiments for parameter identification and metrics to assess this information *a priori* (Chapter 4).

2.3.5 Derivation of a Model Describing a Genetic System with Negative Feedback and External Induction

Using the same approach as the previous example, we can define a system presenting negative feedback as:



where transcription only occurs when the gene promoter G is free from n_1 proteins P. To add a layer of complexity to the system, we consider that protein P only binds to the gene G promoter when it is in the active form P^*. The reaction occurs with binding and dissociation constants k_1 and k_{-1}, respectively. For

protein P to become activated (P^*), n_2 molecules of an external inducer U (e.g. galactose, aTc or IPTG) need to bound to the protein P, happening at a rate k_2^d. This binding is reversible, with a dissociation rate k_{-2}^d. mRNA transcription occurs at a rate k_{g2} in absence of the active repressor, while basal transcription at a rate k_{g1} in its presence (promoter leakiness). mRNA degradation occurs at rate d_1. Protein P gets translated at rate k_m and degraded at rate d_2. Note that for representation simplicity purposes, I excluded the inducer U and protein P binding/unbinding reactions in the translation results (right side), the degradation of P^* and the association/dissociation of the complex $[P_{n_1}^*\mathrm{G}]$ in the transcription and translation results (middle and right), but these should also be considered there. Fig. 2.5b shows a schematic representation for this hypothetical system. Now, we can describe the mRNA and protein P dynamics as:

$$\begin{aligned} \frac{d[\mathrm{mRNA}]}{dt} &= k_{g1}[\mathrm{G_T}] + k_{g2}[\mathrm{G}] - d_1[\mathrm{mRNA}] = \alpha + k_{g1}[\mathrm{G}] - d_1[\mathrm{mRNA}] \\ \frac{d[\mathrm{P_T}]}{dt} &= k_m[\mathrm{mRNA}] - d_2[\mathrm{P_T}] \end{aligned} \tag{2.9}$$

In this case, as the ODE for protein P we describe the dynamics of the total amount of protein (bound or unbound to inducer U), produced at rate k_m (translation) and degraded at rate d_2. We derived the term $k_{g1}[\mathrm{G_T}]$ in the same manner as for the positive feedback system knowing that $\mathrm{G_T} = \mathrm{G} + P_{n_1}^*\mathrm{G}$. Now, following the same quasi-steady-state approximation as for the positive feedback system for the association/dissociation of $[P_{n_1}^*\mathrm{G}]$, the term [G] can be rearranged as:

$$\begin{aligned} \frac{d[P_{n_1}^*\mathrm{G}]}{dt} &= k_1^d[\mathrm{G}][P^*]^{n_1} - k_{-1}^d[P_{n_1}^*\mathrm{G}] = 0 \\ k_1^d[\mathrm{G}][P^*]^{n_1} &= k_{-1}^d[P_{n_1}^*\mathrm{G}] \\ k_1^d[\mathrm{G}][P^*]^{n_1} &= k_{-1}^d([\mathrm{G_T}] - [\mathrm{G}]) \\ k_{-1}^d[\mathrm{G_T}] &= k_1^d[\mathrm{G}][P^*]^{n_1} + k_{-1}^d[\mathrm{G}] = [\mathrm{G}](k_1^d[P^*]^{n_1} + k_{-1}^d) \\ [\mathrm{G}] &= \frac{k_{-1}^d[\mathrm{G_T}]}{k_{-1}^d + k_1^d[P^*]^{n_1}} = \frac{k_{-1}^d[\mathrm{G_T}]}{k_{-1}^d + k_1^d[P^*]^{n_1}} \frac{(k_{-1}^d)^{-1}}{(k_{-1}^d)^{-1}} = [\mathrm{G_T}]\frac{1}{1 + \frac{k_1^d}{k_{-1}^d}[P^*]^{n_1}} \end{aligned} \tag{2.10}$$

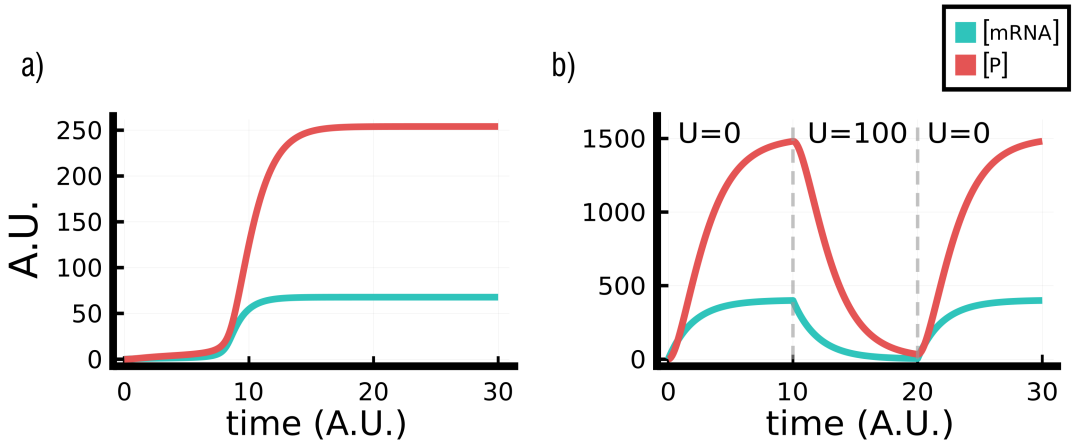


Figure 2.6: Dynamics of gene regulatory networks presenting positive and negative feedback. mRNA and protein P concentrations time evolution for gene regulatory networks with positive (a) and negative (b) feedback described by equations 2.8 and 2.14, respectively. For the system presenting negative feedback (b) an external inducer U activates the repressor activity. To exemplify the situation, I divided the total time in three sections where the inducer U is set to be absent (0) or present at an arbitrary concentration value (100). The parameter values selected for the positive feedback example (a) simulation are: $\alpha = 1$, $V_m = 60$, $K_M = 20$, $n_1 = 3$, $k_m = 3$, $d_1 = 0.9$, $d_2 = 0.8$. The parameter values selected for the negative feedback example (b) simulation are: $\alpha = 1$, $V_m = 200$, $K_M = 20$, $n_1 = 3$, $k_m = 3$, $d_1 = 0.5$, $d_2 = 0.8$, $n_2 = 2$, $K_{M2} = 10$. In both instances, the initial condition for both model states (y_0) was set to 0.

We can substitute the term in the ODE describing mRNA dynamics as:

$$\frac{d[\text{mRNA}]}{dt} = \alpha + k_{g2}[\text{G}_\text{T}]\frac{1}{1 + \frac{k_1^d}{k_{-1}^d}[P^*]^{n_1}} - d_1[\text{mRNA}] \tag{2.11}$$

and rearrange k_1^d and k_{-1}^d so that $K_d = k_{-1}^d/k_1^d$ and $K_M = \sqrt[n_1]{K_d}$:

$$\frac{d[\text{mRNA}]}{dt} = \alpha + V_m\frac{1}{1 + \frac{1}{K_d}[P^*]^{n_1}} - d_1[\text{mRNA}] = \alpha + V_m\frac{1}{1 + (\frac{[P^*]}{K_M})^{n_1}} - d_1[\text{mRNA}] \tag{2.12}$$

where $V_m = k_{g2}[\mathrm{G_T}]$. To handle the term $[P^*]$ (which could also be represented as $[\mathrm{PU_{n_2}}]$) and include the inducer dynamics, we can apply a quasi-steady-state approximation assuming a much faster temporal scale for the binding/unbinding of [U] compared to the rest of the process. The total amount of protein is distributed between inactive protein and unbound and bound active protein ($[\mathrm{P_T}] = [\mathrm{P}] + [P^*] + [P^*_{n1}\mathrm{G}]$). However, we can consider the amount of active protein bound to the gene promoter negligible compared to the amount of free protein, so that $[\mathrm{P_T}] \approx [\mathrm{P}] + P^*$. Now, we can obtain an approximate formulation for $[P^*]$ as:

$$\begin{aligned}
\frac{dP^*}{dt} &= k_2^d[\mathrm{P}][\mathrm{U}]^{n_2} - k_{-2}^d[P^*] = 0 \\
k_2^d[\mathrm{P}][\mathrm{U}]^{n_2} &= k_{-2}^d[P^*] \\
k_{-2}^d[P^*] &= k_2^d[\mathrm{U}]^{n_2}([\mathrm{P_T}] - [P^*]) = k_2^d[\mathrm{U}]^{n_2}[\mathrm{P_T}] - k_2^d[\mathrm{U}]^{n_2}[P^*] \\
k_2^d[\mathrm{U}]^{n_2}[\mathrm{P_T}] &= k_{-2}^d[P^*] + k_2^d[\mathrm{U}]^{n_2}[P^*] = [P^*](k_{-2}^d + k_2^d[\mathrm{U}]^{n_2}) \\
[P^*] &= \frac{k_2^d[\mathrm{U}]^{n_2}[\mathrm{P_T}]}{k_{-2}^d + k_2^d[\mathrm{U}]^{n_2}} = \frac{k_2^d[\mathrm{U}]^{n_2}[\mathrm{P_T}]}{k_{-2}^d + k_2^d[\mathrm{U}]^{n_2}} \frac{(k_2^d)^{-1}}{(k_2^d)^{-1}} = \\
&= [\mathrm{P_T}] \frac{[\mathrm{U}]^{n_2}}{\frac{k_{-2}^d}{k_2^d} + [\mathrm{U}]^{n_2}} = [\mathrm{P_T}] \frac{[\mathrm{U}]^{n_2}}{K_{M2}^{n_2} + [\mathrm{U}]^{n_2}}
\end{aligned} \tag{2.13}$$

Then, we can substitute the term in the ODE defining mRNA dynamics and obtain the final model:

$$\begin{aligned}
\frac{d[\mathrm{mRNA}]}{dt} &= \alpha + \frac{V_m}{1 + (\frac{[\mathrm{P_T}]}{K_M} \frac{[\mathrm{U}]^{n_2}}{(K_{M2}^{n_2} + [\mathrm{U}]^{n_2})})^{n_1}} - d_1[\mathrm{mRNA}] \\
\frac{d[\mathrm{P_T}]}{dt} &= k_m[\mathrm{mRNA}] - d_2[\mathrm{P_T}]
\end{aligned} \tag{2.14}$$

The final model describes the mRNA and protein P dynamics for the defined self-repressing promoter, where the repressor is subjected to external activation. Fig. 2.6b shows an example of the evolution of the model states over time, showing that in the absence of the inducer U the system evolves to a steady-state with high protein and mRNA levels. When inducer U is introduced to the system, it activates the repressing properties of protein P and, with negative feedback, drives the system to a steady-state characterised by low protein and mRNA levels

until U is completely removed. I will use this model to study my strategy to optimally design experiments to improve parameter inference (see Section 7.1.1, p. 186 of Chapter 7). Lugagne *et al.* 2017 [85] followed a similar derivation to obtain a mathematical model describing their genetic toggle switch. This model (Section 3.2.1.1, p. 64 of Chapter 3) is extensively used in this thesis to investigate the problems of parameter inference, quantification of experiment informativeness (Chapter 4) and optimal experimental design for model selection (Chapter 5) and parameter estimation (Chapter 7).

2.3.6 General Model Formulation and Observation Function

The common formulation for all these systems of generally non-linear ODEs describes the vector of state variables $x(t)$ over time t, with $x \in \mathbb{R}^{n_x}$, ([mRNA] and [P] in the previous two examples) as:

$$\dot{x} = \frac{d[x]}{dt} = f(x, \theta, u, t) \tag{2.15}$$

where θ is the vector of model parameters (with $\theta \in \mathbb{R}^{n_\theta}$) and u are externally controlled factors represented as a scalar (with $u \in \mathbb{R}^{n_u}$) or a time depended function [229], [230]. Generally, it is unfeasible to analytically integrate these ODEs, needing the use of initial value problem (IVP) solvers [231]. Such models are related to time discrete experimental data thanks to the observation function y [79] as:

$$y = g(x, \theta, u, t) \tag{2.16}$$

where the experimental measurements in y are generally noise corrupted and discretised per observable (not all model state variables x might be observable in an experiment), sampling time and experiment [57], [79]. For the previous two examples, y would contain the experimental observations for [mRNA] and [P] or just one of them, usually protein as is the case for all systems studied in this

work (see Chapter 7).

2.3.7 Bistability in Biological Systems

Mathematical modelling of gene regulatory networks allows for quantitatively predicting biological systems, but it also aids in revealing crucial properties of others [40], [199]. Bistability, or the presence of multiple stable steady-states for a system (i.e. multistability), is a common property in cell signalling and gene regulatory networks [6]. It plays a crucial role in cell cycle transitions [232], differentiation [233] and fate [234]. Bistable behaviour is also involved in cancer development [235], the lysogeny/lysis[xx] cycle in bacteriophages[xxi] [236], or fast adaptation to environmental changes [237]. Bistable responses have also been replicated in synthetic systems, such as the genetic toggle switch developed by Gardner *et al.* 2000 [9] that set one of the starting points of synthetic biology. This synthetic system and its bistable properties are extensively studied in this thesis (see Chapters 4, 5 and 7). Hysteresis[xxii] (or memory of a system) or all-or-none system responses indicate the potential presence of bistability. Yet, both approaches require access to an adjustable variable in the system, such as an external inducer [234], [238], [239]. Hence, experimental observation of bistability, although possible [16], is a complicated endeavour [239]. Mathematical models aid the detection of such behaviours and the identification of network structures that have the potential to generate bistable responses [6].

Positive feedback loops[xxiii] in biological processes are one of the necessary motifs to generate bistability [239]. Nonetheless, bistability only arises in positive feedback loops under a specific set of conditions [6]. First, the system must present ultrasensitivity or some non-linearity (e.g. cooperativity as represented by the Hill function) which we can investigate thanks to mathematical models derived from the law of mass action [240]. For example, the positively self-regulatory gene network described by equations 2.8 presents a positive feedback loop. However, it can only generate a bistable behaviour if the hill coefficient n is greater than one (i.e. cooperativity). If a system meets the two

[xx] Lysogeny involves the viral infection of a bacteria integrating the virus genes into the host for replication. Lysis involves the exit of viral particles form the cell usually by rupturing the cell membrane or wall.

[xxi] Virus that infects bacteria.

[xxii] Dependence of a state from a system on its past history.

[xxiii] The output of a process induces an increase of the same process.

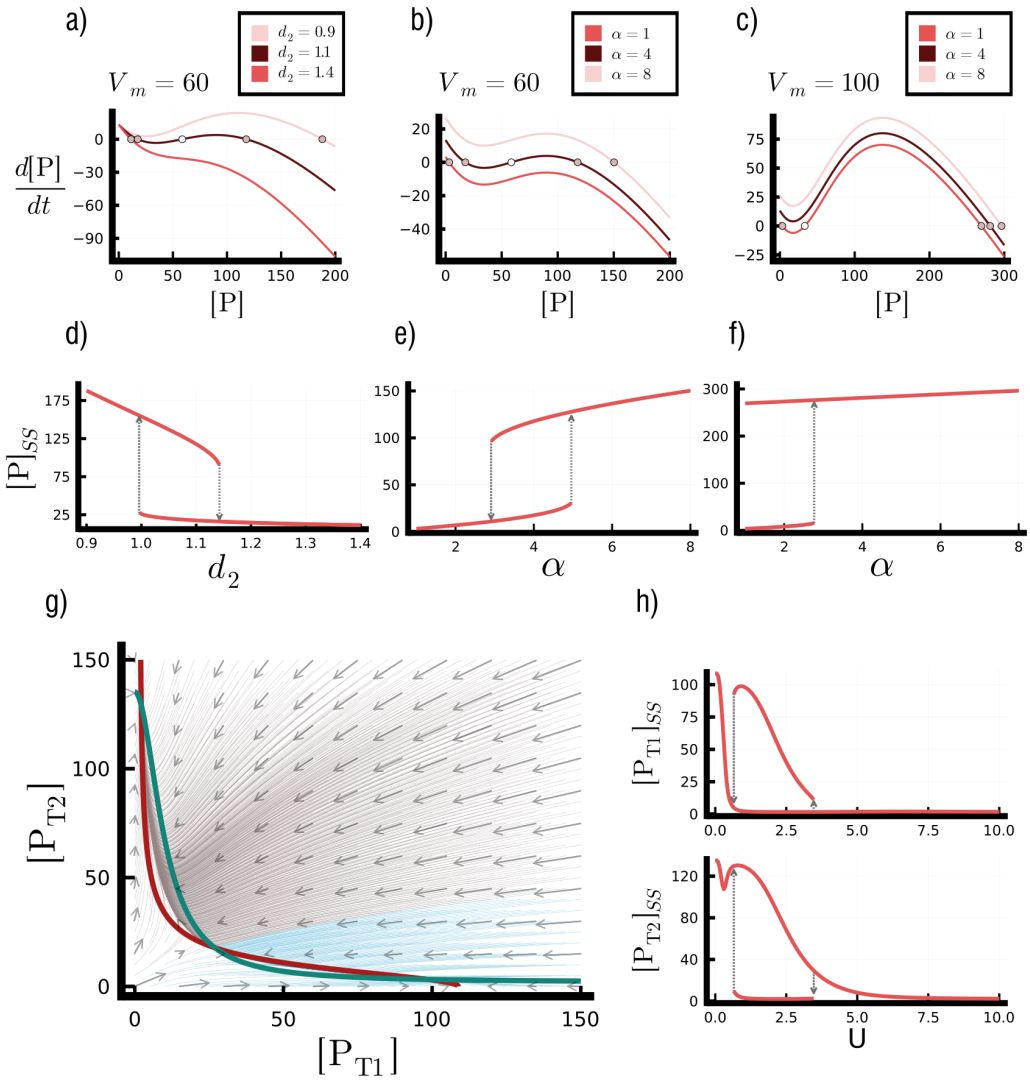


Figure 2.7: Stability analysis of mathematical models. Steady-state analysis (a-c) and bifurcation diagrams (d-f) of a mathematical model for a gene regulatory network with a positive feedback loop dependent on the degradation rate (a, d) and basal transcription rate at moderate (b, e) and high (c, f) maximal transcriptional rate. The nominal parameter values used (set arbitrarily) were $\alpha = 4, V_m = 60, K_M = 100, n = 2, k_m = 3, d_1 = 0.9, d_2 = 1.1$. g) Nullclines for a system containing a double-negative feedback loop showing the three line intersections for the two stable ($[P_{T1}] \approx 2.1, [P_{T2}] \approx 130$ in gray and $[P_{T1}] \approx 97.5, [P_{T2}] \approx 3.4$ in blue) and unstable ($[P_{T1}] \approx 36.7, [P_{T2}] \approx 18$) steady-states. h) Bifurcation diagrams for the system dependent on the external inducer U (bistability when U is between ≈ 0.65 and ≈ 3.47). The nominal parameter values used (set arbitrarily) were $\alpha_a = \alpha_b = 1, V_{ma} = V_{mb} = 70, K_{Ma} = K_{Mb} = K_{M2a} = K_{M2b} = 10, n_{1a} = n_{1b} = n_{2a} = n_{2b} = 2, k_{ma} = k_{mb} = 8, d_{1a} = 0.9, d_{1b} = 1.4, k_{2a} = 5.8, k_{2b} = 3$.

stated conditions, the presence of bistability will depend on the rate at which some reactions occur (i.e. production and degradation rates) [240], [241].

To illustrate the emergence of bistability dependent on reaction rates, we can use the model described by equations 2.8. We can assess the number of steady-states (times $\frac{d[\mathrm{P}]}{dt} = 0$) present at different concentrations of the self-regulatory protein P (Fig. 2.7 a,b,c). If protein degradation is far stronger than its production ($d_2 = 1.4$), the protein P levels in the cell will remain low. If the degradation is weaker ($d_2 = 0.9$), the protein P levels in the cell will remain high. At an intermediate point between these two cases ($d_2 = 1.1$), we can observe three different steady-states; two stable (one at low and one at high [P]) and an unstable[xxiv] (in between the stable ones at the bifurcation point) one (Fig. 2.7a) [239]. The same phenomenon would occur at changing values for the protein P production [7]. A high basal mRNA expression ($\alpha = 8$) leads to high levels of protein P and low basal expression ($\alpha = 1$) to low levels, while an intermediate level ($\alpha = 4$) generates bistability (Fig. 2.7b). By modifying the maximal transcriptional rate (V_m), we can obtain a scenario where the system only presents bistability or monostability at high concentrations of protein P (Fig. 2.7c). We can assess the potential for bistability by analysing the presence or absence of hysteresis (Fig. 2.7 d,e,f) [6]. At values for the assessed parameters (d_2 and α) where the steady-state depends on the history of the system (d_2 between ≈ 0.996 and ≈ 1.142 in panel d, α between ≈ 2.91 and ≈ 4.96 in panel e and α between 1 and $\approx$ 2.77 panel f of Fig. 2.7), this presents bistability [232]. Hysteresis analysis using bifurcation diagrams[xxv] also reveals the presence of reversible (Fig. 2.7 d,e) and irreversible (Fig. 2.7f) transitions between the stable steady-states [242].

Double negative feedback loops[xxvi] in biological processes (indirect positive feedback loop) are another common biological motive generating bistability [241]. For example, while the negatively self-regulatory gene network[xxvii] described by equations 2.14 is unable to produce bistability[xxviii], if we modify it to become two

[xxiv] Bistable systems present two stable and one unstable steady-states

[xxv] Diagram or plot that shows the values of a state variable as a function of a parameter that can cause a bifurcation on the state.

[xxvi] The output of a process induces a decrease of the same process.

[xxvii] While positive feedback loops can generate bistability, negative feedback loops can arise oscillatory responses.

[xxviii] Two drastically different steady-states can be reached in absence or presence of the inducer U, but the two do not happen at the same value of U.

mutually repressing proteins, as done for genetic toggle switches [85] studied in this thesis (Section 3.2.1.1, p. 64 of Chapter 3), this can lead to a bistable behaviour. Hence, we can define the rate equations for two mutually repressing proteins P_{T1} and P_{T2} as a set of coupled non-linear ordinary differential equations of the form:

$$\begin{aligned}\frac{d[\mathrm{P_{T1}}]}{dt} &= k_{ma}((\alpha_a + \frac{V_{ma}}{1+(\frac{[\mathrm{P_{T2}}]}{K_{Ma}}\frac{[\mathrm{U}]^{n_{2a}}}{K_{M2a}^{n_{2a}}+[\mathrm{U}]^{n_{2a}}})^{n_{1a}}})\frac{1}{d_{1a}}) - d_{2a}[\mathrm{P_{T1}}] \\ \frac{d[\mathrm{P_{T2}}]}{dt} &= k_{mb}((\alpha_b + \frac{V_{mb}}{1+(\frac{[\mathrm{P_{T1}}]}{K_{Mb}}\frac{[\mathrm{U}]^{n_{2b}}}{K_{M2b}^{n_{2b}}+[\mathrm{U}]^{n_{2b}}})^{n_{1b}}})\frac{1}{d_{1b}}) - d_{2b}[\mathrm{P_{T2}}]\end{aligned} \tag{2.17}$$

where, for simplicity purposes, we assume that mRNA dynamics are at steady-state. Fig. 2.5c shows a schematic representation for this hypothetical system. Note that this model is closer in structure to the genetic toggle switch model from Lugagne *et al.* 2017 [85] and extensively used in this thesis. See Section 3.2.1.1 (p. 64) of Chapter 3 for details of the model. Depending on the different parameter values, the system can present bistability, which we can analyse by representing the nullclines[xxix] of the system (Fig. 2.7g) [6]. The presence of three intersections (two stable and one unstable steady-states) indicates bistability, where the steady-state of the system will depend on its initial state. In this example, we can modulate the external inducer U (better reproducible experimentally) to transition the system between bistable and monostable behaviours, which we can assess by analysing the hysteresis of the system (Fig. 2.7h). Hence, mathematical models are a powerful tool for studying the properties of biological systems. Yet, the previous two examples strengthen the importance of accurate model parameter estimates.

[xxix] Curves on a phase plane so that the differential equations describing the state variables equal to zero (steady-state).

2.4 Parameter Identification and Model Selection

2.4.1 Problems in Model Development

The usefulness of mathematical models describing biological systems depends on how faithfully these represent the system (usually based on hypotheses) and on having accurate estimates for the model parameters [243]. If multiple hypotheses can potentially describe an unknown biological process, this should result in the formalisation of different model structures. In these scenarios, model selection aims at identifying the one that better describes the system using experimental data to back up the decision [54] as shown in Chapter 5 of this thesis. Even in these situations, researchers need accurate parameter estimates that allow a model to be predictive and generalisable [244]. In most cases, experimentally measuring different model parameters is a consuming, daunting or even unfeasible task, especially for parameters without direct biological interpretation derived from model approximations or restructures [51], [245], [246] as shown in Chapter 4 of this thesis. Hence, the best approach is to infer them from noisy experimental data [51]. Yet, the solution to this problem is far from mundane.

2.4.2 The Problem of Parameter Estimation

Parameter estimation is usually a non-linear optimisation problem [229], which due to its habitual non-convexity, needs global optimisation solvers[xxx] [247] or adequate probabilistic sampling algorithms[xxxi] [248]. Nonetheless, the high interest in the topic has generated extensive literature demonstrating how to solve the problem [51], [229], [249], [250], [251], different metaheuristic and sampling optimisation algorithms suited for the task [252], [253], [254], [255] and multiple computational tools [74], [256], [257], [258] such as the one developed and described in Chapter 8. The common approach to parameter estimation involves defining an objective function that quantifies the distance between experimental data and model predictions to find, thanks to an optimiser algorithm, a parameter set that minimises such distance [259].

[xxx] Algorithms designed to efficiently find global optimum solutions without getting stuck in local ones.

[xxxi] These algorithms are used in Bayesian inference to explore probability distributions

Frequentist and Bayesian methods are the two main statistical approaches to infer model parameters (Fig. 2.8). Each strategy comes with a set of practical and theoretical advantages and disadvantages that one should consider ahead of use [260]. However, while in some scenarios, it is difficult to assess which is the most suitable approach, both play a crucial role in the identification of model parameters [261].

2.4.3 Frequentist Approach to Parameter Estimation

The frequentist approach relies on the belief that parameters are fixed constants that can be uncovered from a frequent sampling of the process [262] where uncertainty comes from randomness from the system (intrinsic) or measurements (extrinsic) [52]. Under this assumption, the general approach to infer parameters from models describing biological processes is to set an objective function to optimise. This function computes the discrepancy between experimental data and model predictions under different parameter values [51]. Often, the error functions selected for the task are a form of the least squares that accounts for the experimental noise associated with the data [229] or the maximum (log-) likelihood under general normality assumptions [263]. Then, the process goal is to find the parameter set that better fits the experimental data, usually with some confidence (i.e. confidence intervals[xxxii]) associated with this point estimate. Such confidence intervals can be computed using the Fisher information matrix (FIM) that, thanks to the Cramér-Rao inequality, provides a lower bound for the covariance matrix (see Section 3.2.5 p. 84 of Chapter 3 for more details) [79], [264], [265]. Alternatively, Monte Carlo sampling and bootstrap methods give a more robust, yet computationally more expensive, approximation [266].

While frequentist parameter estimation is widely used due to its practicality, easier implementation and lower computational complexity compared to Bayesian methods, this presents a series of disadvantages [52]. Such strategy sees a decrease in performance in the presence of high experimental noise [51], usually need to rely on normality assumptions for experimental noise and parameter

[xxxii] Region of the parameter space that has a given probability (usually 95 % or 99 %) to contain the true value for the parameter

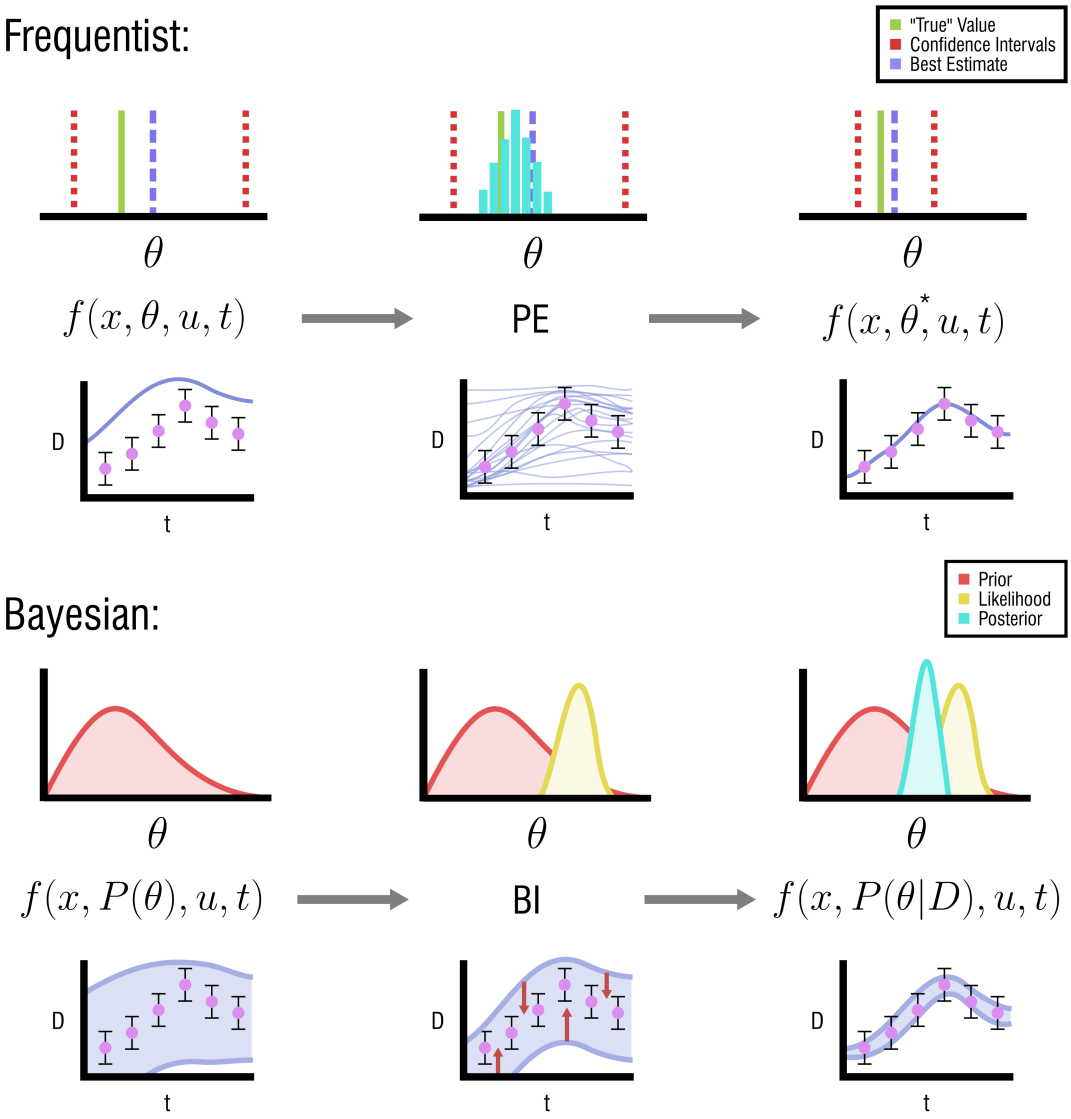


Figure 2.8: Comparison between frequentist and Bayesian parameter inference strategies. Frequentist parameter estimation relies on finding the "true" parameter value underneath a process after frequent sampling with an associated error, usually represented by confidence intervals. Bayesian inference updates prior beliefs portrayed as parameter probabilities following the Bayes' theorem to obtain posterior distributions based on priors and their likelihood after observing some new experimental data.

uncertainties [267], [268] and results are generally not conditioned by observed experimental data [52].

2.4.4 Bayesian Approach to Parameter Estimation

Bayesian strategies consider parameters (or hypotheses) as random variables with probabilities associated with them updated after observing new data following the Bayes' theorem [262]. *A priori* knowledge or beliefs about parameters Y are encoded in the form of probability distributions, introducing the prior distribution $P(Y)$. In the light of new experimental data X, the likelihood $P(X|Y)$ (probability that X is true given Y) of the prior distribution and the probability of the data $P(X)$ (marginal probability or model evidence) are assessed to obtain the newly informed posterior distributions $P(Y|X)$ [269] following:

$$P(Y|X) = \frac{P(X|Y)\ P(Y)}{P(X)} \tag{2.18}$$

This approach easily removes normality assumptions on experimental noise and model parameters and the need for approximating confidence intervals for parameter point estimates [52]. Also, this probabilistic approach is more robust against high experimental noise [51] and parameter probabilities are only updated conditioned on observed experimental data [52]. The drawback of this approach highlighted by frequentist statisticians is that, in the absence of prior knowledge of the parameters, different researchers might assign different prior probabilities, which can condition the posterior probabilities obtained [270], [271].

Regardless of the multiple advantages of Bayesian strategies, these have been less used to infer model parameters when compared to frequentist ones due to their high computational complexity [51]. The Bayes' theorem is intractable analytically for the usual non-linear models used to describe biological systems [272]. However, over the last years, Bayesian inference has grown in popularity amongst the scientific community thanks to advances in probabilistic samplers such as Markov chain Monte Carlo (MCMC) methods [273]. These algorithms allow approximations of posterior probabilities by evaluating samples

of the potential target distribution, which is proportional to the likelihood of samples drawn from prior probabilities [274]. Both strategies are further investigated in Chapter 4.

2.4.5 The Problem of Model Selection

In the case of multiple model structures developed when the underlying system structure is still unknown or if multiple parameterisations when a model is inferred using different sets of data, researchers face the tedious model selection problem which highly depends on available experimental data [51], [54]. For both scenarios, success on this problem will still depend on adequate parameter identification, independently from the inference scheme used [275]. The aim is to be able to identify the model with better predictive capabilities over different data sets while minimising as much as possible its complexity [53]. For example, if we consider the system containing a positive feedback loop described by equations 2.8, we might have insufficient information to know if the protein binding involves cooperativity. In this scenario, we might start from two different model structures, one considering cooperativity (hill coefficient $n > 1$) and one without (no hill coefficient or $n = 1$). We can then use model selection to inform the choice and learn new information about the unknown system thanks to accurate parameter inference and adequate experimental data (Fig. 2.9).

There are two main approaches to target the model selection problem, likelihood-based or Bayesian ones [54]. Likelihood-based methods, extensively used in frequentist statistics, evaluate models using the best point-estimate for the model parameters applying a form of the likelihood function plus a penalisation term for the model complexity (usually measured as in the number of degrees of freedom) [276]. To address model complexity, Ockham's razor plays a central role in most metrics used for model selection penalising overly-complex models unable to provide a significant advantage compared to simpler ones [54]. The most notable examples include the Akaike information criterion (AIC) and the Bayesian information criterion (BIC). In the AIC, the penalisation term is 2 times the number of parameters. In the BIC, the penalisation term is the number of parameters times the logarithm of the number of data points.

Bayesian approaches focus on using the model evidence or marginal likelihood ($P(X)$ in equation 2.18) to rank models since this provides the probability of the

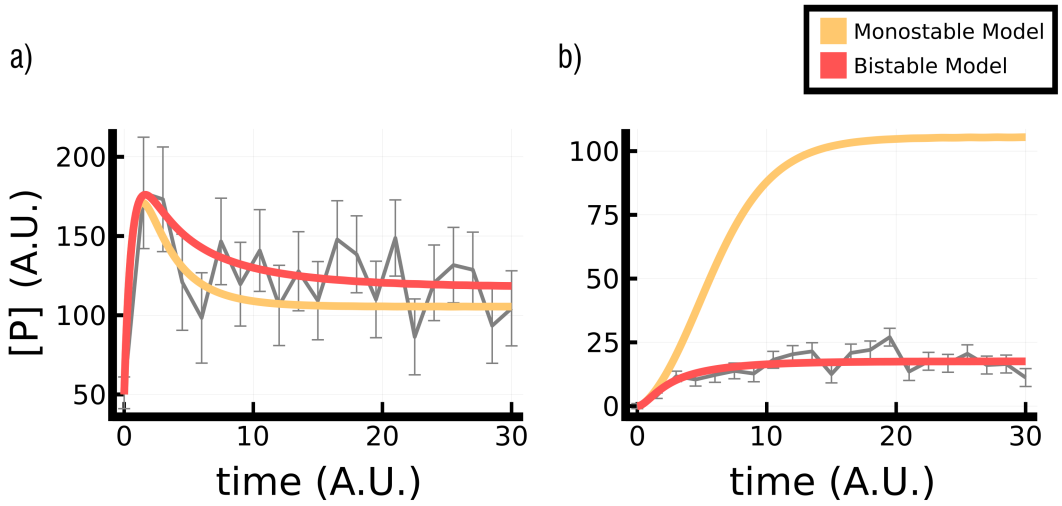


Figure 2.9: Simplified exemplification of the model selection problem for biological processes. Without an in-depth knowledge of a system, different model formalisation can arise, needing adequate experimental data to discriminate between these. Model predictions for the model containing a positive feedback loop described in equations 2.8 considering non-cooperative (yellow) and cooperative (red) binding of the protein to the promoter against pseudo-data (grey) generated (20 % heteroscedastic noise normally observed in microfluidic experiments) by the ultrasensitive model. The initial conditions for the system were set to high mRNA (110 A.U.) and moderate protein P (50 A.U.) to drive the two models to a similar steady-state (a) or to minimum mRNA (0 A.U.) and minimum protein P (0 A.U.) to drive the two models to opposite steady-states (b). The nominal parameter values used (set arbitrarily) for both models were $\alpha = 4, V_m = 60, K_M = 100, k_m = 3, d_1 = 0.9, d_2 = 1.1$. n was set to 1 for the model considering non-cooperativity and to 2 for the model considering cooperativity.

data observed given a specific model M_i ($P(X|M_i)$) [277]. In this case, we include the model M_i in the notation of each probability defined by Bayes' theorem as:

$$P(Y|X, M_i) = \frac{P(X|Y, M_i)\ P(Y|M_i)}{P(X|M_i)} = \frac{P(X|Y, M_i)\ P(Y|M_i)}{\int_\theta P(X|Y, M_i)P(Y|M_i)} \tag{2.19}$$

The model evidence is usually obviated in Bayesian inference algorithms, where $P(Y|X) \propto P(X|Y)P(Y)$, since the term acts as a normalisation factor to ensure that the posterior probability integrates to one. Most sampling algorithms

exclude its computation because it is often analytically unfeasible to calculate, particularly in high-dimensional cases [274]. Nonetheless, there are different approximations with diverse levels of complexity and accuracy, such as the Laplace method [269] or the more accurate but computationally expensive use of power posteriors [276], [278]. Although traditionally shunned due to its difficulty, model selection is recently gaining momentum with increasing examples where the strategy provides a better knowledge of biological systems [279], [280], [281], [282], [283], [284], [285]. This strategies are further investigated in Section 5.1.1 (p. 141) of Chapter 5 of this thesis.

2.4.6 Parameter Identifiability and Sensitivity Analyses

After formalising a mathematical model but before proceeding with parameter inference (frequentist or Bayesian), it is regarded as a good practice to assess the feasibility of the process to avoid potential issues down the line. The examination is commonly done by performing identifiability (structural and practical) and sensitivity analyses [57], [229].

Structural identifiability assesses the possibility of identifying unique values for the model parameters assuming perfect conditions, hence, dependent on the model structure [286]. If a model is structurally unidentifiable, it means that, for at least one model parameter, the parameter estimation problem using continuous and noise-free data has infinite solutions without altering the quality of the model predictions [287]. The opposite and desirable scenario is structural identifiability, where parameter values can, in theory, be identified using experimental data. Two categories arise when models are structurally identifiable; there can be multiple but finite solutions to the parameter estimation problem (structurally locally identifiable) or a potentially unique solution (structurally globally identifiable) [288]. Despite the importance of structural identifiability in the modelling process, the task is commonly disregarded due to its high complexity, particularly with large non-linear systems that make symbolic operations computationally unfeasible [289]. Nonetheless, since the analysis is recently gaining interest in the research community, new methods are being developed to assess the structural identifiability of dynamic biological models. Some examples involve Taylor series and generating series approach [229],

generalised observability concepts [288], Lie derivatives[xxxiii] [290], semi-numerical approaches [291], or the use of experimental data [289]. More recent work focuses on decreasing the computational burden of assessing structural identifiability [292] or extending the examination by including multi-experimental analysis [293].

Once a model is established as structurally identifiable, the next step in the modelling process is to assess for practical identifiability. Practical identifiability evaluates the possibility of identifying model parameters given experimental data. The importance of this analysis is related to the fact that, in real scenarios, experimental data might be insufficient or uninformative for the task [289]. Practical identifiability is usually assessed using confidence intervals of parameter estimates, where one of the intervals becomes infinite or with a value far outside the initial bound [286]. Consequently, adding additional informative measurements can potentially make the model practically identifiable [289]. For a more accurate examination of parameter uncertainties (and correlations), Monte Carlo based approaches can be used where the parameter estimation problem is solved multiple times for a given set of experimental data. This approach, however, is computationally more expensive than estimating confidence intervals [57]. In Bayesian inference, practical unidentifiability is usually associated with highly uncertain posteriors, multimodality or non-converging chains in MCMC methods [294].

Finally, sensitivity analysis helps identify the degree of contribution of each model parameter to the output of the system [295]. A ranking of the parameters highlights the effect that small changes in parameter values have in the simulation of the different model states [229]. Sensitivity analysis using a given parameter vector (local sensitivity analysis) and the partial derivative of the model output with respect to each parameter might be of limited use in non-linear models with highly correlated parameters [296]. A common approach is to repeat the process for a large number of randomly sampled parameter vectors (global sensitivity analysis) [297]. In a Bayesian framework, parameter vectors are extracted from prior distributions, including correlations between parameters and areas in the

[xxxiii] In the presence of a flow described by a vector field, the Lie derivative is used to assess the changes of a second vector field. A vector is a geometric object described by its length (i.e. magnitude) and direction. A tensor is a generalised form of a vector that can be defined in an n-dimensional space, where a scalar is a tensor of rank 0 and a vector a tensor of rank 1.

parameter space with higher probabilities than others [298]. This approach results in a more robust analysis where modellers can rank the different parameters according to their importance. Highly-sensitive parameters tend to be more identifiable since small parameter variations will lead to big output changes. At the same time, lowly-sensitive parameters will tend to be less identifiable for the same reason. Such analysis also aids in the fixing of model parameters to reduce the complexity of the parameter estimation problem [299], [300].

2.5 Optimal Experimental Design

2.5.1 Optimal Experimental Design in Biology

Optimal experimental design (OED) encompasses a series of statistical tools to design maximally informative experimental schemes for a particular task (e.g. parameter identification or model selection), potentially reducing the number of experiments needed [61], [301]. Contrary to a traditional approach to experimental design (e.g. one-factor-at-a-time method or random modulation of factors), OED attempts to assess the informativeness of potential experiments to find the most suitable one for the task, potentially saving crucial time and resources [55], [75]. Biological sciences can highly benefit from such approaches due to the common laborious and expensive nature of experimentation and the noisiness of the generated data [302], [303], [304]. Instead of relying on steady-state experimental data or intuition-driven dynamic experimental designs, OED has the potential to generate highly informative data with less resource allocation [86], [305]. Yet, while many scientific disciplines are already taken full advantage of OED [67], [68], [69], [301], biological sciences are still in the dawn of the process. While the statistical design of experiments is expanding across biology [55], OED for complex problems (e.g. parameter estimation and model selection) is still mainly applied in theoretical or computational studies despite the maturity and potential of the process [57], [75], [306], [307]. A substantial portion of the work described in this thesis focuses on the use and direct application of OED for the problems of model selection (Chapter 5) and parameter inference (Chapter 7) and provide an easy-to-use platform (BOMBs.jl described in Chapter 8) for the task.

2.5.2 Frequentist Approach to Optimal Experimental Design for Parameter Estimation

In systems and synthetic biology, OED studies focus on designing experiments to better infer model parameters from traditionally noisy and scarce experimental data [57], [75], [86], [308]. The most common approach to assess the informative content of experiments regarding parameter identification is to use the FIM [57]. This frequentist approach allows computing, from a parameter point estimate, the lower bound for the covariance matrix as done to estimate confidence intervals

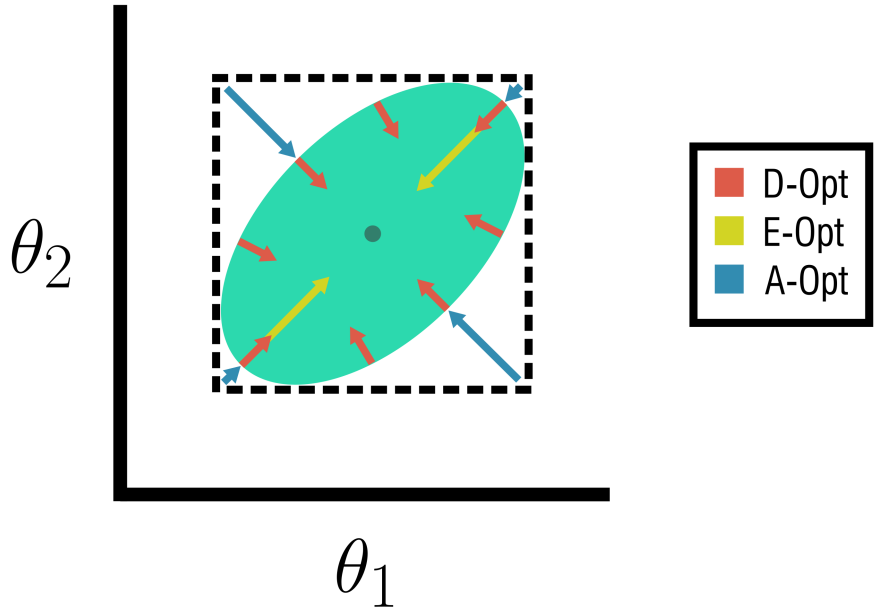


Figure 2.10: Optimality criteria effect on the parameter confidence ellipsoid. Geometric representation of the bivariate confidence ellipsoid for the example parameters θ_1 and θ_2 (green) and the objective of the D-, E- and A- optimality criteria (arrows). Arrows indicate the type of volume reduction of the parameter confidence ellipsoid regarding a specific metric. D-opt focuses on reducing the overall ellipsoid volume, E-opt on reducing the ellipsoid volume with respect to the most uncertain parameter, and A-opt focuses on reducing the average volume of the ellipsoid.

(see Section 3.2.5 p. 84 of Chapter 3 for more details). Hence, computing the FIM and consequently the covariance matrix using the Cramér-Rao inequality for possible experimental designs, provides information about how much these designs have the potential to reduce parameter variances [301]. Different scalar metrics of the FIM summarise the information enclosed in the matrix allowing its integration into optimisation schemes [308]. D- and E- optimality criteria (D-opt and E-opt) are the most employed across literature, with A-optimality (A-opt) and others being less in use across systems biology [57] (Fig. 2.10).

The D-opt criterion maximises the determinant of the FIM, seeking to minimise the volume of the confidence region around the feasible parameter space. The D-opt criterion is the most popular one due to improving the confidence region of all parameters and its robustness when using different parameterisations of the model [309]. The E-opt criterion maximises the minimum eigenvalue of the FIM,

intending to reduce the maximum variance of the parameter set since the covariance matrix is related to the inverse of the FIM. Hence, the E-opt criterion distributes the information of an experiment across parameters in a more direct manner [57]. The A-opt maximises the trace of the FIM, attempting to minimise the average variance of the parameters. This criterion is less used due to potential unreliability in cases with high correlations between parameters [301]. Alternatively, some less used criteria focus on minimising the variance of the model predictions (e.g. I-opt or G-opt) [310]. The criterion of choice is then coupled into an optimisation algorithm to find the optimum experimental scheme. Due to the usual non-convexity of this problem when applied to the non-linear models describing biological systems, global or meta-heuristic optimisation techniques are required [308].

2.5.3 Bayesian Approach to Optimal Experimental Design for Parameter Inference

Strategies for Bayesian OED for parameter identification have been mostly restricted to theoretical definitions due to the high computational complexity of the task, with little progress since the 1990s [78], [82]. Some computationally tractable formulations of Bayesian OED use the frequentist optimality criteria (D-opt, E-opt, A-opt) averaged over the prior knowledge of the parameters (pseudo-Bayesian OED) [311]. Fully Bayesian approaches attempt to evaluate *a priori* the gain in information of the parameter posteriors given an experimental design e_d, usually encoded by a utility function to maximise. This utility function is commonly defined as the expected information gain, hence the expectation of the posterior entropy reduction with respect to the prior entropy [312]:

$$U(e_d) = E_{P(D|e_d)}[H(\theta) - H(\theta|D)] \tag{2.20}$$

where $H(X) = -\int p(x)\log(p(x))dx = E[-\log(p(X))]$. This expected information gain is usually computed as the Kullback–Leibler divergence between prior and posterior distributions [82]:

$$U(e_d) = E_{\theta|D,e_d}[\log(P(\theta|D,e_d)) - \log(P(\theta))] = \int_{\Theta} P(\theta|D,e_d)\ \log(P(D|e_d,\theta))\ d\theta - \log(P(D|e_d)) \tag{2.21}$$

A similar approach is to use as a utility function the mutual information, which indicates how much information from the parameters can be gained from data [56]:

$$U(e_d) = I(\theta, D) = \int_{\Theta}\int_{D} P(\theta, D|e_d)\ \log(\frac{P(\theta, D|e_d)}{P(\theta)P(D|e_d)})\ d\theta\ dD \tag{2.22}$$

with $P(\theta, D|e_d)$ being the joint probability distribution between the parameters and the data given an experimental design.

While these Bayesian utility functions for parameter inference were proposed decades ago [78], their applicability is still hindered due to their computational complexity [82], [312]. Such utility functions for OED are only applied in simple scenarios (e.g. using linear or Gaussian models), uncommon in biological contexts [82], [313], [314]. The problem arises from the accurate approximation of the multidimensional integrals and model evidence $P(D|e_d)$ [315]. Some approximations have been recently proposed to estimate the mutual information utility function in biological contexts [56], [316]. However, their high computational complexity occasionally prohibits their coupling with optimisation schemes, where it is required to evaluate multiple experimental designs to find the maximally informative solution. Such limitations increase in the presence of models of higher complexity, noisy experimental data or incomplete observability of all state variables, all common elements in biological studies [316].

2.5.4 Optimal Experimental Design for Model Selection

Despite its recent upswing, OED for model selection has received less attention than optimal design strategies directed at parameter identification [317]. It is common to formulate alternative model structures, specifically when the biological system under study is still partly unknown. The traditional model

discrimination approach is to reiteratively acquire new experimental data and re-fit the model parameters until finding experiments that make the wrong models unable to predict all the data sets [318]. For this reason, OED becomes an attractive tool to minimise experimental resources and time to identify the best candidate model due to the variability in informative content for the task of different experiments [319]. The general approach to the problem is to maximise the difference between model predictions, known as T-optimality [302], [320], [321], [322]. Such strategies can also include model prediction uncertainty measured by the Kullback-Leibler distance (KL-optimality) [321] or a different statistical distance (e.g. Hellinger distance) [318], [323]. This prediction uncertainty is usually computed from the FIM in frequentist approaches or by using parameter samples from prior distributions in Bayesian ones [321], [324]. Chapter 5 further investigates such strategies for OED using the genetic toggle switch as a case study.

2.6 Aims and Objectives

The foremost important aim of this thesis is to contribute to the transition of some biological disciplines to more quantitative and predictive ones by incorporating a fundamental tool; mathematical modelling. To achieve this aim, the objectives of this work are:

- Exemplify the efficient use of mathematical models to describe diverse biological systems.
- Supply strategies and platforms for an easy use of mathematical models in biological sciences and, consequently, aid their widespread incorporation in less experienced domains.
- Spotlight the potential advantages that probabilistic (Bayesian) approaches provide in developing and characterising mathematical models for biological sciences.
- Highlight the importance of experimental design and *a priori* information quantification to maximise the informative content of data.
- Underline the complexity of the model development process and experimental validations to stress the importance of an interdisciplinary approach and the development of tools for process automation.
- Present straightforward implementable solutions to the problems of parameter estimation, model selection and optimal experimental design.
- Show how recent technological and computational advancements are of crucial importance for the progress of scientific research.
- Provide an efficient strategy to process in real-time large microscopy data sets and build intricated experimental platforms upon this.
- Validate traditionally computationally restricted optimal experimental design strategies *in vivo* to show how to implement the theoretical aspects of this work into real scenarios.
- Deliver an easy-to-use and open-source platform that integrates computational methodologies for model simulations, parameter inference and optimal experimental design.

Chapter 3

Materials and Methods

The present work in this thesis uses an interdisciplinary approach to study the use of mathematical models to describe biological systems and tackle different roadblocks in their development. Hence, a combination of experimental, computational and mathematical methodologies were required for the task. This chapter contains all the necessary computational and experimental materials and methods needed to reproduce and perform the work contained within this thesis.

3.1 Materials

This section encloses the information regarding laboratory chemicals, reagents and equipment used for all experiments presented in the following Chapters 6 and 7. Specific software used for any computational work throughout the thesis is also indicated in this section. Next, the section specifies the yeast and bacteria strains used for the work described in Chapters 6 and 7, together with any genetic modifications included and the growth media used for experiments. Finally, the section specifies any experimental data sets retrieved from publications and used in this work for computational studies in Chapters 4 and 5.

3.1.1 Chemicals and Reagents

Table 3.1 contains a list of all the chemicals and reagents used in this study, including their common abbreviation (if any), their product code and UK supplier.

Table 3.1: Chemicals and reagents used in this study

Name	Abbreviation	Product Code	Supplier
Agar granulated	Agar	AGR10	Formedium Ltd.
Anhydrotetracycline	aTc	37919-100MG-R	Merck Life Sciences UK
Calcium dichloride	$CaCl_2$	10786634	Fisher Scientific UK Ltd
CasAminoAcids	-	1274-7109	Fisher Scientific UK Ltd
Chloramphenicol	CHL	C0378-25G	Merck Life Sciences UK
Glucose	Glu	49159-1KG	Merck Life Sciences UK
Hopkins SCSM supplementary mix drop-out trp, leu	Hopkins SCSM	DSCS011	Formedium Ltd.
Isopropyl ß-D-1-thiogalactopyranoside	IPTG	I6758-5G	Merck Life Sciences UK
L-Leucine	Leu	L8000-50G	Merck Life Sciences UK
M9, Minimal Salts 5X	M9 salts	M6030-1KG	Scientific Laboratory Supplies Ltd.
Magnesium Sulphate	$MgSO_4$	M7506-1KG	Merck Life

			Sciences UK
Miller Luria-Bertani Broth, Large Granules	LB Broth, Miller	12801660	Fisher Scientific UK Ltd
Macherey-Nagel NucleoSpin Plasmid Columns	Mini Prep Kit	15860928	Fisher Scientific UK Ltd
Polydimethylsiloxane	PDMS	101697	Scientific Laboratory Supplies Ltd.
Polyoxyethylene (20) sorbitan monolaurate	Tween 20	BSB0045-BSB-100ml	Stratech Scientific Ltd
Sluforhodamine 101	SR101	S7635-100MG	Scientific Laboratory Supplies Ltd.
Sulfo-Cyanine5 Carboxylic Acid	Sulfo-Cy5	ab146502-1mg	ABCAM INC.
L-Tryptophan	Trp	T0254-25G	Merck Life Sciences UK
Yeast nitrogen base w/o aa	-	Y0626-250G	Merck Life Sciences UK

3.1.2 Equipment

Table 3.2 contains a list of all the project-specific lab equipment used in this study, including the UK supplier used and general specifications.

Table 3.2: Equipment used in this study

Name	Supplier	Specifications
Nikon Eclipse Ti Inverted Microscope	Nikon UK Ltd.	- CFI60 Optics - Perfect Focus Technology - Motorised Control and Adquisition
ANDOR EMCCD camera iXon Ultra 888	Oxford Instruments	Active Pixels: 1024^2 Pixel size: 13 μm
Okolab H101-Basic-B	Nikon UK Ltd.	-
Lumencor Spectra X Light Engine	Nikon UK Ltd.	- Light sources: Blue (440/20), Cyan (470/24), Teal (510/25), Yellow (575/25), Red (640/30).
10X Plan Fluor Ph 1 DLL objective	Nikon UK Ltd.	-
40X Plan Fluor Oil DIC H N2 objective	Nikon UK Ltd.	-
100X Plan Fluor Oil DIC H N2 objective	Nikon UK Ltd.	-
49000 ET-DAPI Fluorescence Filter	Nikon UK Ltd.	-Ex: 350/50 -Em: 460/50

49002 ET-GFP(FITC/Cy2) Fluorescence Filter	Nikon UK Ltd.	- Ex: 470/40 - Em: 525/50
49905 ET-514nm Laser Bandpass Set for EYFP, Venus, Citrine, Fluo3 Fluorescence Filter	Nikon UK Ltd.	- Ex:514/10 - Em: 545/40
49909 ET-561nm Laser Bandpass Set for TagRFP, tdTomato, AlexaFluor 546, TRITC Fluorescence Filter	Nikon UK Ltd.	- Ex: 561/10 - Em: 600/50
49006 ET-Cy5 Fluorescence Filter	Nikon UK Ltd.	- Ex: 620/60 - Em: 700/75
69308 ET-ECFP/EYFP/mCherry with single band exciters Fluorescence Filter	Nikon UK Ltd.	- Ex: 430/24, 500/20, 557/25 - Em: 470/10, 535/15, 635/35
Linear actuators 800mm Stroke	LC Automation Ltd.	- Series: RCP2 - Type: SA7C - Motor: 56P
Reusable Biopsy Punch, sterile	World Precision Instruments Limited UK	- 0.75mm i.d. - For bacteria chip

		ports
Disposable Biopsy Punch, sterile	Kai Medical	- 1mm i.d. - For yeast chip ports
Syringes Plastic Plastipak Luer Lokμ	Camlab Ltd	- 50ml
Masterflex Fitting, Polypropylene Adapters	Cole-Parmer	- Female Luer to Hosebarb Adapters - O.D.: 1.5875mm - Straight
Saint-Gobain Tygon ND 100-80 Tubing	Fisher Scientific UK Ltd	- I.D.: 0.51mm - O.D.: 1.52mm - Needle gauge: 23 - For bacteria chip
Fine Bore Polythene Tubing (SMME800/100/120)	Merk Life Sciences UK	- I.D.: 0.38mm - O.D.: 1.09mm - Needle gauge: 25 - For yeast chip
Ismatec Pump Tubing, Tygon® E-Lab	Cole-Parmer	- I.D.: 0.89mm - To connect smaller tubing to syringe adapters.
Precision Bend 90 Degree Dispense Tip	- Adhesive - Dispensing LTD	I.D.: 0.413mm O.D.: 0.7176mm - Needle gauge: 22

	- To connect tubing to ports in bacteria chip

3.1.3 Software

Table 3.3 contains a list of all the non-custom software used in this study, plus their versions and distributors.

Table 3.3: Software used in this study

Name	Version	Distributor
Atom	1.48.0	Atom
ImageJ	2.3.0	Fiji
Inkscape	0.92.4	Inkscape Project
Julia	1.4.0 - 1.7.2	Julia Computing, Inc.
Matlab	R2018a, R2018b, R2019b	MathWorks
μmanager	1.4.23	Micro-Manager
NIS-Elements AR	4.51.00	Nikon
Python	3.6.10	Anaconda
R	3.5.1	r-project
R-Studio	1.2.5033	RStudio
RTools	3.4.0.1964, 3.5.0.4	CRAN
Stan	2.17.0 in Rstan, 2.20.0 in CmdStan	Stan
Ti-Control	4.4.0	Nikon
VS Code	1.60-1.72	Microsoft Corporation

3.1.4 Bacteria Strain

The *Escherichia coli* K-12 BW25113 strain used to study a genetic toggle switch in this work was the one described and provided by Lugagne *et al.* 2017 [85]. The

previously mentioned authors knocked out in the strain the *acrA* and *acrB* genes to prevent adaptation to prolonged high levels of aTc, the *fliA* gene to down-regulate flagella synthesis and *lacY* to avoid active trans-port of IPTG. Additionally, a low copy number plasmid (derived from pACYC with 10-12 copies/cell) was transformed into cells containing the genetic toggle switch construct (*pTet-lacI-mKate2-Terminator* and *pLac-tetR-mEGFP-Terminator*) and a chloramphenicol resistance cassette (CmR) for selective pressure. For more information see Lugagne *et al.* 2017 [85].

3.1.5 Yeast Strain

The *Saccharomyces cerevisiae* BY4741 strain used to study an inducible promoter in this work was the one described and provided by Gnügge *et al.* 2016 [228]. The authors used a series of plasmid vectors (pRG101,pRG103, pRG105 and pRG106) for the genetic integration of synthetic constructs. The primary integrated construct was the orthogonal inducible promoter expressing a fluorescent reporter ($tetO_2$-$P_{CYC1-TATA}$-$lacO_2$- $P_{CYC1-TSS}$-Citrine-T_{CYC1}). Additionally, there were insertions to ensure constitutive expression of the Lac repressor (P_{ACT1}-lacI-T_{CYC1}), Tet transactivator repressor (P_{ACT1}-tetR-VP16AD-T_{CYC1}) and lactose transporter (P_{ACT1}-LAC12-mTfp1-T_{CYC1}). Finally, there was an insertion of a fluorescent reporter to monitor the expression of the last three genes (P_{ACT1}-mKate2-T_{ADH1}). For more information see Gnügge *et al.* 2016 [228].

3.1.6 Growth Media

Table 3.4 contains a list of all the bacterial (LB Broth and M9 Minimal Medium) and yeast (SC) mediums used in this study with information on their composition.

Table 3.4: Bacteria and Yeast medium used in this study

Name	Composition
Miller Luria-Bertani Broth (LB Broth)	10 g/L Tryptone, 5g/L Yeast Extract, 10 g/L Sodium chloride (NaCl). The medium was supplemented with 0.5 % Glucose (filter-sterilised). If solid media

	is required, 20 g/L Agar granulated.
M9 Minimal Medium	200 mL/L M9 5X salts, 25 mL/L Thiamine Hydrochloride (10 mg/mL), 40 mL/L Glucose 20 %, 20 mL/L CasAminoAcids 10 %, 20 mL/L MgSO4 0.1M, 1 mL/L CaCl2 0.1 M. Thiamine Hydrocloride, Glucose, MgSO4 and CaCl2 must be filter-sterilised.
Synthetic Complete (SC)	6.8 g/L Yeast Nitrogen Base (without aminoacids), 1.742 g/L Hopkins SCSM supplementary mix (drop-out Tryptophan, Leucine), 10 mL/L 100X Tryptophan, 10 mL/L 100X Leucine. The media was complemented with 2 % Glucose. If solid media is required, 20 g/L Agar granulated. Tryptophan, Leucine and Glucose must be filter-sterilised.

3.1.7 External Experimental Data for the Genetic Toggle Switch

Experimental data from Lugagne *et al.* 2017 [85] was retrieved and used to perform model inference in Sections 4.1.3 (p. 117) and 4.3.1 (p. 131) of Chapter 4. Table 3.5 encloses a summary of all the experiments used (randomly selected), linking the identifiers used in this work with the ones used in Lugagne *et al.* 2017 [85], number of inducer concentration changes during the experiments (steps), average step duration and number of data samples. Additionally, Table 3.6 encloses a summary of experiments (randomly selected) used only for validation purposes in Section 5.1.1 (p. 141) of Chapter 5.

Table 3.5: Information of the experimental sets retrieved from Lugagne *et al.* 2017 [85] and used in this work

Experiment identifier	Original identifier	Number of steps	Average step duration (min)	Number of samples
Ec_1	$Calibration_4$	2	365	147
Ec_2	$Calibration_5$	3	225	181
Ec_3	$Calibration_6$	3	270	217
Ed_1	$DynStim_1$	13	85	222
Ed_2	$DynStim_2$	11	123	271
Ed_3	$DynStim_3$	13	74	193
Ed_4	$DynStim_8$	11	85	189
Ed_5	$DynStim_9$	9	133	241
Ed_6	$DynStim_{11}$	11	109	241
Ed_7	$DynStim_{14}$	13	85	223

Table 3.6: Information of the experimental sets retrieved from Lugagne *et al.* 2017 [85] and used in this work on validation steps

Experiment identifier	Original identifier	Number of steps	Average step duration (min)	Number of samples
Ec_4	$Calibration_1$	2	512	257
Ec_5	$Calibration_2$	3	398	300
Ec_6	$Calibration_3$	2	364	147
Ed_8	PI_3	180	5	181
Ed_9	$BangBang_2$	144	5	145

Ed_{10}	$DynStim_2$	13	100	262
Ed_{11}	$DynStim_3$	11	103	229
Ed_{12}	$DynStim_4$	13	88	231
Ed_{13}	$DynStim_{10}$	11	109	241
Ed_{14}	$DynStim_{12}$	11	103	229

3.2 Methods

This section encloses the information necessary to replicate any results (computational and experimental) presented in this thesis. First, there is a detailed methodology description (Sections 3.2.1, 3.2.2, 3.2.3, 3.2.4, 3.2.5, 3.2.6 and 3.2.7) for all mathematical models and computational and mathematical methodologies implemented and/or developed throughout Chapters 4, 5 and 7. Most of these methodologies are incorporated in the Julia package presented in Chapter 8. Then, the section will focus on the experimental conditions needed for the growth of bacteria and yeast cells (Sections 3.2.8 and 3.2.8) and the different settings implemented in our microscopy platform (Section 3.2.10). The section ends with a detailed description of the methodologies employed to perform and process the data (Section 3.2.11) from the microfluidic experiments presented in Chapters 6 and 7.

3.2.1 Mathematical Models and Simulations

3.2.1.1 *E. coli* Genetic Toggle Switch

The computational work enclosing the genetic toggle switch used three different model formulations $M_{1,2,3}$ (as sets of ODEs), derived from the model presented in Lugagne *et al.* 2017 [85]. The models were used to predict the expression dynamics of the two repressor proteins (LacI and TetR) of the system fused to two fluorescent proteins (RFP and GFP, respectively), dependent on the extracellular concentrations of the transcriptional regulators aTc and IPTG. A quasi-steady-state approximation on mRNA dynamics was used to solve structural identifiability issues obtaining as time derivative for the two repressor proteins:

$$\begin{aligned}
\frac{d[LacI]}{dt} &= \frac{1}{g_L^m}\left(\kappa_L^{pm0} + \frac{\kappa_L^{pm}}{1+\left(\frac{[TetR]}{\theta_{TetR}}\ \frac{1}{1+\left(\frac{[aTc]}{\theta_{aTc}}\right)^{\eta_{aTc}}}\right)^{\eta_{TetR}}}\right) - g_L^P\ [LacI] \\
\frac{d[TetR]}{dt} &= \frac{1}{g_T^m}\left(\kappa_T^{pm0} + \frac{\kappa_T^{pm}}{1+\left(\frac{[LacI]}{\theta_{LacI}}\ \frac{1}{1+\left(\frac{[IPTG]}{\theta_{IPTG}}\right)^{\eta_{IPTG}}}\right)^{\eta_{LacI}}}\right) - g_T^P\ [TetR],
\end{aligned} \tag{3.1}$$

where the protein concentrations $[LacI]$ and $[TetR]$ are expressed in arbitrary fluorescent units ($A.U.$) due to the fused fluorescent proteins RFP and GFP, respectively. The intracellular concentration of the inducers are $[IPTG]$ (mM) and $[aTc]$ ($ng \cdot mL^{-1}$). The degradation rates $g^m_{L/T}$ and $g^P_{L/T}$ were set to the constant rates $1.386 \cdot 10^{-1}\ min^{-1}$ and $1.65 \cdot 10^{-2}\ min^{-1}$ respectively, according to the average mRNA half-life and cell cycle times in *E. coli*. Here it was assumed that cell growth and division overshadowed protein degradation. The parameters $k^{pm0}_{L/T}$ and $k^{pm}_{L/T}$ are the product of the proteins translation and basal transcription rates and the proteins translation and maximum induced transcription rates respectively. The parameters η_i and θ_i, with $i = \{IPTG, aTc, LacI, TetR\}$, are the different Hill coefficients and dissociation constants for the protein-inducer and protein-DNA binding site complexes.

The differences between $M_{1,2,3}$ rely on the ODEs that model the flux of $IPTG$ and aTc between the extracellular medium and the intracellular environment.

Model M_1

The first model considered was retrieved from Lugagne *et al.* 2017 [85], assuming a non-symmetrical movement of the inducers between the intracellular and extracellular compartments:

$$\begin{aligned}
\frac{d[aTc]}{dt} &= \begin{cases} k^{in}_{aTc}(u_{aTc} - [aTc]), & \text{if} \quad u_{aTc} > [aTc] \\ k^{out}_{aTc}(u_{aTc} - [aTc]), & \text{otherwise} \end{cases} \\
\frac{d[IPTG]}{dt} &= \begin{cases} k^{in}_{IPTG}(u_{IPTG} - [IPTG]), & \text{if} \quad u_{IPTG} > [IPTG] \\ k^{out}_{IPTG}(u_{IPTG} - [IPTG]), & \text{otherwise,} \end{cases}
\end{aligned} \tag{3.2}$$

where $k^{in/out}_{aTc/IPTG}$ are the different transport rates and u_{IPTG} and u_{aTc} the extracellular concentrations of the inducers. This model was used extensively in the work presented in Chapters 4 and 5.

Model M_2

The second model developed assumes passive diffusion of aTc and IPTG between the two compartments, following Fick's law of diffusion according to the strain genetic modifications stated in Section 3.1.4 (p. 59) of this chapter:

$$\begin{aligned} \frac{d[aTc]}{dt} &= k_{aTc}\,(u_{aTc} - [aTc]) \\ \frac{d[IPTG]}{dt} &= k_{IPTG}\,(u_{IPTG} - [IPTG]). \end{aligned} \tag{3.3}$$

This model was used extensively in the work presented in Chapters 4 and 5.

Model M_3

The third model developed builds on M_2 by introducing an effective degradation rate g^P (fixed at $1.65 \cdot 10^{-2}\ min^{-1}$) for the inducers due to cell growth and division:

$$\begin{aligned} \frac{d[aTc]}{dt} &= k_{aTc}\,(u_{aTc} - [aTc]) - g^P\,[aTc] \\ \frac{d[IPTG]}{dt} &= k_{IPTG}\,(u_{IPTG} - [IPTG]) - g^P\,[IPTG]. \end{aligned} \tag{3.4}$$

This model was used extensively in the work presented in Chapters 4, 5, 6, 7 and 8.

In all instances, aTc was constrained to a maximum value of 100 ng/mL and IPTG to a maximum value of 1 mM. Table 3.7 contains more information about the parameters as well as the bounds and priors used in this work. The priors represented include mean and standard deviation ($\mathcal{N}(\mu, \sigma)$ or $Log\mathcal{N}(\mu, \sigma)$) as parameters, following Stan's notation [83]. Table 3.8 shows the different parameter values selected for computational studies of model M_3.

Table 3.7: Parameters information for the toggle switch models

Parameter	Units	Bounds	Prior_a	Prior_b
κ_L^{pm0}	$\frac{1}{min^2}$	$[2.7 \cdot 10^{-3}, 0.34]$	$Log\mathcal{N}(-3.51, 1.15)$	$\mathcal{N}(0.17, 0.08)$
κ_L^{pm}	$\frac{1}{min^2}$	$[0.89, 1.12 \cdot 10^{2}]$	$Log\mathcal{N}(2.30, 1.15)$	$\mathcal{N}(56.55, 27.83)$
θ_{TetR}	$A.U.$	$[2.67, 3.37 \cdot 10^{2}]$	$Log\mathcal{N}(3.40, 1.15)$	$\mathcal{N}(169.64, 83.48)$
θ_{aTc}	$\frac{ng}{mL}$	$[0.89, 1.12 \cdot 10^{2}]$	$Log\mathcal{N}(2.30, 1.15)$	$\mathcal{N}(56.55, 27.83)$
$\eta_{aTc/IPTG}$	-	$[0, 5.13]$	$\mathcal{N}(2.50, 1.25)$	$\mathcal{N}(2.56, 1.28)$
$\eta_{TetR/LacI}$	-	$[0, 5.13]$	$\mathcal{N}(2.50, 1.25)$	$\mathcal{N}(2.56, 1.28)$
κ_T^{pm0}	$\frac{1}{min^2}$	$[5.6 \cdot 10^{-3}, 1.78]$	$Log\mathcal{N}(-2.30, 1.15)$	$\mathcal{N}(0.89, 0.44)$
κ_T^{pm}	$\frac{1}{min^2}$	$[8.9 \cdot 10^{-2}, 11.2]$	$Log\mathcal{N}(2.22 \cdot 10^{-16}, 1.15)$	$\mathcal{N}(5.65, 2.78)$
θ_{LacI}	$A.U.$	$[2.67, 3.37 \cdot 10^{2}]$	$Log\mathcal{N}(3.40, 1.15)$	$\mathcal{N}(169.64, 83.48)$
θ_{IPTG}	mM	$[8.9 \cdot 10^{-3}, 1.12]$	$Log\mathcal{N}(-2.30, 1.15)$	$\mathcal{N}(0.57, 0.28)$
$k_{aTc}/k_{aTc}^{in/out}$	$\frac{1}{min}$	$[8.9 \cdot 10^{-3}, 1.12]$	$Log\mathcal{N}(-2.30, 1.15)$	$\mathcal{N}(0.57, 0.28)$
$k_{IPTG}/k_{IPTG}^{in/out}$	$\frac{1}{min}$	$[3.6 \cdot 10^{-3}, 0.45]$	$Log\mathcal{N}(-3.22, 1.15)$	$\mathcal{N}(0.23, 0.11)$

The bounds selected for the different parameters cover an order of magnitude expansion or reduction the parameter values reported in Lugagne *et al.* 2017 [85] except for the Hill coefficients $\eta_{aTc,IPTG,TetR,LacI}$ which range between experimentally plausible values. In this manner, the priors Prior_a (mostly used in inferences presented in Chapters 4 and 5) were derived considering the parameter values reported in Lugagne *et al.* 2017 [85] as the mean for the Log-Normal distributions and $2 \cdot \sigma$ accounts for the one order of magnitude deviation from the mean (except for the Hill coefficients). The priors Prior_b (used in inferences presented in Chapters 6 and 7) were derived from the selected bounds assuming normally distributed priors such that $\mu = (U + L)/2$ and $\sigma = (U - L)/4$, where U indicates the upper and L the lower bounds for each parameter.

Table 3.8: Parameters values selected for computational studies for the toggle switch model M_3

Parameter	Θ_{TS}^{T1}	Θ_{TS}^{T1}
κ_L^{pm0}	0.05980078	0.05980078
κ_L^{pm}	6.32831827	36.32831827
θ_{TetR}	91.80251520	91.80251520
θ_{aTc}	0.91779024	40.91779024
η_{aTc}	0.44236521	2
η_{TetR}	3.26829704	2
κ_T^{pm0}	0.27499435	0.27499435
κ_T^{pm}	4.48180358	4.48180358
θ_{LacI}	2.68995461	112.68995461
θ_{IPTG}	0.01720242	0.41720242
η_{IPTG}	0.80772911	2
η_{LacI}	0.71316179	2
k_{aTc}	0.01334992	0.41334992
k_{IPTG}	0.4437322	0.2437322

The parameter values Θ_{TS}^{T1} were derived from the multi-experimental inference of model M_3 obtained in Chapter 4 by considering the mean vector of the multidimensional multi-variate posterior. The parameter values Θ_{TS}^{T1} were obtained from the previous set by manually adjusting parameter estimates that were close to their respective bounds, modifying them to be closer to the means of Prior_b. The second set of parameters was introduced to investigate the effect of proximity of parameters used to generate pseudo-data to regions of low prior probability during Bayesian inference schemes. Both are used for *in silico* studies in Chapter 4 and Section 7.1 (p. 186) of Chapter 7.

3.2.1.2 *S. cerevisiae* Orthogonal Inducible Promoter

The computational work enclosing the orthogonal inducible promoter developed by Gnügge *et al.* 2016 [228] used the mathematical model described in Bandiera *et al.* 2018 [86]. The model was used to predict the expression dynamics (transcription, translation and maturation) of the fluorescent protein Citrine, subject to the extracellular concentration of the transcriptional regulator IPTG:

$$\begin{aligned}
\frac{d[mRNA]}{dt} &= \alpha_1 - V_m \frac{[IPTG]^h}{K_m^h + [IPTG]^h} - \lambda[mRNA] \\
\frac{d[Citrine_f]}{dt} &= \alpha_2[mRNA] - (\lambda_2 + k_f)[Citrine_f] \\
\frac{d[Citrine_m]}{dt} &= k_f[Citrine_f] - \lambda_2[Citrine_m] \\
[Citrine_{A.U.}] &= sc_{molec}[Citrine_m]
\end{aligned} \tag{3.5}$$

where $[mRNA]$ represents the mRNA concentration, $[Citrine_f]$ the concentration of immature (folded) Citrine and $[Citrine_m]$ the concentration of mature fluorescent Citrine expressed as number of molecules per cell. $[Citrine_{A.U.}]$ represents the concentration of fluorescent Citrine expressed as arbitrary fluorescent units ($A.U.$) per cell, which is an the observable from microscopy experiments. In all instances, IPTG was constrained to a maximum value of $100\mu M$. Table 3.9 contains more information about the parameters as well as the bounds used in this work and parameter priors. The priors represented include mean and standard deviation ($\mathcal{N}(\mu, \sigma)$) as parameters, following Stan's notation [83]. Table 3.10 shows the different parameter values selected for computational studies of model. The model was used in computational studies presented in Chapter 4 and experimental ones presented in Chapter 6.

Table 3.9: Parameters information for the orthogonal inducible promoter

Parameter	Description	Units	Bounds	Prior
α_1	Basal transcription	$\frac{1}{min}$	$[3.88 \cdot 10^{-5}, 4.95 \cdot 10^{-1}]$	$\mathcal{N}(0.25, 0.12)$
V_m	Maximum induced transcriptional rate	$\frac{1}{min}$	$[3.88 \cdot 10^{-2}, 4.95 \cdot 10^{-1}]$	$\mathcal{N}(0.27, 0.11)$
h	Hill coefficient	-	$[0.5, 4.9]$	$\mathcal{N}(2.70, 1.10)$
K_m	Michaelis-Menten constant	μM	$[2, 10]$	$\mathcal{N}(6.00, 2.00)$
λ_1	mRNA degradation rate	$\frac{1}{min}$	$[7.7 \cdot 10^{-3}, 0.23]$	$\mathcal{N}(0.12, 0.06)$
α_2	Translation rate	$\frac{1}{min}$	$[2.43 \cdot 10^{-1}, 6.81]$	$\mathcal{N}(3.53, 1.64)$
λ_2	Citrine degradation rate	$\frac{1}{min}$	$[5.98 \cdot 10^{-5}, 0.25]$	$\mathcal{N}(0.12, 0.06)$
k_f	Maturation rate	$\frac{1}{min}$	$[1.2 \cdot 10^{-2}, 2.17 \cdot 10^{-2}]$	$\mathcal{N}(0.017, 0.002)$
sc_{molec}	Scaling factor	$\frac{molec}{A.U}$	$[0.01, 100]$	$\mathcal{N}(50, 25)$

The parameter bounds selected were extracted from Bandiera *et al.* 2018 [86]. Parameter priors were assumed to be normally distributed with $\mu = (U + L)/2$ and $\sigma = (U - L)/4$, where U indicates the upper and L the lower bounds for each parameter.

Table 3.10: Parameters values selected for computational studies for the orthogonal inducible promoter model

Parameter	α_1	V_m	h	K_m	λ_1	α_2	λ_2	k_f	sc_{molec}
Θ_{PL}^T	0.0164	0.2916	1.7176	5.144	0.13	5.6378	0.0558	0.0177	69

The parameter values Θ_{PL}^T were result of a test maximum likelihood estimation performed on real experimental data (not shown) generated by Dr Lucia Bandiera (Univeristy of Edinburgh).

3.2.1.3 The Repressilator and Hes1 Models

The initial tests for the new Bayesian OED strategy for model inference presented in Sections 4.3 (p. 131) of Chapter 4 and 7.1 (p. 186) of Chapter 7 used two models described in Liepe *et al.* 2013 [56] to allow comparison to an already established method. The first model is used to describe the mRNA and protein dynamics of the repressilator [10], a synthetic oscillatory system:

$$\begin{aligned}
\frac{d[mRNA_1]}{dt} &= -[mRNA_1] + \frac{\alpha}{1+[Prot_3]^h} + \alpha_0 \\
\frac{d[mRNA_2]}{dt} &= -[mRNA_2] + \frac{\alpha}{1+[Prot_1]^h} + \alpha_0 \\
\frac{d[mRNA_3]}{dt} &= -[mRNA_3] + \frac{\alpha}{1+[Prot_2]^h} + \alpha_0 \\
\frac{d[Prot_1]}{dt} &= \beta([mRNA_1] - [Prot_1]) \\
\frac{d[Prot_2]}{dt} &= \beta([mRNA_2] - [Prot_2]) \\
\frac{d[Prot_3]}{dt} &= \beta([mRNA_3] - [Prot_3])
\end{aligned} \tag{3.6}$$

The top half of Table 3.11 contains more information about the model parameters and the priors used in the study. Table 3.12 shows the different parameter values selected for computational studies of the repressilator model. The set of initial conditions y_0 for the different model states was set as defined in Liepe *et al.* 2013 [56] as $y_{0,i} = 0, 0, 0, 1, 0, 0$ for i being the model state defined in the same order as the above equations.

The second model describes the mRNA and cytoplasmatic ($Prot_1$) and nuclear ($Prot_2$) protein dynamics of the Hes1 oscillator [325]:

$$\begin{aligned}
\frac{d[mRNA]}{dt} &= -k_d + \frac{1}{1 + (\frac{[Prot_2]}{P_0})^{h_b}} \\
\frac{d[Prot_1]}{dt} &= -k_d[Prot_1] + v[mRNA] - k_1[Prot_1] \\
\frac{d[Prot_2]}{dt} &= -k_d[Prot_2] + k_1[Prot_1]
\end{aligned} \tag{3.7}$$

As in the original study from Liepe *et al.* 2013 [56], the protein degradation rate k_d was fixed to a value of $0.03min^{-1}$. The bottom half of Table 3.11 contains more information about the model parameters and the priors used in the study. Table 3.13 shows the different parameter values selected for computational studies of the Hes1 model. The set of initial conditions y_0 for the different model states was set as defined in Liepe *et al.* 2013 [56] as $y_{0,i} = 0.5, 3.0, 2.0$ for i being the model state defined in the same order as the above equations. Alternatively the set $y_{0,i} = 100.5, 1.0, 10.0$ was also used when specified. Both models were used for computational studies presented in Chapter 4.

Table 3.11: Parameters information for the Repressilator and Hes1 models

Parameter	Description	Units	Priors
α	Transcription Rate	$\frac{1}{min}$	$\mathcal{U}(500, 2000)$
α_0	Basal transcription rate	$\frac{1}{min}$	$\mathcal{U}(0, 20)$
h	Hill coefficient (Repressilator)	-	$\mathcal{U}(1, 10)$
β	Ratio of protein to mRNA decay rate	-	$\mathcal{U}(0, 10)$
P_0	Concentration of protein in nucleus, when transcription rate is at half its maximum value	$\frac{molec}{cell}$	$\mathcal{U}(0, 2)$
h_b	Hill coefficient (Hes1)	-	$\mathcal{U}(1, 10)$

v	Translation rate	$\frac{1}{min}$	$\mathcal{U}(0, 0.1)$
k_1	Rate of protein transport to nucleus	$\frac{1}{min}$	$\mathcal{U}(0, 0.1)$

Table 3.12: Parameters values selected for computational studies for the repressilator model

Parameter	α	α_0	h	β
Θ_R^T	1000	10	2	5

Table 3.13: Parameters values selected for computational studies for the Hes1 model

Parameter	P_0	h_b	v	k_1
Θ_H^T	1.4	5.7	0.02	0.09

All priors and parameter values used for the Repressilator and the Hess1 system were extracted from Liepe *et al.* 2013 [56].

3.2.1.4 Simulations and Event Inputs Implementation

To simulate the response of the genetic toggle switch (equations 3.1, 3.2, 3.3, 3.4) and the inducible promoter (equation 3.5), subject to step-wise perturbations of the external inducers in an experiment, an event inputs implementation was introduced. Nonetheless, this implementation was unnecessary for any simulations performed in Matlab using the toolbox AMIGO2 [74], since this is a tool that already integrates the simulation of this type of model besides other functions (e.g. identifiability and sensitivity analysis, parameter estimation or frequentist optimal experimental design). This implementation was used to simulate the models and generate computational results presented in Chapters 4, 5, 6 and 7. This was also incorporated in the Julia package presented in Chapter 8. The event inputs implementation relies on solving the set of ODEs iteratively for each section of the experiment with a constant value of the inducers. The initial condition for each section was then set to the value that the states reached

in the previous one. Unless stated otherwise, the initial condition for any experiment was set to the analytical steady-state of the system given an inducer concentration (inducible promoter) or to the numerical steady-state attained after 24 hours of constant induction (toggle switch) in concordance to the over-night induction used experimentally. For the genetic toggle switch, unless stated, over-night induction was done using maximum IPTG concentration (1 mM) and absence of aTc.

Model simulations were also used to generate pseudo-data traces with heteroscedastic (variances depend on data observations) or homoscedastic (variances are constant independently of observations) additive Gaussian noise. This results in the formulation for the pseudo-data $\hat{y}$:

$$\hat{y}_s^{e,o} = y_s^{e,o} + \epsilon_s^{e,o} \tag{3.8}$$

where y indicates a model prediction for an experimental design e, system observable o and sampling time s, and ϵ is the error added in order to obtain the pseudo-data $\hat{y}$. For heteroscedastic additive Gaussian noise ϵ was defined as:

$$\epsilon_s^{e,o} = \mathcal{N}(0, y_s^{e,o} \cdot pr) \tag{3.9}$$

with pr indicating a percentage of $y_s^{e,o}$. In this work, 10 % and 20 % were used, representing low and moderate error with respect to the variances obtained from our fluorescence microscopy data on average. For the homoscedastic additive Gaussian noise ϵ was defined as:

$$\epsilon_s^{e,o} = \mathcal{N}(0, \sigma) \tag{3.10}$$

with σ indicating a previously set standard deviation for each experiment.

Simulations done in Stan used the *bdf* integrator [326] during Bayesian inference and the $rk45$ integrator [327] for model predictions. Simulations in Matlab (using the toolbox AMIGO2 [74]) used the solver $CVODES$ [326]. Simulations performed in Python used the odeint library included in SciPy [328]. Simulations performed in Julia used the package DifferentialEquations.jl, using the solver $CVODE_BDF$ [326] included in Sundials.jl.

3.2.2 Model Parameter Identification for ODEs

All the models to describe biological systems used in this work follow the form:

$$\frac{d[x]}{dt} = f(x, \theta, u, t) \tag{3.11}$$

to describe the dynamics of the state variables x given a vector of model parameters θ, a vector of external factors u (set to nothing for the repressilator and Hes1 models) at a given time t [229]. The model parameters θ were estimated using frequentist and Bayesian approaches given a series of *in silico* (simulated) or *in vivo* (experimental) data, subject to experimental designs e, observables o and sampling times s. Hence, in order to estimate the different model parameters a function that establishes the relationship between experimental data and the different model states was used, the observation function:

$$y_s^{e,o} = g_s^{e,o}(x, \theta, u, t) \tag{3.12}$$

as stated in Chapter 2, section 2.3.6 (p. 33).

Experimental schemes e were defined by the amount of the external inducers for each time t (toggle switch and inducible promoter) or the initial condition for the state variables (Hes1). For the toggle switch models, the observables o were set (unless specified differently) to the states $LacI$ and $TetR$. For the inducible promoter model, the observable was set to the state $Citrine_{A.U.}$. For the

repressilator, the observable state was set to $Prot_1$. For the Hes1 model it was set to either $mRNA$ or else the total protein level ($Prot_1 + Prot_2$). Unless specified, sampling times s were considered to happen every 5 min for the toggle switch (10 min for *in vivo* studies) and inducible promoter. For the repressilator, the sampling time was set every 2 arbitrary time units and every 30 for the Hes1 system.

3.2.2.1 Frequentist Parameter Estimation

The goal of parameter estimation was set to estimate the different parameters θ for each of the models minimising the distance between model predictions and experimental data. To assess this distance the weighted least-squares errors function:

$$J_{lsq}(\theta) = \sum_{e=1}^{E}\sum_{o=1}^{O}\sum_{s=1}^{S}\frac{(y_s^{e,o}(\theta) - \hat{y}_s^{e,o})^2}{(\sigma_s^{e,o})^2} \tag{3.13}$$

and the log-likelihood function (assuming normally distributed noise):

$$J_{llk}(\theta) = \sum_{e=1}^{E}\sum_{o=1}^{O}\sum_{s=1}^{S} -\frac{1}{2}\left[\log(2\pi) + \log((\sigma_s^{e,o})^2) + \frac{(y_s^{e,o}(\theta) - \hat{y}_s^{e,o})^2}{(\sigma_s^{e,o})^2}\right] \tag{3.14}$$

were used. E indicates the total number of experiments, O the total number of observables and S the total number of sampling times. Overall, the two metrics represent a final sum of errors over samples, observables and experiments.

To estimate the different model parameters (decision variables), subject to some constraints in their feasibility range, the problem was defined as a non-linear optimisation to minimise $J_{lsq}(\theta)$ or $J_{llk}(\theta)$. Due to the non-convex tendency of these problems (posing many potential sub-optimal solutions) [229], this was solved using global optimisation methods. Unless stated otherwise, frequentist parameter estimation was solved using the Matlab toolbox AMIGO2 [74] and the hybrid metaheuristic algorithm enhanced Scatter Search (eSS) [252] (Chapter 5). When the problem was solved in Julia, I used custom code and a genetic

algorithm (i.e. differential evolution) from the package BlackBoxOptim.jl (Chapter 7).

3.2.2.2 Bayesian Inference

In a Bayesian framework, the posterior probabilities $P(\theta|D)$ for the model parameters were approximated following Bayes' theorem:

$$P(\theta|D) = \frac{P(D|\theta)\ P(\theta)}{P(D)} \tag{3.15}$$

where $P(D|\theta)$ is the likelihood term of parameters θ given the experimental data D, $P(\theta)$ is the prior information about the model parameters θ and $P(D)$ is the marginal likelihood or model evidence (a normalisation term that ensures that $P(\theta|D)$ integrates to 1) [269]. Due to the complexity of computing $P(D)$, the Bayes' theorem is formulated as:

$$P(\theta|D) \propto P(D|\theta)\ P(\theta) \tag{3.16}$$

indicating that the posterior probability $P(\theta|D)$ is proportional to the product of the likelihood $P(D|\theta)$ and prior knowledge $P(\theta)$ [329]. Such proportionality relationship allows for the exploration of posterior distributions using sampling methods. To differentiate between different models M_i, equation 3.16 can be written as:

$$P(\theta_i|D,\ M_i) \propto P(D|\theta_i,\ M_i)\ P(\theta_i|M_i) \tag{3.17}$$

Unless stated otherwise, the likelihood term $P(D|\theta_i,\ M_i)$ was assumed to follow a Gaussian distribution for the residuals between data $D_t^{e,o}$ and observables o at each time point t for each experiment e.

All inferences were performed using Stan [83], which uses Hamiltonian Monte Carlo (HMC), a sampling-based method to approximate $P(\theta|D)$. This Markov chain Monte Carlo (MCMC) method uses integration-based Hamiltonian simulation dynamics corrected by a metropolis acceptance step [330]. This platform was used thoughout the results presented in Chapters 4, 5, 6 and 7, and incorporated in the Julia package presented in Chapter 8.

To take maximum advantage of the algorithm without spending time in tuning its settings, this was automated by the No-U-Turn Sampler (NUTS) [331]. The only manually adjusted settings were the target average proposal acceptance probability (set to 0.95 to prevent divergent transitions), jitter (set to 0.5 to increase sampling flexibility in regions of high curvature) and maximum allowed tree depth (set to 13 to avoid premature termination of the algorithm). Unless indicated otherwise, a minimum of 4 parallel randomly initialised Markov chains were used for the inferences, targeting a final number of 8000 samples from the target posterior. As suggested by Stan developers [83], this amount of samples was sufficiently large to represent non-hard to sample posteriors. For each of the initialised chains, 1000 warmup iterations were used to allow the chains to converge to the equilibrium target posterior. For all inferences performed (except if stated otherwise), all chains successfully converged before this warmup period, making unnecessary its increase. When indicated, to increase the computational speed of inferences, a good initial estimate for the parameters to initialise the chains was inferred through maximum likelihood estimation (MLE) coupled with an optimisation scheme lowering the warmup iterations to a value between 200 and 500. The MCMC results were verified by visual examination of chain traces and final posterior kernel densities and by analysing the scale reduction factor, the autocorrelation function, the effective sample size and the Monte Carlo standard error.

Where possible, to ease inferences, reduce the computational time allocated to these and prevent divergent transitions, prior sampling was done in the [-2,2] sampling region [83], converting each draw to the correspondent value to the prior distribution defined. This strategy was extensively used in Chapters 4 and 7. This sampling region was selected to make all prior regions equivalent and in accordance with Stan sampling procedures, where unless enforced, all inferences will randomly start in this. For uniform priors, these were set to $\mathcal{U}(-2, 2)$ and sample conversion was done following:

$$\theta_i = LB + \frac{(\theta_s - R_{min})(UB - LB)}{R_{max} - R_{min}} \tag{3.18}$$

where θ_s indicates the prior sample from the [-2,2] region, θ_i the sample mapped to the original prior region, $R_{min} = -2$, $R_{max} = 2$, and LB and UB respectively indicate the lower and upper bounds of the original prior distribution. The same formulation was used to map samples coming from Laplace and Cauchy distributions. For Exponential, Gamma, Pareto and Rayleigh priors $R_{min} = 0$ and for Beta priors $R_{min} = 0$ and $R_{max} = 1$. Prior normal distributions were sampled from standard normal distributions ($\mathcal{N}(0, 1)$) and samples were then mapped to the original prior range following:

$$\theta_i = (\theta_s \cdot \sigma_n) + \mu_n \tag{3.19}$$

where μ_n and σ_n are the mean and standard deviation of the original normal prior distribution. For log-normal distributions, prior sampling was also done on standard normal priors and samples were then mapped to the original prior range following:

$$\theta_i = e^{(\theta_s \cdot \sigma_l) + \mu_l} \tag{3.20}$$

where μ_l and σ_l are the mean and standard deviation of the original log-normal prior distribution.

3.2.2.3 Inferred Posterior Quality Assessment

For computational studies, were the true θ_T generating the pseudo-data was know, the posterior sample distance to θ_T, d_{PSD}, was proposed as a metric to assess the inference results. To ease comparisons between parameters and inferences, each distance was normalised by the prior allowed upper ($P(\theta)_{Up}$) and

lower bounds ($P(\theta)_{Dw}$), resulting in:

$$d_{PSD} = \frac{P(\theta|D)_i - \theta_T}{P(\theta)_{Up} - P(\theta)_{Dw}} \tag{3.21}$$

where $P(\theta|D)_i$ indicated a sample i from the approximated sampled posterior $P(\theta|D)$. The summary distance metric d_{PSDM} was derived from d_{PSD} to assess the average normalised distance of the samples as:

$$d_{PSDM} = \frac{1}{Ps} \sum_{i=0}^{Ps} \sqrt{\left(\frac{P(\theta|D)_i - \theta_T}{P(\theta)_{Up} - P(\theta)_{Dw}} \right)^2} \tag{3.22}$$

where Ps indicated the total number of samples of the posterior $P(\theta|D)$.

In cases where the true θ_T was not known (*e.g.* when using experimental data), a metric from the posterior covariance was used to assess the inference results [82]. The metric computes the logarithm of the determinant of the covariance matrix computed from the posterior distribution, hence, assuming the posterior to be Normally distributed. This metric is comparable to computing the entropy of the distribution where all constant terms are removed, since the closed-form entropy of a multivariate Gaussian distribution with mean μ and covariance matrix Σ is defined as:

$$H(\mathcal{N}(\mu, \Sigma)) = \frac{1}{2} \log(|2\pi e \Sigma|) \tag{3.23}$$

The three metrics are extensively employed to analyse the inference results of Chapters 4 and 7.

3.2.3 Information theory to assess informative content of experiments

To assess the informative content of experiments an information-theoretic approach was followed (see Section 4.2.2, p. 127 of Chapter 4 and Chapter 7). The gain in information G_i after having performed inference i was set to the difference between the prior and posterior Shannon entropy [56]:

$$G_i = H(P(\theta)_i) - H(P(\theta|D)_i) \tag{3.24}$$

where the Shannon entropy for a prior $P(\theta)$ is defined as:

$$H(P(\theta)) = -\int P(\theta)\ \log(P(\theta))\ d\theta \tag{3.25}$$

and for the posterior $P(\theta|D)$ is defined as:

$$H(P(\theta|D)) = -\int P(\theta|D)\ \log(P(\theta|D))\ d\theta \tag{3.26}$$

For the instances where priors or distributions were defined by independent distributions with defined probability density functions and a closed-form for the Shannon entropy, this was analytically computed [332]. Considering that for independent priors the total prior probability is equal to the product of the marginal prior probabilities:

$$P(\theta) = \prod_{n=1}^{N} P(\theta_n) \tag{3.27}$$

with N equal to the total number of parameters for the model. For priors where all marginal distributions follow a Gaussian distribution formulation, the prior

entropy can be computed following equation 3.23 as:

$$H(P(\theta)) = \frac{1}{2}\log((2\pi e)^N|\Sigma|) \tag{3.28}$$

where Σ is a diagonal covariance matrix containing the variances of each marginal prior probability. In indicated cases, the multivariate normal prior came from reparameterisation of the original prior. In these instances, the reparameterised samples of the posterior distribution had to be used to assess posterior entropies [333].

To approximate the differential entropy of sampled posterior distributions or priors based on previous sampled posterior distributions, their probability density functions were first approximated as Gaussian mixtures of the form:

$$P(\theta|D) = \sum_{i=1}^{C} w_i \, \mathcal{N}(\theta;\ \mu_i, \Sigma_i) \tag{3.29}$$

where w_i, μ_i and Σ_i are the non-negative weights, means and covariance matrices of the i_{th} multivariate Gaussian component C. Note that w_i must follow:

$$\sum_{i=1}^{C} w_i = 1 \tag{3.30}$$

The Gaussian mixture parameters were estimated using the *mclust* R-package (Expectation-Maximisation algorithm) when computed using R and *ScikitLearnBase.jl* (K-means clustering algorithm) when computed using Julia. In both instances, the optimal number of components was selected within a range ([1-30] in R, [1-40] in Julia) based on the best Bayesian Information Criteria (R) or log-likelihood (Julia). The differential entropy of these Gaussian mixtures was then approximated using the method described in Huber *et al.* 2008 [334], based on using a second-order Taylor-series expansion for the logarithm term in equation 3.26, so that the integral can be solved.

3.2.4 Goodness of Fit Metrics

Different metrics to assess the prediction quality of inferred models were used in this work, especially in Section 5.1.1 (p. 141) of Chapter 5. The first two metrics, already defined, are the least-squares error function (equation 3.13) and the maximum log-likelihood function (equation 3.14), used for quick assessments of model fit quality on seen or unseen data. A similar metric used was the adimensional, normalised root mean square error (nRMSE) over all samples S and observables O, defined as:

$$nRMSE = \frac{1}{O}\sum_{o=1}^{O}\sqrt{\frac{1}{S}\sum_{s=1}^{S}\frac{(y_s^o(\theta)-\hat{y}_s^o)^2}{(\sigma_s^o)^2}} \tag{3.31}$$

The two information criteria used in this work were the Akaike information criterion (AIC), defined as:

$$AIC = 2k + \sum_{o=1}^{O}\sum_{s=1}^{S}\frac{(y_s^o(\theta)-\hat{y}_s^o)^2}{(\sigma_s^o)^2} \tag{3.32}$$

with k as the number of model parameters, and the Bayesian information criterion (BIC), [335], defined as:

$$BIC = k\cdot ln(S\cdot O) + \sum_{o=1}^{O}\sum_{s=1}^{S}\frac{(y_s^o(\theta)-\hat{y}_s^o)^2}{(\sigma_s^o)^2} \tag{3.33}$$

AIC and BIC introduce a penalisation term for the dimensionality of the model, with BIC posing a higher penalisation according to the number of parameters of the model and total number of samples observed. The previous metrics were also used to perform cross-validation on unseen data for a model to detect the most generalisable model and aid parameter identification.

The last metric used (only for Bayesian cases) was the estimation of the model evidence $P(D)$ approximated by the Laplace method [269] as:

$$P(D) \approx P(D|\theta_p) \cdot P(\theta_p) \left| \frac{H}{2\pi} \right|^{-\frac{1}{2}} \tag{3.34}$$

where H is the Hessian matrix of the posterior $P(\theta|D)$ when this is approximated as a multivariate Gaussian distribution, calculated using the parameter values θ_p that maximise the likelihood. This model evidence approximation introduces the Occam's factor $P(\theta_p) \left| \frac{H}{2\pi} \right|^{-\frac{1}{2}}$ to penalise models based on their parameter number and the prior selection [269].

3.2.5 Model Prediction Uncertainty Quantification

Once a set of ODEs was defined and parameters inferred (Section 3.2.2, p. 75 of this chapter), model prediction uncertainty was estimated using sample-based methods. In a Bayesian scenario, a posterior distribution $P(\theta|D)$ can be linked to the model prediction $h(\theta)$ through the prediction posterior [79]:

$$P(h|D, e_d) = \int P(h|\theta, e_d) P(\theta|D) \, d\theta \tag{3.35}$$

where e_d indicates a specific experimental design scheme.

Due to the absence of a closed-form for $P(\theta|D)$ and $P(h|D)$, parameter samples from the inferred posterior were used to simulate a system obtaining samples from the corresponding prediction posterior. Prediction uncertainty (PU) was measured as the Euclidean distance between the 99.5th and 0.5th percentiles [79] for each sampling time s, observable o and experiment e:

$$PU_e^C = \frac{1}{O} \sum_{o=1}^{O} \sum_{s=1}^{S} y(P(\theta|D))_{s,o}^{per=99.5} - y(P(\theta|D))_{s,o}^{per=0.5} \tag{3.36}$$

An alternative quantification when the prediction uncertainty shape was wanted to be taken into account was the approximation of the Shannon entropy for $y(\theta)$.

First, for each $y(P(\theta|D))_{s,o}$, these were fitted to Beta, Exponential, Log-Normal, Normal, Gamma, Laplace, Pareto, Rayleigh, Cauchy and Uniform distributions, where the one yielding the best log-likelihood was picked. The selection of distributions was based on choosing a subset of commonly used continuous distributions (mostly from the exponential family) of varying shapes (represented by the distribution moments). Then, since a differential entropy closed-form exists for each of these probability density functions, the entropy was analytically computed, resulting in a prediction uncertainty quantification defined as:

$$PU_e^E = \frac{1}{O}\sum_{o=1}^{O}\sum_{s=1}^{S} H(y(P(\theta|D))_{s,o}) \tag{3.37}$$

In all instances, a minimum of 400 samples from $P(\theta|D)$ were used. The selection of this lower bound was determined empirically to still represent the original distribution according to the Kolmogorov-Smirnov [336] (non-parametric test do determine if two samples are drawn from a same distribution) test while minimising the value as much as possible to reduce computational times. These were sampled from the fitted Gaussian mixture (see Section 3.2.3, p. 81 of this chapter) or randomly picked from the resulting inference sampled posterior ensuring a similar distribution shape using a two-sample Kolmogorov-Smirnov [336] test while maintaining covariances. This metrics to assess *a priori* the informative content of experiments were further investigated in computational studies (see Section 4.3, p. 131 of Chapter 4) and used to design optimal experiments (see Section 3.2.7, p. 87 of this chapter and Chapters 6 and 7).

To quantify model prediction uncertainties in a frequentist scenario, the same sampling-based method was used while assuming a multivariate Normal distribution for the parameter estimates. The best parameter estimates defined the mean for the distribution while the covariance matrix was bounded by the inverse of the Fisher information matrix (FIM), usually used under normality assumptions for uncertainties [80]:

$$FIM(\theta) = \sum_{e=1}^{E}\sum_{o=1}^{O}\sum_{s=1}^{S}\left[\frac{\partial y_s^{e,o}(\theta)}{\partial\theta}\right]\frac{1}{(\sigma_s^{e,o})^2}\left[\frac{\partial y_s^{e,o}(\theta)}{\partial\theta}\right]^T \tag{3.38}$$

following the Cramér-Rao inequality [79]:

$$Cov(\theta) \geq FIM(\theta)^{-1} \tag{3.39}$$

where $\left[\frac{\partial y_s^{e,o}(\theta)}{\partial \theta}\right]$ is the matrix containing the sensitivities of the different observables concerning each parameter. Then, the confidence intervals for a parameter i were computed using the covariance matrix approximated from the FIM as:

$$\pm t_{1-\alpha}^{n} \sqrt{Cov(\theta_{i,i})} \tag{3.40}$$

where $t_{1-\alpha}^{n}$ is a Student's t-distribution with n degrees of freedom and confidence interval $1-\alpha$ [265] or approximated using Monte-Carlo sampling [337]. FIM (see Section 6.2, p. 170 of Chapter 6), covariances and confidence intervals (see Section 4.3.1, p. 131 of Chapter 4) computations were always done using the Matlab toolbox AMIGO2 [74]. In instances where sampling of the multivariate Normal distributions results in unfeasible values for the parameters (negative values) due to the symmetry of the confidence intervals [79], these are still considered unless numerical integration errors were raised during model simulations. If so, samples were ignored if this would happen at low frequency. Otherwise, if errors occurred at high frequencies problematic parameters were set to 0, assuming a truncation for all parameter distributions at that value.

3.2.6 Bi-Stability Analysis for the genetic Toggle Switch

To study the changes in the stability properties for a model of genetic toggle switch subject to different concentrations of aTc and IPTG, matrices containing information about mono-stable or bi-stable behaviour of the system (and fluorescent reporter dominance) were computed. This approach was used to study the optimal experimental design for model selection results (see Section 5.2, p. 151 of Chapter 5) and further applied in computational studies for the genetic

toggle switch (see Section 7.1, p. 186 of Chapter 7). First, both inducers were linearly discretised at 1 % between the set bounds, resulting in a total of 101 points for each one. Then, for each possible combination of aTc and IPTG values ($101 \times 101 = 10201$), the steady-state equations for the model were used to analytically compute the model nullclines (over reasonable values for the model observables) to assess the presence of hysteresis. The resulting nullclines intersections were used to identify the model fixed stable points [338] graphically, where only one intersection point indicated mono-stability and three intersection points indicated bi-stability (two stable steady-states plus a meta-stable state). To compute the intersections of the curves, the Matlab function InterX (Curve intersections) was used. To construct the bistability matrices, the number and value of the model steady-states was computed analytically to increase computational efficiency and accuracy. All model states where set to the analytical steady-state equation except for one of the two fluorescent reporters. Then, the zeros of the remaining differential equation using the Julia package Roots.jl were identified to find the steady-states of the system (number of times the ODE equals to 0) in a feasible range of values for the observable (from 0 to 100000). In the case of mono-stability, we performed the computation for the two system observables in order to obtain the steady-state fluorescence value for each one. Finally, mono-stable points were split into RFP-LacI or GFP-TetR dominant according to the model's highest expressed reporter. This representation of a model stability properties eases the comparison of steady-state stability properties for different models by comparing the computed results for each allowed combination of aTc and IPTG.

3.2.7 Optimal Experimental Design

Optimal experimental design (OED) was used to minimise time and laboratory resources while maximising the amount of information extracted from experiments. Due to experimental constraints that might result in low data quantity and/or quality, OED was proposed to aid the selection of the best model amongst a set of equally plausible ones (model selection) and identification of model parameters (model calibration/inference). These model-based strategies aim to calculate the optimum external stimuli profiles to apply to a system to ease model selection and identification, subject to algebraic (and experimental) constraints on the inducers u_i:

$$u_i^L \leq u_i \leq u_i^U \tag{3.41}$$

Two different OED strategies were included in this work; off-line OED and on-line OED. Off-line OED strategies follow the traditional approach of optimally designing experimental profiles (i.e. external stimuli profiles) including previous information about the system or past experiments (if any), and then experimentally executing these [229]. Alternatively, on-line OED exploits the information gathered during an experiment interval (sub-experiment) to update model information and optimally re-design the remaining experimental profile conditions before finalising it [86]. The second strategy aims to potentially reduce further the total amount of experiments needed to obtain, at least, the same amount of information. Unless specified otherwise, for all computational work, designed experiments were constrained to a maximum time of 24 hours with a sampling frequency of 5 minutes. Experimental work for the genetic toggle switch used a maximum time of 38 hours and a sampling frequency of 10 min. The input profile u was allowed to vary (subject to the constraints stated in equation 3.41) in equally time-spaced administration steps N (where $N \in \mathbb{N}$), where the distance between time steps was set to 3 hours (unless indicated otherwise).

3.2.7.1 Optimal Experimental Design for Model Selection

OED for model selection was considered in both frequentist and Bayesian scenarios. The aim for both strategies was to optimally design the external stimuli profiles u that maximised the systems observables discrepancy between two competing models. The difference between the frequentist and the Bayesian approaches resides in the fact that the latter includes information about model uncertainty. Both strategies were formulated as non-linear dynamic optimisation problems solved by global optimisation methods. The further described strategies were implemented using a case study (the genetic toggle switch) in Chapter 5.

In the frequentist scenario, the distance between model observables for an experiment presenting an external stimuli u was quantified as the Euclidean distance between the model observables predictions o for each sampling time s for the two competing models M_α and M_β:

$$d(M_\alpha, M_\beta) = \sqrt{\sum_{s=1}^{S} (y_\alpha^{s,o}(\theta_\alpha, u) - y_\beta^{s,o}(\theta_\beta, u))^2} \tag{3.42}$$

Regarding the assessment of multiple observables when needed (i.e. the genetic toggle switch), two options were investigated, the average of Euclidean distances for the observables:

$$d_{mean}(M_\alpha, M_\beta) = \frac{1}{O} \sum_{o=1}^{O} d(M_\alpha, M_\beta) \tag{3.43}$$

and the product of Euclidean distances for the observables:

$$d_{prod}(M_\alpha, M_\beta) = \prod_{o=1}^{O} d(M_\alpha, M_\beta) \tag{3.44}$$

The non-linear optimisation problem of finding the external stimuli u over time so as to maximise $d(M_\alpha, M_\beta)$, $d_{mean}(M_\alpha, M_\beta)$ or $d_{prod}(M_\alpha, M_\beta)$ was solved using the hybrid meta-heuristic solver eSS. All computations were done using the Matlab toolbox AMIGO2 [74]. Unless stated otherwise, $fmincon$ was used as the local optimisation solver for the hybrid algorithm eSS and the total number of function evaluations was set to 800.

In the Bayesian scenario, the distance between model observables was quantified as the Bhattacharyya distance [339] (closed at the second moment) between the model observables posterior predictive distributions $P(h|D, M_i)$ for the two competing models M_α and M_β:

$$\begin{aligned} dB(P(h|D, M_\alpha),\ P(h|D, M_\beta)) = \\ \frac{1}{8} (\mu_\beta - \mu_\alpha)^T \left[\frac{\Sigma_\alpha + \Sigma_\beta}{2}\right]^{-1} (\mu_\beta - \mu_\alpha) + \frac{1}{2} \log\left(\frac{|(\Sigma_\alpha + \Sigma_\beta)/2|}{|\Sigma_\alpha|^{\frac{1}{2}} |\Sigma_\beta|^{\frac{1}{2}}}\right) \end{aligned} \tag{3.45}$$

where since the Bhattacharyya distance is closed at the second moment, $P(h|D, M_i)$ is assumed to follow a multivariate Gaussian distribution $\mathcal{N}(\mu_i, \Sigma_i)$. As for the frequentist scenario, when multiple observables for the models where taken into account, the two options considered were the average of Bhattacharyya distances for the observables:

$$\begin{aligned} &dB_{mean}(P(h|D, M_\alpha),\ P(h|D, M_\beta)) = \\ &\frac{1}{O}\sum_{o=1}^{O} dB(P(h|D, M_\alpha),\ P(h|D, M_\beta)) \end{aligned} \tag{3.46}$$

and the product of Bhattacharyya distances for the observables:

$$\begin{aligned} &dB_{prod}(P(h|D, M_\alpha),\ P(h|D, M_\beta)) = \\ &\prod_{o=1}^{O} dB(P(h|D, M_\alpha),\ P(h|D, M_\beta)) \end{aligned} \tag{3.47}$$

In all instances, a minimum of 400 samples from $P(\theta_i|D, M_i)$ were used to reconstruct $P(h|D, M_i)$. As described in Section 3.2.5 (p. 84) of this chapter, samples came from the fitted Gaussian mixture or were randomly picked from the inference sampled posterior (two-sample Kolmogorov-Smirnov test used to ensure same distribution shape).

In order to compute $dB(P(h|D, M_\alpha),\ P(h|D, M_\beta))$, $dB_{mean}(P(h|D, M_\alpha),\ P(h|D, M_\beta))$ and $dB_{prod}(P(h|D, M_\alpha),\ P(h|D, M_\beta))$ the covariance matrices Σ_i require to be full ranked and positive semi-definite, which was rarely the case due to the high and frequent data sampling set for the simulations. Initially, this was circumvented by sub-sampling the time-series 1:10 (Python implementation). Later on, to avoid the sub-sampling, the different covariance matrices were regularised [340] following:

$$\Sigma_i^R = \Sigma_i + mI \tag{3.48}$$

where m was set to a value of 0.1, a small number compared to the range of the observable (less than 0.1 % of the maximum observable values for the models used) to minimise its effect in the resultant multivariate Gaussian distribution (Julia implementation).

After all considerations, the non-linear optimisation problem of finding the external stimuli u to maximise $dB(P(h|D, M_\alpha), P(h|D, M_\beta))$, $dB_{mean}(P(h|D, M_\alpha), P(h|D, M_\beta))$ or $dB_{prod}(P(h|D, M_\alpha), P(h|D, M_\beta))$ was solved using Bayesian Optimisation [341]. Bayesian Optimisation was used to efficiently explore the design space for u considering the higher computational cost related to model simulations when approximating $P(h|D)$, where each function evaluation could take between 10 seconds to 2 minutes, depending on model complexity. The Python implementation of the method uses a matern covariance function with smoothness parameter $v = 5/2$ for the Bayesian Optimisation Gaussian process and the Upper Confidence Bounds algorithm [342] for the design space sampling selection. The Julia implementation, present in the implemented package BOMBs.jl (see Chapter 8), uses an automatic relevance determination (ARD) squared exponential covariance function for the Bayesian Optimisation Gaussian process and the Expected Improvement algorithm [341] for the design space sampling selection. Unless stated otherwise, the total number of function evaluations was set to 800.

3.2.7.2 Frequentist Optimal Experimental Design for Model Calibration

The aim of the frequentist OED for model calibration strategy used was to optimally design the stimuli profiles u to maximise the information content of experiments regarding model parameters to improve identifiability and potentially reduce experimental work. To quantify the a priori information content for a given experiment, the Fisher Information Matrix was used (equation 3.38, p. 85). The FIM estimates the expectation for a given set of parameters when these are close to the optimal solution [229], [57], hence, providing information about practical identifiability and parameter uncertainty (see Cramér-Rao inequality, equation 3.39, p. 86) [79]. Nonetheless, to be able to use the FIM in an optimisation scheme, a scalar metric needs to be computed. The metrics used in this work were the determinant of the FIM ($D - optimality$ (D-opt)) and the minimum eigenvalue of the FIM ($E - optimality$ (E-opt)) criterion. D-opt seeks to maximise the determinant of the FIM ($argmax\ |FIM|$) while E-opt seeks to

maximise the minimum eigenvalue of the FIM ($argmax\ \lambda_{min}(FIM)$) [57]. Hence, a D-opt design tries to find an experimental scheme that minimises the hyper-ellipsoid volume (confidence region) of the parameter space [57]. On the other hand, an E-opt design searches for the experimental scheme that minimises the highest parameter uncertainty (or error) [81].

The non-linear optimisation problem was solved as a global dynamic optimisation problem using eSS [252] as the meta-heuristic hybrid optimisation solver for the task. Generally (unless stated otherwise), a high number of iterations was used (*e.g.* $6 \cdot 10^5$) to ensure convergence of the solver. Due to experimental constraints in the admissible discretisation of the external inducer values applied to the cells, when required, a Mixed Integer Non-Linear Programming (MINLP) version of eSS was used [343], integrating it into AMIGO2. In this case, the local MINLP solver *nomad* was coupled in the hybrid exploration of the design space. All computations were performed using the Matlab toolbox AMIGO2 [74]. This strategy was used to develop the on-line OED platform described in Section 6.2 (p. 170) of Chapter 6.

3.2.7.3 Bayesian Optimal Experimental Design for Model Inference

Similarly to the frequentist counterpart, Bayesian OED for model inference sought to optimally design the stimuli profile u to maximise the information content of an experiment concerning the identification of model parameters. The strategy exploits the relationship highlighted in the prediction posterior $P(h|D)$ (equation 3.35, p. 84), where the model posterior prediction uncertainty $h(\theta)$ is directly related to the data observed D, the experimental design e_d and the parameter posterior uncertainty $P(\theta|D)$. Note that equation 3.35 can also be formulated to represent a prior prediction distribution:

$$P(h|e_d) = \int P(h|\theta, e_d)P(\theta)\ d\theta \tag{3.49}$$

and hence, allowing a prior prediction uncertainty PU estimation (equations 3.36, p. 84 and 3.37, p. 85). The Bayesian OED strategy adopted in this work aims at maximising a model (prior or posterior) prediction uncertainty ($argmax\ PU_e^C$ or $argmax\ PU_e^E$), highlighting the experiment where parameter uncertainty most

extremely translates into prediction uncertainty. This strategy allows iterative parameter uncertainty reduction parallel to model prediction uncertainty improvement, up to the point where no more information can be gained due to experimental constraints.

The non-linear dynamic optimisation problem was solved using Bayesian Optimisation to efficiently explore the design space for u (ARD squared exponential as kernel and Expected Improvement as sampling algorithm). The methodology was further investigated computationally and experimentally (see Chapter 7). The method was implemented in the Julia Programming Language and a generalised algorithm implementation can be found in the Julia package BOMBs.jl (see Chapter 8).

3.2.8 *E. coli* Culture Growth Conditions and Maintenance

E. coli cells (already transformed with the synthetic plasmid) were stored in 20 % glycerol stock solutions at -80 °C. This cells were used for the results presented in Section 6.2 (p. 170) of Chapter 6 (Bayesian platform) and in Section 7.2 (p. 206) of Chapter 7. When cells were needed, a sample from the frozen stock was streaked into LB plates with $50\mu g/mL$ of Chloramphenicol to obtain isolated colonies. Plates were then incubated at 37 °C overnight ($\approx 14-20$ hours) and subsequently stored at 4 °C for a maximum of one week. Single colonies were then used to grow cells in liquid media (always following 1 to 10 ratio between media and air in sealed vials) containing $50\mu g/mL$ of Chloramphenicol in a shaking incubator at 37 °C. LB media was used for microfluidic experiments and any other microscopy experimental data acquired. M9 minimal medium supplemented with Glucose was used for preliminary tests and assessment of LB media autofluorescence. When needed for experiments, cells were grown overnight ($\approx 14-20$ hours). Maximum concentration of IPTG or aTc was used to ensure that the toggle switch system reached a known steady-state during the overnight growth. Note that, although the acrA and acrB genes were knocked out to prevent adaptation to aTc, through long and constant exposure to high levels of the tetracycline, cells started to show adaptation [344]. Hence, unless specified, IPTG was the inducer used for the overnight growth of cells before an experiment, ensuring cells were in the GFP-TetR dominant state. Additionally, for microfluidic experiments, Tween 20 (0.075 %) was added to the media to prevent cell adhesion to PDMS surfaces.

3.2.8.1 Toggle switch plasmid transformation

To refill the -80 ° *E. coli* stock containing the genetic toggle switch plasmid C, I isolated the plasmid, transformed *E. coli* K-12 BW25113 $\Delta lacY \Delta fliA \Delta acrA \Delta acrB$ cells with it and prepared new glycerol stocks. To purify the plasmid from previously transformed *E. coli* glycerol stocks, I used a Macherey-Nagel NucleoSpin Plasmid mini kit following the manufacturer's protocol and stored the purified plasmid vials at 4 °C.

Prior to transformation, chemically competent cells were prepared using a standard protocol with calcium chloride ($CaCl_2$). First I inoculated cells into 5 mL of LB media and grew cells overnight in a shaking incubator (200 rpm) at 37 ° C. Next day a 1 to 100 dilution of cells to a final volume of 100 mL was prepared letting cells grow in a shaking incubator at 37 ° C until they reached mid log-phase (OD_{600} = 0.4-0.6). Then I centrifuged the media containing cells at 4500 rpm for 5 minutes and at 4 ° C (previously pre-cooled), discarded the supernatant, re-suspended cells into 20 mL of ice cold $CaCl_2$ 100 mM and incubated cells on ice for 1 hour. Then I centrifuged the media containing cells under the same conditions previously stated, discarded the supernatant, re-suspended cells into 2 mL of ice cold $CaCl_2$ 85 mM containing 15 % glycerol, aliquot 50 μL of the media containing cells in pre-cooled 1.5 mL tubes and stored them at -80 ° C.

Cells where then transformed with the plasmid by heat shock. First, a tube from the glycerol stock containing 50 μL of chemically competent cells was thawed on ice for 20 minutes. Next, 1 μL of mini prep purified plasmid was added to the media containing cells and the mix was incubated in ice for 30 minutes. Then, the mix was heat shocked at 42 ° C for 90 seconds followed by an incubation in ice for 3 minutes. After this, 0.5 mL of LB media were added and incubated in a shaking incubator at 37 ° C and 200 rpm for 1 hour. Finally, 100 μL of the media containing the transformed cells was plated on LB agar plates containing 50 $\mu g/mL$ of Chloramphenicol and incubated overnight at 37 ° C. Next day, 5 mL of LB media containing 50 $\mu g/mL$ of Chloramphenicol was prepared introducing a colony from the plate to grew overnight in a shaking incubator at 37 ° C and 200 rpm. Finally, aliquots were prepared in 2 mL tubes containing the incubated cells with 20 % glycerol and stored at -80 ° C.

3.2.9 *S. cerevisiae* Culture Growth Conditions and Maintenance

S. cerevisiae cells were stored in 20 % glycerol stock solutions at -80 °C. This cells were used for the results presented in Section 6.2 (p. 170) of Chapter 6 (frequentist platform). When cells were needed, a sample from the frozen stock was streaked into SC plates to obtain isolated colonies. Plates were then incubated at 30 °C for a maximum of 2 days and subsequently stored at 4 °C for a maximum of one week. Single colonies were then used to grow cells in liquid SC media (always following 1 to 10 ratio between media and air in sealed vials) in a shaking incubator at 30 °C. When needed for experiments, cells were grown overnight ($\approx 14 - 20$ hours). Unless specified, no IPTG was used for overnight growth resulting in no induction of the inducible promoter by the beginning of an experiment.

3.2.10 Fluorescence Microscopy

All microscopy work (see Chapter 6 and Section 7.2, p. 206 of Chapter 7) was done with an Nikon Eclipse Ti inverted microscope (Fig. 3.1a). For differential interference contrast (DIC) microscopy imaging, the in-build D (Diffusion Filter to equalise illumination) and PFS (blocks the infra-red spectrum used by the perfect focus system to avoid interference) filters were used. For fluorescence imaging the Neutral Density ND4 in-build filter (reduces light intensity to 25 % transmittance) was used to avoid photo-bleaching and reduce cell exposure to light. The fluorescence filter (from Chroma technology corp.) equipped in the microscope were the 49000 ET-DAPI (Ex: 350/50 , Em: 460/50), 49002 ET-GFP(FITC/Cy2) (Ex: 470/40, Em: 525/50), 49905 ET-514nm Laser Bandpass Set for EYFP, Venus, Citrine, Fluo3 (Ex:514/10 , Em: 545/40), 49909 ET-561nm Laser Bandpass Set for TagRFP, tdTomato, AlexaFluor 546, TRITC (Ex: 561/10, Em: 600/50), 49006 ET-Cy5 (Ex: 620/60, Em: 700/75) and 69308 ET-ECFP/EYFP/mCherry with single band exciters (Ex: 430/24, 500/20, 557/25, Em: 470/10, 535/15, 635/35). The FITC filter (49002) was used to image mEGFP (GFP), the Citrine filter (49905) was used to image Citrine, the TagRFP filter (49909) was used to image mKate2 (RFP) and the Cy5 filter (49006) was used to image Sulfo-Cyanine5 Carboxylic Acid. The light source for fluorescent imaging was provided by the

Lumencor Spectra X Light Engine with 6 independently controlled LED light sources. The light sources available were violet (395/25), blue (440/20), cyan (470/24), teal (510/25), yellow (575/25) and red (640/30). Temperature control of samples over experiments was performed by an Okolab H101-Basic-BL system. All microscope images were acquired with the ANDOR EMCCD camera iXon Ultra 888. The camera had a sensor format of 1024 $\times$ 1024 pixels (with a pixel size of 13 μm and 16 bit type). The image amplification used in all cases was electron multiplying with a gain equal to 20, read-out mode equal to 1 MHz (10 MHz was used to ease sample preparations), pre-amp gain 1 and binning equal to 1. The camera sensor was stabilised and maintained at -70 $^\circ$ C. The microscope objectives used were a 10X Plan Fluor Ph 1 DLL objective for general use and sample preparation, a 40X Plan Fluor Oil DIC H N2 objective for yeast image acquisition and a 100X Plan Fluor Oil DIC H N2 objective for bacteria image acquisition. Microscope stage and image acquainting during experiments was automated using μmanager (NIS-Elements AR was used only for checks).

3.2.11 Microfluidic Experiments for Model-Based Bio-Systems Identification

Microfluidics was used to generate dynamic data to infer model parameters (see Section 6.2, p. 170 of Chapter 6 (Bayesian platform) and in Section 7.2, p. 206 of Chapter 7). The approach was used due to its ability to precisely control media and inducers administered to the cells over time and the easy acquisition of microscopy images of mono-layers of cells contained in the different device traps [21]. For experiments regarding the synthetic inducible promoter developed in *S. cerevisiae*, the microfluidic chip described in Ferry *et al.* 2011 [170] was used. A schematic of the chip can be seen in Fig. 3.2a, indicating the different inlets, channel architectures (to allow mixing of media from two ports) and the trap where cells will grow in a single layer. For experiments regarding the synthetic toggle switch developed in *E. coli*, the microfluidic chip described in Baumgart *et al.* 2017 [184] was used incorporating a Dial-A-Wave (DAW) junction to allow the mixing of media from two different inlets. A schematic of the chip can be seen in Fig. 3.2b, also indicating inlets, media mixing architectures and cell traps. Fig. 3.1b shows a microfluidic chip attached to the microscope stage with all the tubing connected, ready for an experiment.

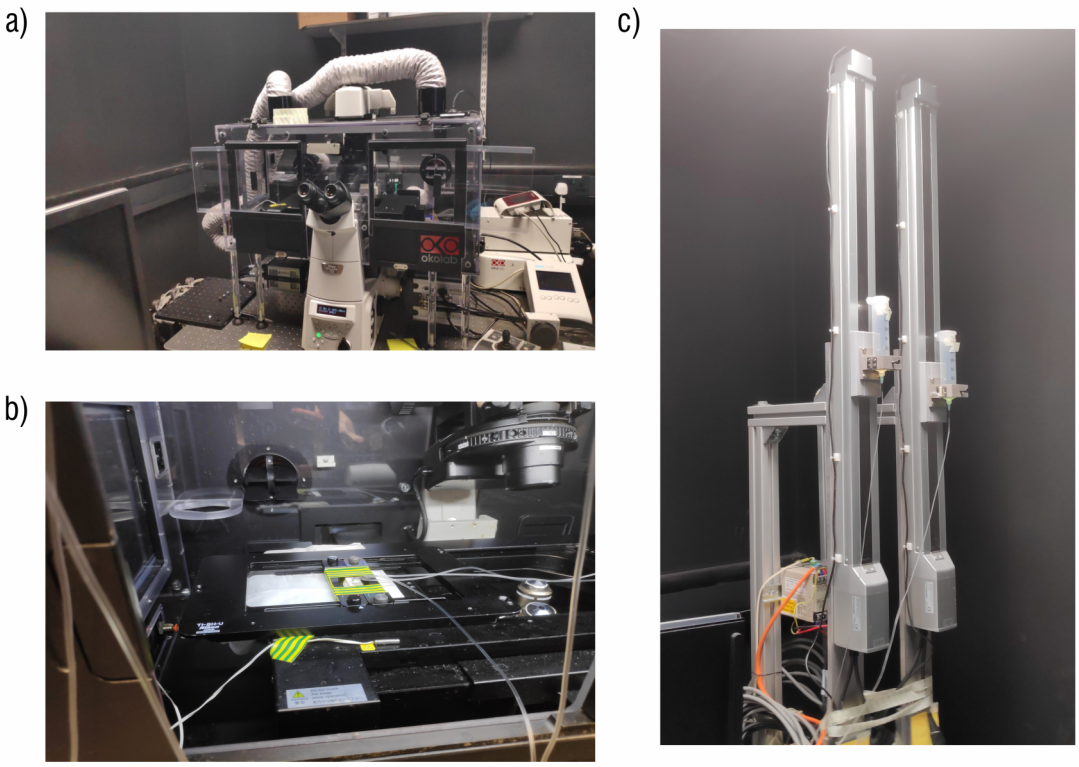


Figure 3.1: Experimental imaging set-up for microfluidic experiments. a) Image of the Nikon Eclipse Ti inverted microscope used for imaging during microfluidic experiments. The chamber around the stage and the temperature control system (Okolab H101-Basic-BL) allowed for constant temperature control of the chamber during experiments. b) Interior of the microscope chamber with a microfluidic chip attached to the stage. c) Image of the linear actuators used to control the height of the two syringes that administer the external inducers into the microfluidic chips.

3.2.11.1 Microfluidic Devices Preparation

First, a patterned wafer of the two chip designs (Fig. 3.2) was made using photolithography [170], creating a master mould made of photoresist layers with a negative image of the different features of the chips. Once this was obtained, using soft lithography, a silicon rubber mould made of polydimethylsiloxane (PDMS) was prepared every time chips were needed (Fig. 2.3, p. 18 of Chapter 2). Each master mould contained multiple chips. To prepare the PDMS moulds the curing agent and base had to be mixed in a ratio of 1:9. The total mass of the mixture can be calculated as:

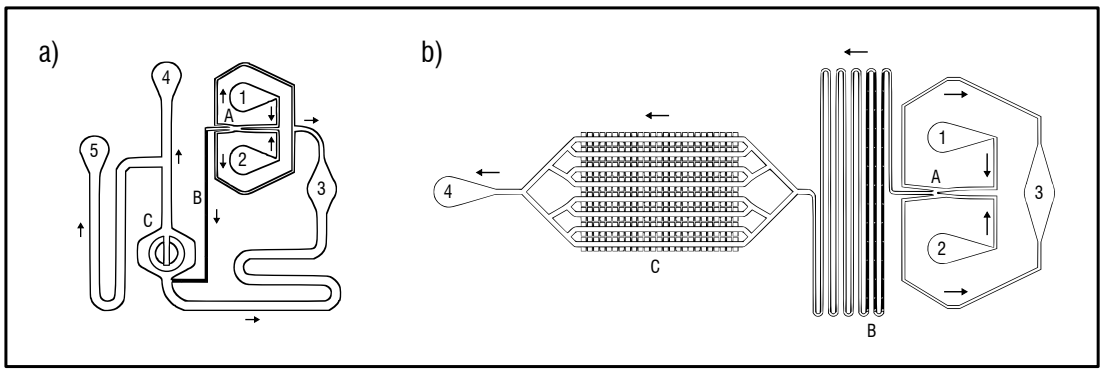


Figure 3.2: Schematics of the microfluidic devices used in this work. a) Structure of the yeast device chip (MFD005$_a$) architecture. Numbers indicate the two input ports (1 and 2), waste ports (3,4,5) and cell loading port (5). Letters indicate the Dial-A-Wave (DAW) junction (A) used to adjust the mixing ratio between media coming from port 1 and 2, the staggered herringbone mixer channel (B) used to adequately mix the two fronts of media coming from the DAW junction, and the cell trap region (C) used to trap a mono-layer of cells for microscope imaging. Arrows indicate the flow direction during an experiment. b) Structure of the bacteria device chip architecture. Numbers indicate the two input ports (1 and 2), waste ports (3,4) and cell loading port (4). Similarly, with the yeast chip, letters indicate the DAW junction (A), staggered herringbone mixer channel (B) and the multiple cell traps (C) represented by the 400 different squares emerging from the channels. Arrows also indicate the flow direction during an experiment.

$$M_{PDMS} = \rho V = \rho \frac{\pi}{4} d^2 h \approx \frac{19}{44} d^2 \tag{3.50}$$

where chip thickens h was set to 0.5 cm, the density ρ of the mixture was 1.1 $g \cdot cm^{-3}$ and the diameter d of the section of interest for the master mould was 9 cm. Considering the need of having a ratio of 1:9 between the curing agent and the PDMS base, the final mass of the curing agent was $\frac{19}{440}9^2 = 3.5\ g$, and the one for the base was $\frac{171}{440}9^2 = 31.5\ g$. The mixture was then carefully poured into the master mould and, any air bubbles in the mix were raised to the top and removed using a degassing dissector. Note that the master mould with PDMS cannot be maintained in vacuum for more than 15 consecutive minutes but, the process can be repeated multiple times. Once all air bubbles were removed from the PDMS, the master mould was placed into an oven at $65°C$ for 3 hours for curing. Next, the PDMS was cut and removed from the master mould, individual

chips cut and, ports punched (1 mm biopsy punch for yeast devices and 0.75 mm biopsy punch for bacteria devices) manually. Finally, chips were cleaned using tape and bonded into clean coverslips using a vacuum plasma cleaner. For the plasma to act on the side of the chip containing the features, these must be facing up inside the plasma cleaner. After 30 s of plasma exposure at a high RF level of chips and coverslips, chips were quickly placed into the coverslip surface for bonding and finally left into a $65°C$ oven for 15 min.

3.2.11.2 Microfluidic Experiments for the Synthetic Toggle Switch – *E. coli*

Experiments (see Section 6.2, p. 170 of Chapter 6 describing the Bayesian platform and Section 7.2, p. 206 of Chapter 7) were performed using the microfluidic device described in Baumgart *et al.* 2017 [184]. A schematic of the chip features can be seen in Fig. 3.2b. Cell media (LB with Chloramphenicol $50\mu g/mL$) was administered into the device using hydrostatic pressure through syringes placed at specific heights. A set of two linear actuators controlled (Fig. 3.1c) the syringe containing the chemical inducers IPTG and/or aTc plus fresh LB media. Each inducer was added to one of the syringes attached to the linear actuators. In this work, due to experimental constraints, I only considered the presence of one inducer at a time during experiments. The Matlab MODBUS communication interface was used to automate the movement of the linear actuators to set the presence of a determined inducer. Alternatively, the communication interface could be used to set a determined media mixing ratio where only one inducer was used for the experiment (absence of the other inducer in the second input media syringe). The mixing ratio was tracked by including sulfo-cy5 carboxylic acid (0.5 μg/mL) into one of the syringes. Cells were streaked at least once a week if needed for experiments as described in Section 3.2.8 (p. 93) of this chapter.

For an experiment, cells were set to grow overnight the day before with maximum IPTG (1 mM) to obtain them in the GFP-TetR dominant steady-state. On the day of the experiment, a 1:60 dilution of the cells was done (in a final volume of 30 mL). Cells were then incubated at 37 °C for 2 hours to reach exponential phase ($0.1 \leq O.D._{600} \leq 0.3$) for the experiment. In the meantime, two syringes were loaded with 10 mL of LB media with Chloramphenicol $50\mu g/mL$ (syringes 3 and 4). Syringes 1 and 2 were loaded with 15 mL of media with Chloramphenicol $50\mu g/mL$ plus IPTG or aTc. Tween 20 at 0.075 % was added to all the media

used for the experiment and the overnight growth. The fluorescent dye sulfo-cy5 carboxylic acid (0.5 μg/mL) was added into syringe 1. The next step was the visual inspection of the devices (discard ones with collapsed channels, debris or other imperfections) and wetting of the chip. The order followed to wet the chips were ports 3, 4, 2, and 1 (Fig. 3.2b), ensuring that no air bubbles or debris were present anywhere in the device. Once wetting was finished, each syringe was connected to the chip following the same order as during wetting. Syringes 1 and 2 were attached to the linear actuators to control their hydrostatic pressure (with an allowed height range between 40.72 cm and 44.36 cm above the stage). Syringes 3 and 4 were attached at 26.5 and 24.5 cm above the microscope stage, respectively, considering the interface between media and air as the measuring point.

After ensuring the absence of air bubbles and debris in the chip, the following step was setting manually in the Matlab MODBUS communication interface the syringe heights that ensured that all the media from syringe 1 or syringe 2 went to the cell traps (no media mixing). If needed (e.g. varying concentrations of one inducer during an experiment), we calibrated the mixing ratio obtained in the DAW junction (Fig. 3.2b, section A) concerning the different complementary heights of the linear actuators. Next, or while calibration was proceeding, the overnight cultures $O.D._{600}$ was checked. Cells were then centrifuged at 3400 rpm for three minutes and the pellet was resuspended in 2 mL of fresh media. The concentrated cell culture was then added into a fifth syringe to allow cell loading into the device.

The final step before starting the experiment was the cell loading into the chip. The cell syringe was connected to the cell loading port (port 4). The syringe height was then increased (hence its flow) to allow cells to move towards the cell traps. This syringe needed to stay lower than syringes 1 and 2 to redirect any excess cells towards the waste port 3, avoiding contamination of the media ports 1 and 2. Once a high volume of cells was located in the main channels ($30\mu m$ of height) next to the cell traps ($1.6\mu m$ of height), the cell syringe was slightly lowered to retain the cells in the region and allow them to enter the cell traps over time (Fig. 3.2b, section C). Once enough cells entered the traps (cells present in at least one-third of at least eight cell traps, corresponding to $\approx$ 60-120 cells per trap), the cell syringe was lowered to wash away any excess cells from the main channels. Fig. 3.3a shows a DIC image of one cell trap after the cell

loading. The syringe was then replaced with a new one containing fresh media and set to the predefined height. The experiment was then ready to run after a final inspection of the device channels (to identify and remove air bubbles or debris introduced during cell loading). However, a wait of 10 hours before the experiment started was in place to allow cells to recover and grow to occupy the entirety of the traps.

The microscope chamber was maintained at 37 °C for the full length of the experiment. The same software controlling the image acquisition was in charge to control the linear actuators to administer the desired input profile over time. All experiments were set to last 38 hours with a sampling time of 5 min (except for the GFP channel which, was every 10 min to reduce aTc photodegradation). Fig. 3.3b shows a DIC image of a cell trap at the last time-frame of an experiment. Microscopy image acquisition was performed with a 100X Plan Fluor Oil DIC H N2 objective. DIC images (exposure time of 30 ms) were acquired to allow cell segmentation during image processing. Cellular mKate2 was measured using the 449909 ET-561nm Laser Bandpass Set for TagRFP, tdTomato, AlexaFluor 546, TRITC filter (exposure time of 250 ms). Cellular mEGFP was measured using the 49002 ET-GFP(FITC/Cy2) filter (exposure time of 30 ms). Tracking of adequate inducer presence or mixing ratio was done by also acquiring images using the 49006 ET-Cy5 filter (exposure time of 50 ms) to track the dye concentration over time. All experiments where performed at least twice (two biological replicates) and the final experimental data includes the average cell fluorescence for three different cell traps (three technical replicates).

3.2.11.3 Microfluidic Experiments for the Synthetic Inducible Promoter – *S. cerevisiae*

Experiments (see Section 6.2, p. 170 of Chapter 6 describing the frequentist platform) of Chapter 7) were performed using the microfluidic device described in Ferry *et al.* 2011 [170]. A schematic of the chip features can be seen in Fig. 3.2a. Cell media (SC) was administered into the device using hydrostatic pressure through syringes placed at specific heights. A set of two linear actuators (Fig. 3.1c) controlled the syringe containing the chemical inducer IPTG in SC media and a second one with only SC media to adjust the mixing ratio between the two. The Matlab MODBUS communication interface was used to automate the movement of the linear actuators to achieve specific mixing ratios. The mixing ratio was tracked by including sulfo-cy5 carboxylic acid (1 μg/mL) into the

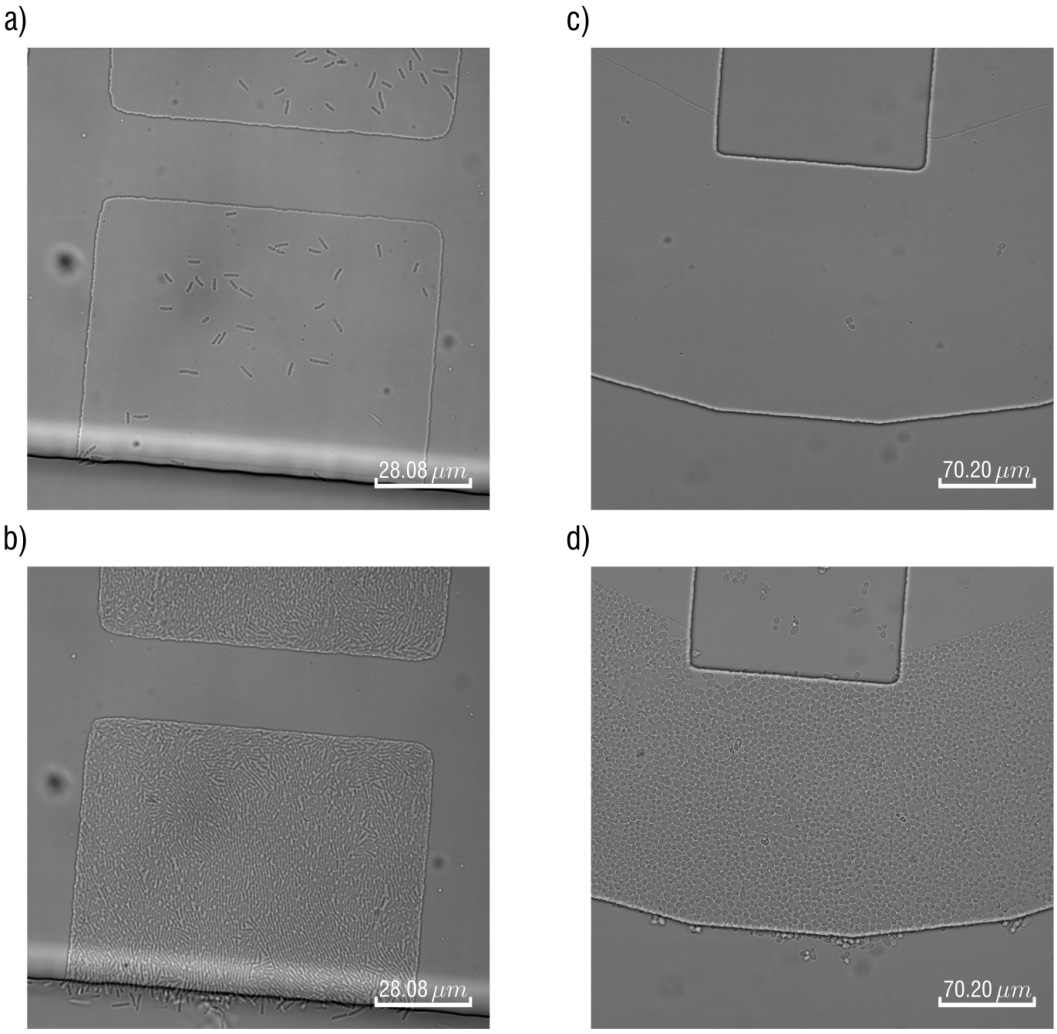


Figure 3.3: DIC microscopy images from bacteria and yeast microfluidic experiments. DIC image of a trap from the bacteria microfluidic device containing a low density (a) and the trap full (b) of cells at a 100X magnification. The main device channel from where cells are loaded is located at the bottom of the images. DIC image of the main trap from the yeast microfluidic device (MFD005$_a$) containing three budding cells (c) and the trap full of them (d) at a 40X magnification. The main device channel from where cells are loaded is located at the top of the images.

syringe containing the inducer IPTG. Cells were streaked at least once a week if needed for experiments as described in Section 3.2.9 (p. 95) of this chapter.

For an experiment, cells were set to grow overnight the day before without IPTG. Next, three syringes were loaded with 10 mL of SC media (syringes 2, 3, 4) and one with 5 mL of SC media (syringe 5). The fifth one was loaded with SC media containing the maximum concentration of IPTG used (100 μM, unless specified otherwise) and the fluorescent dye sulfo-cy5 carboxylic acid (1 μg/mL) in a final volume of 10 mL (syringe 1). The next step was the visual inspection of the devices (discard ones with collapsed channels, debris or other imperfections) and wetting of the chip. The order followed to wet the chips were ports 5, 4, 3, 2, and 1 (Fig. 3.2a), ensuring that no air bubbles were present anywhere in the device. Once wetting was finished, each syringe was connected to the chip following the same order (5 to 1). Syringes 1 and 2 were attached to the linear actuators to control their hydrostatic pressure (with an allowed height range between 38.47 cm and 56.61 cm above the stage). Syringes 3, 4 and 5 were attached at 18, 23 and 23 cm above the microscope stage, respectively, considering the interface between media and air as the measuring point.

After ensuring the absence of air bubbles and debris in the chip, the following step was calibrating the mixing ratio obtained in the DAW junction (Fig. 3.2a, section A) concerning the different complementary heights of the linear actuators. While calibration was proceeding, a sixth syringe was prepared with 5 mL of SC media plus cells at exponential phase ($0.2 \leq O.D._{600} \leq 0.3$) from the overnight cultures. The overnight cultures $O.D._{600}$ was first checked and, if outside range, diluted accordingly.

The final step before starting the experiment was the cell loading into the chip. The cell syringe was connected to the cell loading port (port 5). The syringe height was then increased (hence its flow) to allow cells to move towards the cell trap. This syringe needed to stay lower than syringes 1 and 2 to redirect any excess cells towards the waste port 3, avoiding contamination of the media ports 1 and 2. At an adequate cell flow rate, some cells entered the central directional channel ($10\mu m$ of height) towards the circular cell trap ($3.525\mu m$ of height) represented in Fig. 3.2a, section C. Once one to five budding cells were contained in the cell trap, the cell syringe was disconnected and replaced by one containing fresh medium. Fig. 3.3c shows a DIC image of the main cell trap containing

three budding cells after cell loading. The experiment was then ready to run after a final inspection of the device channels (to identify and remove air bubbles or debris introduced during cell loading).

The microscope chamber was maintained at 30 °C for the full length of the experiment. The same software controlling the image acquisition was in charge to control the linear actuators to administer the desired input profile over time. All experiments were set to last 24 hours with a sampling time of 5 min. Fig. 3.3d shows a DIC image of the main cell trap at the last time-frame for an experiment. Microscopy image acquisition was performed with a 40X Plan Fluor Oil DIC H N2 objective. DIC images (exposure time of 30 ms) were acquired to allow cell segmentation during image processing. Cell citrine was measured using the 49905 ET-514nm Laser Bandpass Set for EYFP, Venus, Citrine, Fluo3 filter (exposure time of 130 ms). Tracking of adequate inducer mixing ratio was done by acquiring images using the 49006 ET-Cy5 filter (exposure time of 50 ms) to track the dye concentration over time. All experiments where performed at least twice (two biological replicates).

3.2.11.4 Calibration of the DAW junction

Calibration of the DAW was required to relate specific mixing ratios for media coming from syringes 1 and 2 to specific height positions for these [170]. Section 6.1.1 (p. 158) of Chapter 6 shows further details and experimental results. This procedure had to be done for each experiment since a new chip, syringes and tubing were used each time. For this, the actuator syringe containing the fluorescent dye (syringe 1) was progressively increased in small steps between a minimum (0 %) and maximum (100 %) mixing ratio established visually. The second actuator syringe was hence simultaneously decreasing height to change the mixing ratios. After each movement of the actuators, images of the fluorescent dye were taken in the aperture of the staggered herringbone mixing channel (Fig. 3.2a,b, section B).

The increasing pressure process was replicated three sequential times and automated using the Matlab MODBUS communication interface. Images were processed in Matlab, extracting the average sulfo-cy5 carboxylic acid fluorescence signal for each position of the actuators. Next, the extracted data was normalised by the mean maximum background signal and the mean minimum highest signal (100 % of syringe 1). Once the data was processed, the automated software fitted

polynomials of orders 1, 5, 9 and 14, linking the different actuator heights to mixing ratios. The user then was able to select the best fit according to visual inspection of the fits (to avoid clear over-fitting) and the goodness of fit metrics SSE (sum squared error), RMSE (root mean square error) and r^2.

3.2.11.5 Image Processing for Microscopy Images from Microfluidic Experiments

Custom code was written in Matlab to build a platform to process fluorescence microscopy images in real-time during experiments to extract the average pixel fluorescence signals from *E. coli* or *S. cerevisiae* cells. Section 6.1 (p. 158) of Chapter 6 shows further details and experimental results of the image processing used in this thesis. The first step was for the user to select the region of interest (ROI) where cells would grow (cell trap excluding edges to avoid segmentation errors). For experiments using the bacteria device, three different traps could be selected to be processed as the experiment was undergoing. While a higher number of traps could be processed, this would introduce delays. The next step for the user was to introduce information about the inducers on the syringes (metadata). The automated code gathered any other necessary information from the microscope platform to generate any necessary files (*e.g.* files that contained the fluorescent results, input profiles or cell counts). All the rest of the image processing steps were automated and independent from user integration.

The automated code first cut the ROIs from each image acquired (DIC and fluorescent images) and saved them as separate files to ease image segmentation and processing. Next, any necessary images were segmented using the convolutional neural network (CNN) U-Net [345]. The CNN could run in a remote server by using an ImageJ plug-in [346] or locally by using TensorFlow (Python) [347]. In experiments for the inducible promoter (*S. cerevisiae*), only DIC images were segmented to, later on, extract the average citrine fluorescence signal. In experiments for the genetic toggle (*E. coli*), images from DIC and the red (mKate2) and green (mEGFP) fluorescent channels were segmented. The additional image segmentations were required due to cell movement between image acquisitions when traps were not filled with cells, resulting in DIC images being imprecise. The masks obtained by segmenting images from the different fluorescent channels were used to extract the background fluorescence signals at the beginning of the experiments (not accounted for the final data). The strategy

was possible thanks to cell fluorescence always being present to some extent independently of the concentration of the inducer (basal transcription). When the traps started to get full of cells, the segmentation quality on fluorescence channels decreased drastically. At that point, segmentation switched to only DIC images since cell movement was no longer an issue due to cell density in the traps. The average centroid distance between the k-nearest neighbours (with k=*3* cells) was used to assess the cell density of the traps at each sampling time and hence, automate the switch between segmentation channels. When the average distance between centroids was lower than a threshold (27 pixels), segmentation for the position switched only to DIC images.

After segmentation, the resultant segmentation masks had to be automatically post-processed to ease data extraction. First, each mask element (cells) was indexed with a unique identifier since U-Net outputs were binary masks (zero indicated background, one indicated cells). The binary masks were also used to generate background masks by morphological dilating each of the elements (to avoid any segmentation errors on the cell edges or point spread function effects) and by inverting the logical values of the binary mask. The background masks were first used to extract the fluorescent dye (sulfo-cy5 carboxylic acid) signal from the images to track the DAW junction mixing ratio. Alternatively, the user could select to acquire frames from the same region where the mixing calibration was performed to ease computations and avoid losing signal confidence when cell traps were full.

The final step in the image processing was to extract the average cell fluorescence signals from the different acquired images. First, background masks were used to compute the background fluorescence signal from any of the fluorescence channels used as long as cells occupied less than 40 % of the ROI. When cells were present in more than 40 % of the ROI, the average background signal computed from the past time points was used. Alternatively, the user could select to acquire a different chip position to calculate the background signal to prolong the time where cell occupancy was low. The optional new position still had to be inside the cell trap due to the different device sections height and the potential media autofluorescence in some acquisition channels. The processed background signal Bg_j for the current time frame j was used to compute the average fluorescence $cX_{i,j}$ for each cell i as:

$$cX_{i,j} = \frac{1}{P_{i,j}} \sum_{p}^{P_{i,j}} (\chi_{i,j,p} - Bg_j) \tag{3.51}$$

where $P_{i,j}$ and $\chi_{i,j,p}$ indicate the number of pixels and the pixel p fluorescence signal from cell i in frame j respectively. The mean and standard deviation of $cX_{i,j}$ for all the cells present in the frame was computed to obtain the average cell fluorescence mX_j and standard deviation sX_j for the time frame processed. Cell size (in pixels) and $cX_{i,j}$ was also stored to ease posterior data analysis. An HTML file containing fluorescence plots (cells and dye), and the last frame for each position and channel acquired, was generated for remote tracking of the experiments progress. Images displayed in the HTML file were updated after each processed time frame. The web page was then hosted in GitHub pages for public remote access.

3.2.11.6 *In Vivo* On-Line Optimal Experimental Design

In vivo on-line OED was achieved thanks to continuous interaction between the image processing platform developed (Section 3.2.11.5) and the one controlling the movement of the linear actuators, combined with the integration of a package to perform OED (Matlab toolbox AMIGO2 [74] or the developed Julia package BOMBs.jl described in Chapter 8) at the same time. Before starting the experiment, the number of sub-experiments was selected, and, if desired, a discretisation level as defined as percentage for the optimisation. The minimum and maximum inducer concentration contained in the syringe was also introduced. The discretisation level selected followed the confidence in mixing ratios when using a Kolmogorov–Smirnov (KS) test to compare the distribution of fluorescence signal between the different actuator positions across nine different calibrations. Discretisation during optimisations in AMIGO2 [74] was achieved by coupling a MINLP version of eSS from the toolbox MEIGO [343]. Discretisation during optimisations in BOMBs.jl was achieved by rounding the inputs in the model simulation since no gradient-based scheme was used during the optimisation. For frequentist on-line OED using AMIGO2 [74], it was also possible to select the optimiser for parameter estimation (eSS as the default one, differential evolution or SRES) and OED (eSS as the default one, genetic algorithm, NOMAD and VNS). For parameter estimation, the type of cost function could also be selected (log-likelihood as the default one or least squares)

as well as the maximum number of iterations (default set as $6 \cdot 10^6$) and time allowed for the optimiser (default set as 45 min). For OED, the FIM metric to use for the optimisation could also be selected (D-opt as default or E-opt) as well as the maximum number of iterations (default set as $6 \cdot 10^6$) and time allowed for the optimiser (default set as 2 hours). For the Bayesian on-line OED using BOMBs.jl, it was also possible to select the maximum time (default set as 50 min) and function evaluations (default set as 10^7) for a pre-inference MLE step to start inferences with a set of good initial guesses (as described in Section 3.2.2.2, p. 77 of this chapter). The number of inference iterations could also be selected (default set at $4 \cdot 10^2$) and the maximum number of iterations (default set at $1 \cdot 10^3$) for OED. Finally, the type of utility function ($argmax\ PU_e^C$ set as default or $argmax\ PU_e^E$) and the number of posterior sub-samples (default set at $4 \cdot 10^2$) to use for OED to increase the computational speed of the optimisation (as described in Section 3.2.5, p. 84 of this chapter) were selected. In both schemes, a file containing information about the mathematical model to be used, steady-state and parameters were introduced in the form of Matlab functions (frequentist) or Julia dictionaries (Bayesian). In the frequentist platform, a file containing information about past experiments could also be given.

Both strategies (frequentist and Bayesian) for *in vivo* on-line OED followed the same general structure. First, a time vector for each experiment sampling was defined and divided into as many equally spaced fragments as sub-experiments were specified. Then, with the information introduced about maximum times and iterations, the total time allocated to parameter inference and OED was calculated. This delay time was then used to determine when the scheme should start, and hence the number of data points needed to be processed. Since in the Bayesian strategy, the time allocated to a Stan inference could not be worked out in advance, it was assumed to take a maximum time of 3 hours using the default settings. However, this time would depend on the model used and the number of parameters to estimate. Hence, it must be adjusted for each particular scenario. The code accounted for the inference to last longer than the time allocated, allowing a shorter optimisation of the rest of the sub-experiment if time was left in the process. At this point, the result files generated by the image processing platform were checked with the same frequency as the sampling time selected for the experiment. Once enough data for the sub-experiment was processed, the frequentist parameter estimation or Bayesian inference were triggered. Frequentist parameter estimation was performed as described in Section 3.2.2.1 (p. 76) of

this chapter, using all the data available up to the trigger point (including past experiments introduced in the platform) and repeating the process in parallel for as many cores (34 in this work) as the computer running it had available. Bayesian inference (with prior MLE) was performed as described in Section 3.2.2.2 (p. 77) of this chapter, dividing the total number of samples selected into four parallel chains. Once parameter estimation or Bayesian inference finished, OED began to define the optimum inducer concentration to administer to the cells at each period established as a step (3 hours unless stated otherwise) for the following sub-experiment. Frequentist OED was performed as defined in Section 3.2.7.2 (p. 91) of this chapter with the selected settings and Bayesian OED as defined in Section 3.2.7.3 (p. 92) of this chapter. Finally, once the optimisation finished, the file indicating the position of the actuators was updated with the new input profile, allowing the experiment to follow the new optimal input design. Section 6.2 (p. 170) of Chapter 6 describes the results of an *in vivo* experimental implementation of the described methodology.

3.3 Chapter Conclusions

An interdisciplinary approach is critical for the widespread and efficient incorporation of mathematical models in biological sciences. In this chapter, I described all the materials and methodologies required to replicate the computational and experimental results shown in the following chapters of this thesis. First, I presented all the necessary laboratory equipment (including the type of cells and media used) and computational software and data used throughout this work. Then, I described in extensive detail how to use these materials to generate the results described in Chapters 4, 5, 6, 7 and 8. I started the section introducing the mathematical models used to describe four different biological systems (i.e. genetic toggle switch, orthogonal inducible promoter, Repressilator and the Hes1 oscillator) and any constraints and details associated with them (e.g. prior distributions used for Bayesian inference) and their simulations (i.e. event inputs implementation). Next, I dived into the strategies used for parameter estimation (frequentist and Bayesian), assessment of goodness of fit and quantification of experimental information for the task before (*a priori*) and after (*a posteriori*) observing the data. Then, I focus on describing the concept of optimal experimental design and how I used it for the problems of model selection and parameter inference. After these mathematical and computational methodologies get presented, I followed with the description of the experimental methods needed to replicate the experimental data presented in subsequent chapters. These methods range from cell culturing (bacteria and yeast) and fluorescence microscopy settings used, to the protocols needed to perform microfluidic experiments. I also described procedures to process the large amount of data obtained from images and the incorporation of optimal experimental design within the experimental platform. Hence, this chapter is a central element to replicating and ultimately understanding the different computational and experimental results presented in the subsequent chapters.

Chapter 4

Tools for Bayesian Parameter Inference of Models Describing Biological Systems: Uncertainty and Information Quantification of Experiments

Thanks to technological advancements, Bayesian probabilistic approaches to mathematical modelling are increasingly gaining momentum in the scientific community. These methodologies are also attaining popularity in biological sciences, with an increasing number of research groups using Bayesian inference to estimate model parameters. In this chapter, I will implement some of the tools and methodologies defined in Chapter 3 (Sections 3.2.1, 3.2.2, 3.2.3 and 3.2.5) to infer parameters of mathematical models of high interest in synthetic biology (Section 4.1). After establishing the procedure, I will focus on analysing the quality of the obtained posteriors and the predictive capabilities of the updated models. Then, I will go deeper into understanding how these posteriors tell us the amount of information gained after seeing different data sets to highlight the fact that different experiments contain varying degrees of information for a specific problem (Section 4.2). Yet, this information is far more valuable before generating the usually expensive and laborious data. For this reason, I will investigate the relationship between model predictions and parameter uncertainties to finally show that the former can be used to improve the latter (Section 4.3).

4.1 Bayesian Inference of Deterministic Models for Biological Systems

Bayesian statistics can be used to infer parameters of biological models, relaxing noise assumptions of estimates (e.g. by deviating from normality) and providing a better understanding of the behaviour of the system under study. In this section, I will describe how I used Bayesian statistics (through the probabilistic programming language Stan [83]) to infer parameters of well-established models in systems biology; the repressilator [10] and the Hes1 transcription factor [325]. I then extended the inference scheme to include inducible systems, such as an orthogonal inducible promoter [86], [228] or a genetic toggle switch [85] due to its high interest in synthetic biology [8]. Finally, I inferred three mathematical models of the genetic toggle switch on *in vivo* data retrieved from Lugagne *et al.* 2017 [85] and analysed the parameter posteriors and their predictive capabilities.

4.1.1 Bayesian Inference of the Repressilator and Hes1 Models

I successfully employed Bayesian statistics to infer two mathematical models for two biological oscillatory systems; the repressilator and the Hes1 transcription factor. The repressilator is a synthetic system originally developed in *Escherichia coli* composed of three genes whose product represses the transcription of the following gene in a cycle (see a schematic of the system in Fig. 4.1a). The Hes1 system describes the dynamics of the transcription factor Hes1 that, once translocated into the nucleus, represses its own transcription (see a schematic of the system in Fig. 4.1d). The mathematical models used for the two circuits are two sets of ordinary differential equations (ODEs) retrieved from Liepe *et al.* 2013 [56] (see Section 3.2.1.3, p. 71 of Chapter 3 for details of the models and constraints used).

For the repressilator, I considered the same five experimental conditions as defined in Liepe *et al.* 2013 [56]: the original system oscillatory progress (set 1), a 10-fold decrease of the transcription rate α for all genes (set 2), a 10-fold decrease of α for all genes and a 5-fold decrease of the basal transcription rate α_0 for gene two (set 3), a 10-fold decrease of α for gene one (set 4) and a 10-fold decrease of α for all genes and the ratio of protein to mRNA decay rate β for gene two (set 5). For the pseudo-data generation of all sets, I only considered the protein

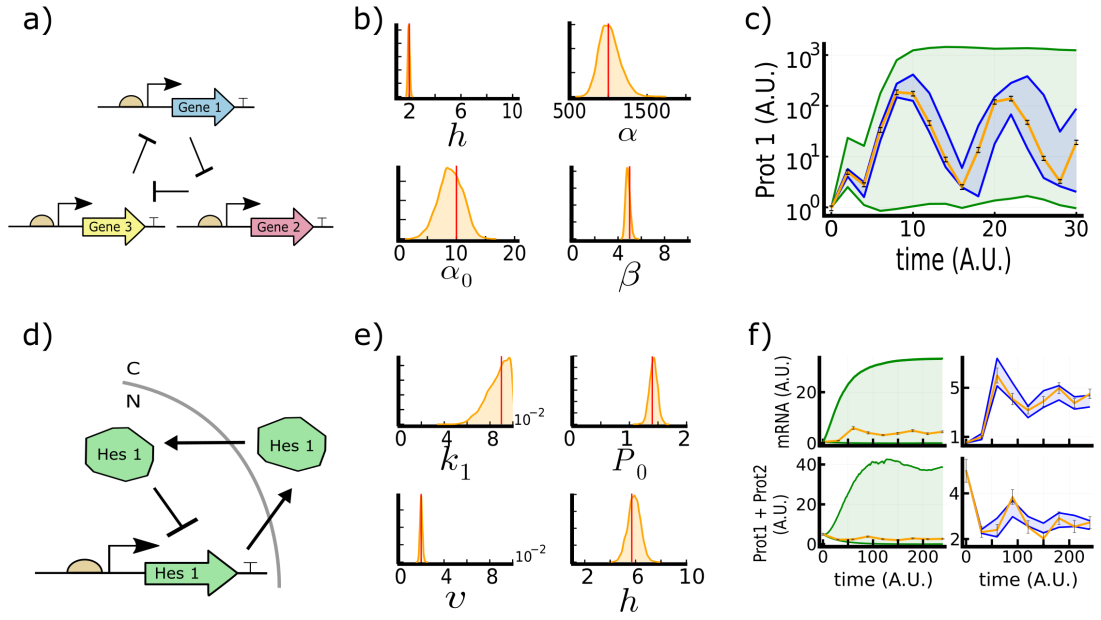


Figure 4.1: Bayesian inference of the oscillatory repressilator and Hes1 systems. a) Schematic of the oscillatory repressilator system. The circuit is composed of three genes, where the product of one represses the transcription of a subsequent gene, as illustrated. b) Kernel densities of the posterior distributions (orange) inferred, compared to the parameter values used to generate the pseudo-data (red). The pseudo-data used for the inference assumes a 10-fold decrease in the transcription rates α and the ratio of protein to mRNA decay rate β for gene two. Protein 1 is the only observable with heteroscedastic noise of 10 %. c) Comparison between the prior (green) and posterior (blue) predictions (represented as the 99.5th and 0.5th percentiles and area between the two for the resultant simulations) and mean (orange) and standard deviation (grey) of the pseudo-data used for the inference. d) Schematic of the oscillatory Hes1 system. The system is composed of the protein Hes1, which is translated in the cytoplasm and, when translocated into the nucleus, represses its own transcription. e) Kernel densities of the posterior distributions (orange) inferred, compared to the parameter values used to generate the pseudo-data (red). The pseudo-data used for the inference considers only mRNA as an observable with 10 % heteroscedastic noise. f) Comparison between the prior (green) and posterior (blue) predictions (99.5th and 0.5th percentiles) and mean (orange) and standard deviation (grey) of the pseudo-data used for the inference (top panel) and pseudo-data representing the total Hes1 protein in cells (bottom panel).

translated from gene one as a system observable, with a 10 % additive heteroscedastic Gaussian noise (a low error as observed in our microfluidic experiments for the genetic toggle switch, see Section 3.2.1.4, p. 73 of Chapter 3 for more details) associated with it (Fig. 4.1c, orange line with grey error bars).

For the Hes1 oscillator, I considered the same experimental scenarios as described in Liepe *et al.* 2013 [56], where only mRNA (set 1) or the combination of cytosolic (protein 1) and nuclear (protein2) proteins (set 2) are considered as system observables. I decided to add a 10 % additive heteroscedastic Gaussian noise in the generation of the pseudo-data (Fig. 4.1f, orange line with grey error bars). I then used the probabilistic programming language Stan [83] to infer the different model parameters on the experimental sets defined.

The inferred posteriors (obtained using the methodology described in Section 3.2.2.2, p. 77 of Chapter 3) align with the results reported in Liepe *et al.* 2013 [56]. For the repressilator, inference on experimental set 5 results on posterior probabilities with less uncertainty and centred around the parameter values Θ_R^T (Table 3.12, p. 73 of Chapter 3) used to generate the pseudo-data. Fig. 4.1b shows the marginal posterior kernel distributions (orange) compared to Θ_R^T (red) for this case, indicating a high centralisation to Θ_R^T and uncertainty reduction compared to the uniformly distributed prior beliefs (Table 3.11 from Section 3.2.1.3, p. 71 of Chapter 3). Additionally, the updated model shows the highest improvement when predicting the system states (Fig. 4.1c, blue shaded area) compared to its predictive capabilities before the prior knowledge is updated (Fig. 4.1c), green shaded area.

The supplementary Fig. S1 shows the parameter posteriors (orange kernel densities) inferred in the alternative four experimental sets compared to the parameter values used to generate the pseudo-data (red). The results show a generally low uncertainty reduction when using sets 1, 2 and 4. While when using set 3 yielded a high uncertainty reduction, this resulted in posteriors centred farther away from the parameter values used to generate the pseudo-data compared to when using experimental set 5. The supplementary Figs. S2, S3, S4, S5 and S6 show the model predictions for all system states, experimental sets and posteriors inferred when using sets 1, 2, 3, 4 and 5, respectively, to infer the repressilator model parameters. The quality of model predictions (regarding the accuracy and prediction uncertainty) is highly correlated with the quality of the

posteriors inferred (regarding uncertainty reduction and centralisation to the parameter values used to generate the pseudo-data).

Regarding the Hes1 system, considering mRNA as an observable yielded lower posterior uncertainty and marginal posteriors centred closer to the parameter values Θ_H^T (Table 3.13, p. 73 of Chapter 3) used to generate the pseudo-data. Fig. 4.1e again illustrates the marginal posterior kernel distributions (orange) compared to Θ_H^T (red) for this case. This results also highlight a high centralisation of posterior modes to Θ_H^T and variance reduction compared to the uniformly distributed prior beliefs (Table 3.11 from Section 3.2.1.3, p. 71 of Chapter 3). Furthermore, the updated model shows the highest improvement when predicting the system observables (Fig. 4.1f, right, blue shaded area) compared to its predictive capabilities before the prior knowledge is updated (Fig. 4.1f, left, green shaded area).

The supplementary Fig. S7a shows the parameter posteriors (orange kernel densities) inferred in the scenario where the total Hes1 cell protein is considered as observable. The results show a significantly higher posterior uncertainty for all parameters and a poorer centralisation of the parameter values used to generate the pseudo-data (red). The supplementary Fig. S7b shows the prior and posterior model predictions for the two experimental sets for the posterior inferred using the total Hes1 cell protein as an observable. These results indicate a prediction uncertainty of lower quality, especially for the observable mRNA.

4.1.2 Bayesian Inference of the Orthogonal Inducible Promoter Developed in *S. cerevisiae*

By implementing a step-wise approach on model simulations (see Section 3.2.1.4, p. 73 of Chapter 3), I extended the Bayesian inference scheme (described in Section 3.2.2.2, p. 77 of Chapter 3) to include inducible systems. The orthogonal inducible promoter developed by Gnügge *et al.* 2016 [228] is an example of a synthetic system where an external molecule (IPTG) is used to induce the expression of the fluorescent reporter Citrine (see a schematic of the system in Fig. 4.2a). See Section 3.2.1.2, p. 69 of Chapter 3 for details of the model and constraints used. Considering the experimental constraints imposed by our microfluidic platform, I set any pseudo-data generated to follow a sampling time

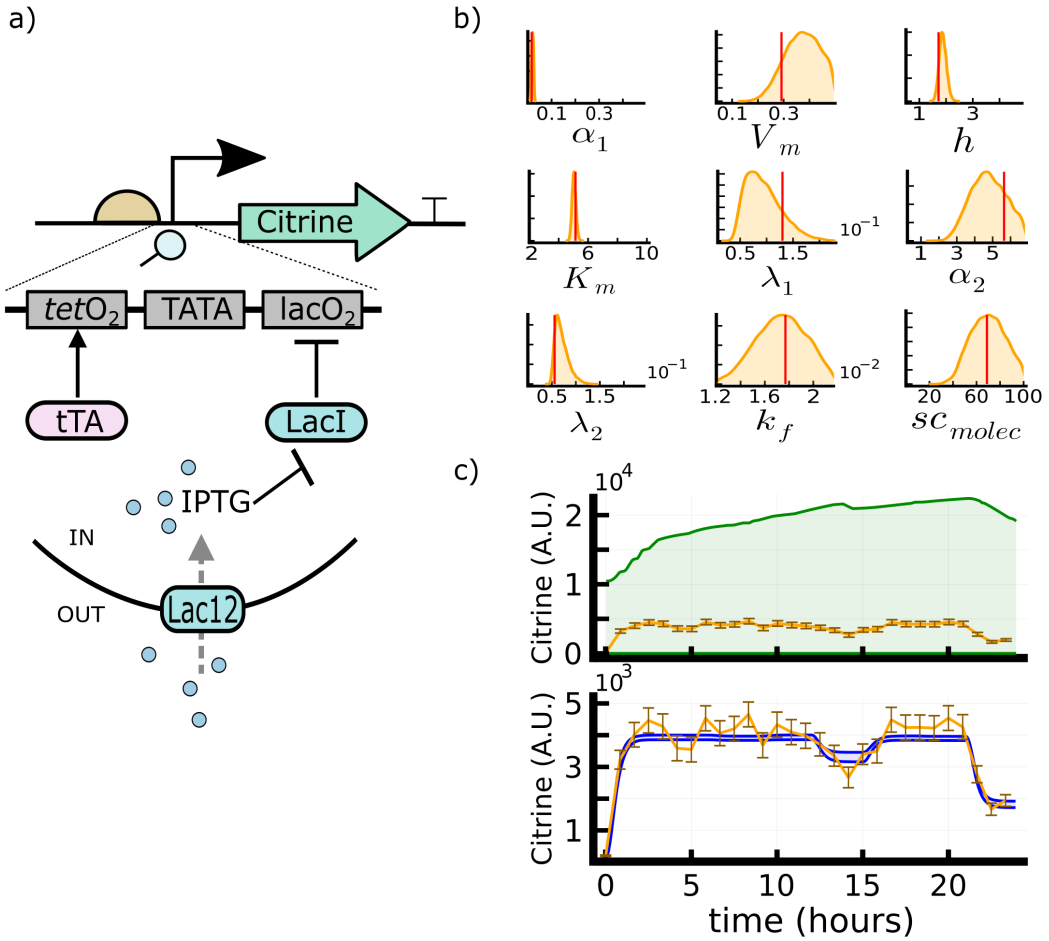


Figure 4.2: Bayesian inference of the inducible orthogonal promoter developed in *S. cerevisiae*. a) Schematic of the orthogonal inducible promoter. The inducible promoter, which contains the operator sequences $(tetO)_2$ and $(lacO)_2$, drives the expression of the fluorescent reporter Citrine. The chemical Isopropyl β-D-1-thiogalactopyranoside (IPTG) is used as an inducer to the system, blocking the LacI repressor activity and hence, allowing transcription of the reporter. The strain constitutively expresses the lactose permease Lac12 and the transcription factors LacI and tTA. b) Kernel densities of the posterior distributions (orange) inferred, compared to the parameter values used to generate the pseudo-data (red). In the experiment simulated, the concentration of IPTG changes every 3 hours at a random value selected between the bounds 0 and 100 μM (87, 84, 54, 92, 11, 58, 50, 4). The pseudo-data used for the inference assumes a 10 % heteroscedastic noise. c) Comparison between the prior (green) and posterior (blue) predictions (99.5th and 0.5th percentiles) and mean (orange) and standard deviation (grey) of the pseudo-data used for the inference.

of 5 minutes, a total experiment length of 24 hours and a constant concentration of IPTG allowed to change in events of 3 hours of duration. All pseudo-data generated assumed a 10 % additive heteroscedastic (a low error as observed in our microfluidic experiments, see Section 3.2.1.4, p. 73 of Chapter 3 for more details) Gaussian noise (Fig. 4.2c, orange line with grey error bars). I used the priors defined in table 3.9 (p. 70) of Chapter 3 truncated using the specified upper and lower bounds for all inferences regarding the system.

Bayesian inference on this type of dynamic experiments allows for a significant shrinkage for most parameter priors, centring the parameter posteriors around the parameter values Θ_{PL}^{T} (Table 3.10, p. 71 of Chapter 3) used to generate the pseudo-data. The marginal posterior kernel distributions illustrated in Fig. 4.2b show this effect for most of the parameters considered (except V_m and λ_1) comparing them to the priors used for inference (Table 3.9 in Section 3.2.1.2, p. 69 of Chapter 3). Nonetheless, I do not observe a noteworthy difference between the parameter posteriors obtained on three different experiments, hinting a similar information content for these. Moreover, the updated model shows a high improvement predicting the system observables (Fig. 4.2c, bottom panel, blue shaded area) compared to its predictive capabilities before the prior knowledge is updated (Fig. 4.2c, top panel, green shaded area).

The supplementary Fig. S8 shows the posterior probabilities (a,b) and model predictions (c,d) for the two alternative sets of pseudo-data. As indicated, the parameter posteriors from panels a and b show a high similarity to Fig. 4.2b, except for the parameter λ_1. All three model predictions were of similar quality, with a significantly lower uncertainty when compared to the prior ones.

4.1.3 Bayesian Inference of the Genetic Toggle Switch Developed in *E. coli*

The results of this sub-section were published in ***D. G. Cabeza†, L. Bandiera†, E. Balsa-Canto and F. Menolascina, "Information content analysis reveals desirable aspects of in vivo experiments of a synthetic circuit," 2019 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), 2019, pp. 1-8, doi: 10.1109/CIBCB.2019.8791449 .***

To assess the informative content of different *in vivo* time-series and the predictive capabilities of alternative model structures for a genetic circuit, I adapted the step-wise (see Section 3.2.1.4, p. 73 of Chapter 3) Bayesian inference approach for the genetic toggle switch developed by Lugagne *et al.* 2017 [85]. The synthetic circuit is a dual-input dual-output system with two mutually repressing transcriptional factors (LacI and TetR) fused to fluorescent reporters (RFP and GFP, respectively). To externally control the system, two external inducers (IPTG and aTc) can be administered to inhibit the activity of the transcriptional factors (see a schematic of the system in Fig. 4.3a).

I consider the mathematical formulation of the circuit as described in Lugagne *et al.* 2017 [85], however, due to uncertainties on the inducers flux mechanism across the cell membrane, I assessed three alternative models to describe the transport of these. The first model (Fig. 4.3b), M_1, assumes non-symmetrical influx and efflux rates for the inducers, as described in Lugagne *et al.* 2017 [85]. Instead, the second model (Fig. 4.3c), M_2, follows Fick's law of diffusion to equalise the two flux rates. The last model considered (Fig. 4.3d), M_3, expands on the pure diffusion assumption of M_2 to introduce an effective degradation rate caused by cell growth and division. The different ODEs for each model are described in Section 3.2.1.1 (p. 64) of Chapter 3.

Finally, I retrieved a subset of 10 microfluidic experiments from Lugagne *et al.* 2017 [85] (Table 3.5, p. 62 of Chapter 3) to infer the previously mentioned models (as described in Section 3.2.2.2, p. 77 of Chapter 3). The experiments track the fluorescence of the reporter proteins of the system (RFP and GFP), using fluorescence microscopy, when cells become exposed to step-wise constant perturbations of IPTG and aTc over time. The retrieved data sets included three calibration experiments (Ec_{1-3}), characterised by a low number of changes in the inputs profile, and seven dynamic experiments (Ed_{1-7}), with a higher number of input changes per experiment. The former set was used in Lugagne *et al.* 2017 [85] to calibrate the original model, while the latter was used to maintain the system around the meta-stable steady-state.

Inference of model M_3 on individual data sets uncovers a difference in informative content between the calibration and the dynamic experiments. For most parameters, the marginal posteriors show lower uncertainty when inferred on

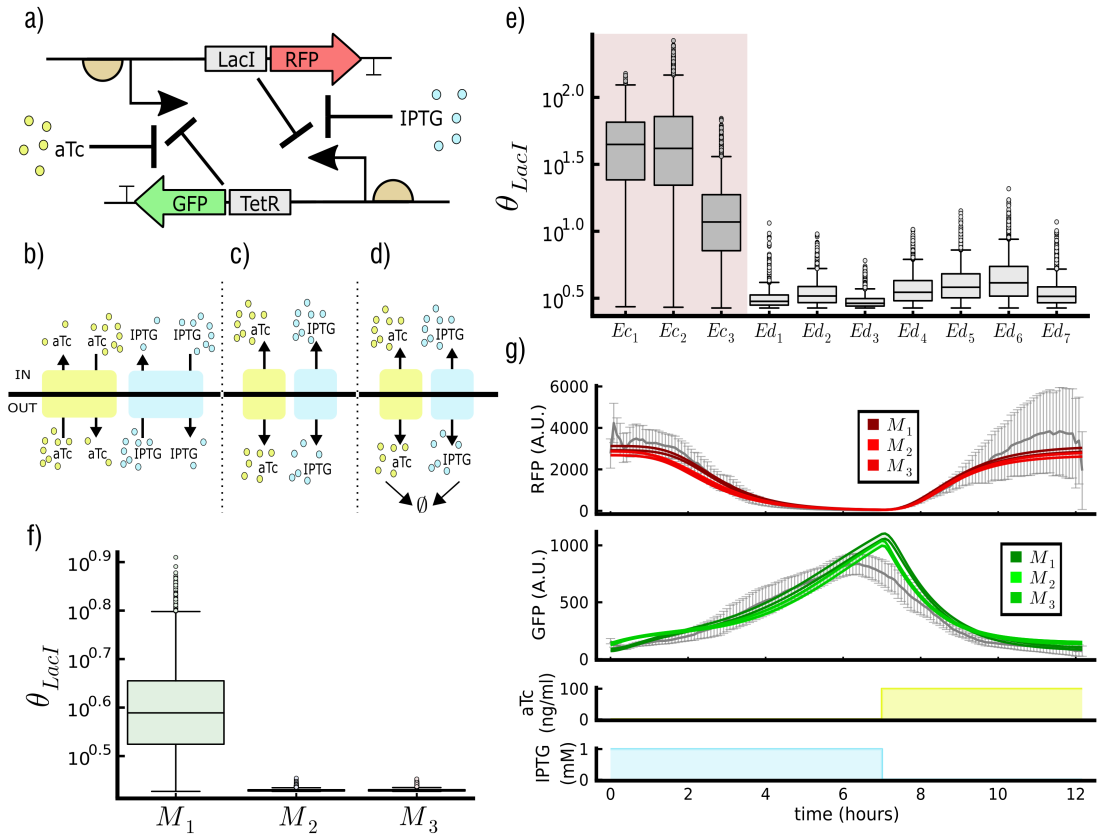


Figure 4.3: Bayesian inference of three models for the genetic toggle switch developed in *E. coli*. a) Schematic of the toggle switch system. The circuit is composed of two mutually repressing proteins (LacI and TetR) fused to two different fluorescent reporters (mKate2 and mEGFP). The exogenous inducers Isopropyl-β-D-1-thiogalactopyranoside (IPTG) and anhydrotetracycline (aTc) inhibit the activity of the two transcription factors. b) Schematic for the first alternative model structure, assuming an asymmetric transport rate for the two inducers through the cell membrane. c) Schematic for the second alternative model structure, assuming a symmetric transport rate (diffusion). d) Schematic of the third alternative model structure, assuming a symmetric transport rate (diffusion) as well as a dilution factor for the inducers caused by cell growth and division. e) Marginal distribution box plots of θ_{LacI} (least sensitive parameter) posteriors for model M_3 inferred in each experiment considered from Lugagne *et al.* 2017 [85]. The three marginal distributions in the red area come from inferences of calibration experiments, while the rest come from inferences of dynamic experiments. f) Marginal distribution box plots of θ_{LacI} posteriors for models M_1, M_2 and M_3 inferred in all ten experiments. g) Posterior predictions (99.5th and 0.5th percentiles) for models M_1, M_2 and M_3 inferred on the ten experimental data sets, superposed to experimental data for Ec_1. The lower panels show the input profiles for aTc and IPTG.

dynamic experiments compared to inferences using the calibration data set. Fig. 4.3e shows bar plots of the posterior samples for the least sensitive model parameter, θ_{LacI}, inferred on each experiment. As highlighted in the figure, the posteriors inferred on the three calibration experiments present a significantly larger spread across the parameter space when compared to the ones inferred on the seven dynamic experiments, suggesting a lower informative content for the task. I also noted a difference in trend between the first moments of the parameter posteriors when inferred on calibration or dynamic experiments. This trend was a general observation across most parameters, accentuating the variability in the informative content of experiments, suggesting that a single set was insufficient for accurate inference. The supplementary Fig. S9 contains the bar plots for all model inferred parameters, further highlighting the different effects in inferred posteriors when using calibration or dynamic experiments, as represented for θ_{LacI} in Fig. 4.3e.

To further explore the overall effect of the experiments on the first and second moment of the marginal posteriors and the predictive capabilities of the models, I inferred the three alternative models in the combined data sets. The obtained marginal posteriors for model M_3 inferred on all experiments show a general agreement tendency on the first moment of the distributions to the cases where I inferred the model using only dynamic experiments as indicated in Fig. 4.3f. Considering more dynamic experiments than calibration ones on the inference could explain this tendency. However, I want to highlight that the general model prediction capabilities improve after the multi-experimental inference (Fig. 4.4a, b, c, d), suggesting the incapability to recover the underlying parameter probabilities with calibration experiments alone. Furthermore, posteriors for the matching parameters on the three considered models show a general agreement in their first moment, with M_1 obtaining an overall slightly higher second moment when compared to M_2 and M_3 (Fig. 4.3f). Nonetheless, there is no drastic apparent prediction degradation on model M_1 compared to the other models as seen in Fig. 4.3g and Fig. 4.4c and Fig. 4.4d).

The supplementary Fig. S10 contains the bar plots for the inferred posteriors for the three considered models. The results show a generally higher posterior uncertainty for model M_1. These results also indicate a high concordance between the first and second moments of the posteriors from models M_2 and M_3 for most of the shared parameters, suggesting that the introduction of pure

diffusion in the model had a significant effect on the moments of the inferred posteriors. The supplementary Fig. S11 shows the prior predictions of the three considered models on the ten data sets to showcase the high uncertainty generated when using highly uninformative priors. The supplementary Fig. S12 shows the posterior predictions of the three considered models for the remaining nine data sets to highlight the quality of the posteriors obtained in terms of model predictions, the significant uncertainty reduction and the similarity in the predictions of the three models.

The predictions of model M_3 range amongst a broad spectrum of accuracies when inferred on individual data sets, while all models are capable to better generalise predictions when inferred on all data sets. I first computed the average normalised root mean square error (nRMSE) of model M_3 inferred on each experiment predicting the observables RFP (Fig. 4.4a) and GFP (Fig. 4.4b) for all the considered data sets. As expected, the model better predicts the experiment used for inference when compared to the remaining nine experiments (heat map diagonals of Fig. 4.4a,b). When predicting unseen experiments, the model better recovers GFP measurements than RFP. For all inferred posteriors, the model generally predicts better experiments Ed_5, Ed_6 and Ed_7 for RFP measurements (Fig. 4.4a with lower nRMSE values) and experiments Ed_4, Ed_5 and Ed_6 for GFP measurements (Fig. 4.4b with lower nRMSE values). The model using posteriors inferred from experiments Ed_5, Ed_6 and Ed_7 is generally worst at recovering the RFP dynamics of the rest of the experiments as seen by the higher nRMSE values. Also, in all cases, the model shows poor predictive capabilities for GFP measurements of Ec_1 and Ed_7. The prediction accuracy difference between the best scenario (lowest average nRMSE of RFP and GFP measurements excluding predictions of experiments used for the inference) shown in Fig. 4.4e and the worst scenario (highest average nRMSE) shown in Fig. 4.4f becomes more evident when observing the posterior predictions of model M_3. Finally, I inferred the three considered models on all experimental sets using a multi-experimental inference approach, recovering better predictions for the RFP (Fig. 4.4c) and GFP (Fig. 4.4d) measurements of all experiments. This results suggest the general insufficiency of a single experiment inference to be able to predict the system observables on unseen data.

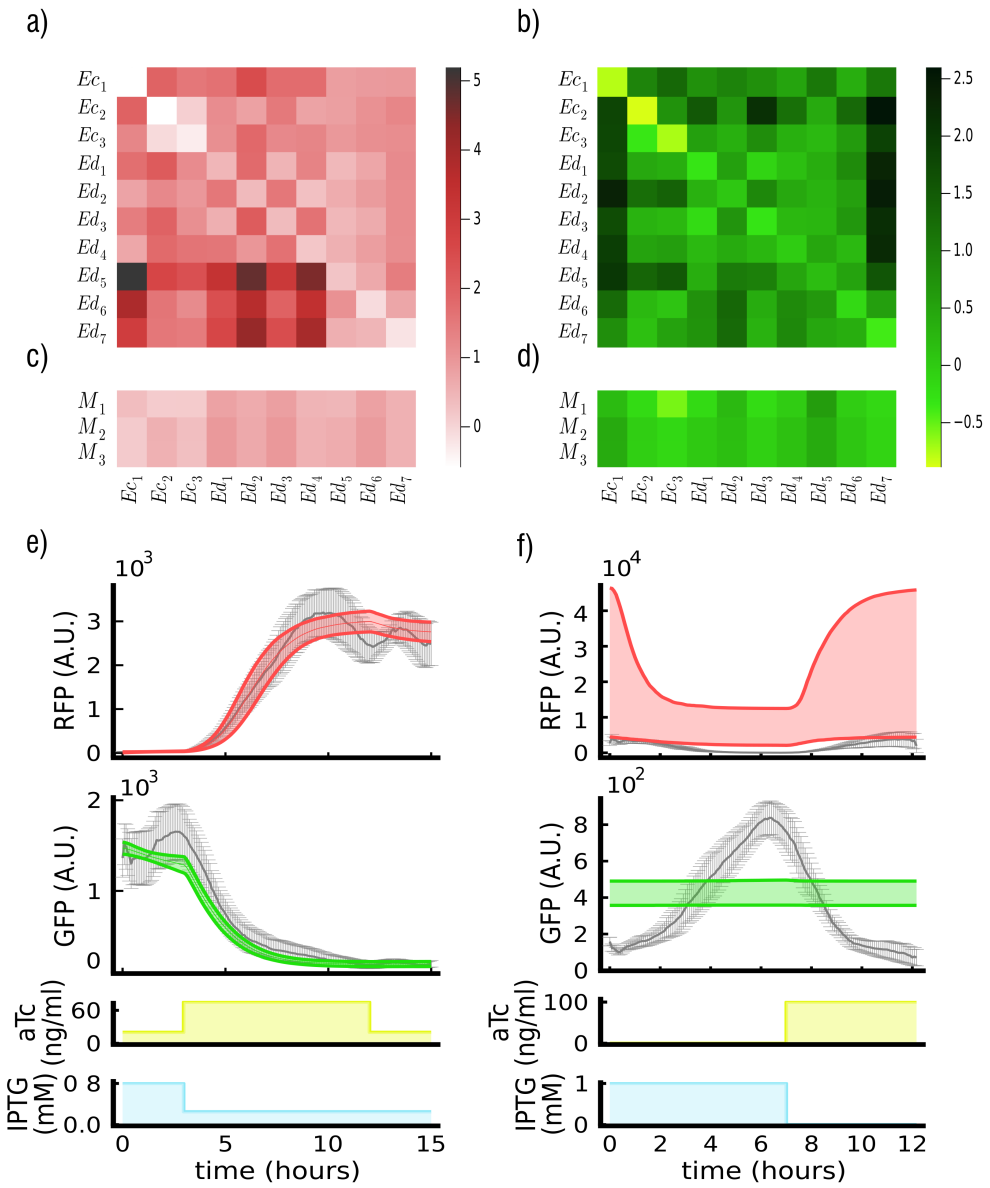


Figure 4.4: Predictive capabilities of the different toggle switch models inferred on experimental data. Heat maps of the RFP (a) and GFP (b) mean normalised root mean square errors (nRMSE) for the toggle switch model M_3 when predicting each of the ten experiments used from Lugagne *et al.* 2017 [85] (horizontal axis) after being inferred by each experiment (vertical axis). Heat map of the RFP (c) and GFP (d) mean nRMSE for the three toggle switch models considered when predicting each of the ten experiments used from Lugagne *et al.* 2017 [85] after being inferred by all experiments. e) Posterior prediction (99.5th and 0.5th percentiles) from model M_3 of experiment Ec_2 after being inferred using experiment Ec_3 (best case scenario). f) Posterior prediction (99.5th and 0.5th percentiles) from model M_3 of experiment Ec_1 after being inferred using experiment Ed_5 (worst case scenario). The lower panels show the input profiles for aTc and IPTG.

4.1.4 Conclusions

Parameter identification for models describing biological systems remains a daunting task due to the inherent difficulty of their measurements, needing them to be inferred from experimental data. In this section, I showed how Bayesian statistics could be applied to infer model parameters of four different biological systems of increasing complexity. Furthermore, parameter posteriors allowed me to assess the confidence around parameter estimates without relaying on over-simplistic noise assumptions, allowing me to evaluate model predictions and the uncertainties linked to these. First, I compared inferences on models for the repressilator and the Hes1 system to previously published studies to validate the strategy. Then I extended my inference scheme to include inducible systems since these are becoming of increasing popularity in synthetic biology, something previously done only in frequentist scenarios. Inferences of models for the orthogonal inducible promoter and the genetic toggle switch highlight the importance of careful experimental design. Some experiments might provide insufficient information about the system. Others might not be informative enough to obtain reliable predictive models. Yet, as shown in the results regarding the genetic toggle switch, there are experimental designs that are more informative than others when used to infer model parameters. Overall, the results show that Bayesian statistics can aid the identification of parameters and the uncertainties associated with these, helping to understand the predictive capabilities of models and hence, if more or better experimental data is required.

4.2 *A Posteriori* Informative Content Analysis of Experiments for Biological Systems

Quantitative metrics are needed to assess how much information a model gained from an experiment or type of stimuli applied. In this section, I will show that, in computational studies, the posterior sample distance to the "true" parameters (the ones used to generate pseudo-data) can become a powerful analysis tool. The metric aids to distinguish good inference results from inferences with low posterior uncertainty but a high distance to the true parameters. I then used the posterior covariance to assess the informative content of different experiments [82]. Since normality of posterior probabilities for non-linear systems infrequently stands in practice, I used Gaussian mixture models to approximate sampled posteriors. Then, I defined the gain in information, and hence the informativeness of experiments, as the difference between prior and posterior entropies [56]. Finally, I compared all metrics to determine the effect of imposing a normality assumption on parameter posteriors to rank the *a posteriori* informative content of experiments.

4.2.1 A Quantitative Analysis of Parameter Knowledge Improvement

To quantify the quality of inferred posteriors for biomodels, I assessed the average sample distance to set parameter values in computational studies. In these studies, where I fixed a set of parameters Θ^T to generate pseudo-data, I adopted a normalised posterior sample distance d_{PSD} to Θ^T to quantify the quality of the inferred posteriors (see Section 3.2.2.3, p. 79 of Chapter 3 for more details). To ease comparison between parameters, the sample distance was normalised using prior bounds. Applying the metric to posteriors inferred from the repressilator and Hes1 system models (Section 4.1.1, p. 112 of this chapter) helped understand the overall tendency of uncertainty shrinkage of an inference for the set priors and the error identifying parameters used to generate the pseudo-data.

The best experiment defined in Liepe *et al.* 2013 [56] for the repressilator (set 5) clearly shows higher uncertainty shrinkage for all the parameters. Posteriors inferred on set 5 were closer to the parameter set Θ_R^T used to generate the pseudo-data compared to posteriors from set 3, which also attained good

uncertainty shrinkage but a poorer centralisation to Θ_R^T. Fig. 4.5a shows violin plots for the d_{PSD} for each parameter posterior obtained for each set, highlighting the general lower variance for posteriors inferred from set 3 and 5, and a closer centralisation around 0 (i.e. Θ_R^T) for the latter. The results also indicate that the transcription factor α was the toughest parameter to infer (Fig. 4.5a top-right). I observed similar tendencies from the Hes1 system results. Set 1 attains a higher overall shrinkage of posterior uncertainties (indicated in Fig. 4.5c using d_{PSD}) and better centralisation of the marginal posteriors over the parameter set Θ_H^T (value of $d_{PSD} = 0$) used to generate the pseudo-data. The rate of protein transport to the nucleus k_1 appeared to be hard to identify in both inferences, showing a high similar variance for both posteriors, yet, with a better centralisation to $d_{PSD} = 0$ for the one obtained from set 1. The supplementary Fig. S13a shows the two metrics (d_{PSDM} and $log(|\Sigma|)$) applied to the Hes1 system inferences, showcasing the high correlation between the two. Hence, most of the differences between posteriors resided in their uncertainty.

Posterior variances aid in assessing the quality of posterior inferences, especially in cases where Θ^T is unknown, as it would happen with *in vivo* experiments. Commonly used Fisher Information Matrix (FIM)-based methods [82] close the parameter posteriors at the second moment seeking to assess posterior variances as a quality metric. The metric I adopted to quantify the multivariate posterior variances was the logarithm of the determinant of the covariance matrix of posteriors (see Section 3.2.2.3, p. 79 of Chapter 3 for more details). Fig. 4.5b shows the correlation between the metric and the average d_{PSD} for the five repressilator posteriors. The high correlation between the two highlights their comparative efficiency in ranking the *a posteriori* informative content of experiments for the repressilator, since the same experimental ranking is achieved by both. Similarly, both metrics achieve the same experimental ranking for the three inferences performed with the orthogonal inducible promoter model as seen in Fig. 4.5d. As hinted by observing the different marginal posteriors for the three inferences (Section 4.1.2, p. 115 of this chapter), both metrics indicate not much informative difference between the three experiments. The supplementary Fig. S13c shows the violin plots of d_{PSD} for the three inferences of the orthogonal inducible promoter, highlighting the high similarities for the posteriors inferred using the three selected experiments and hence the comparable informative content of these.

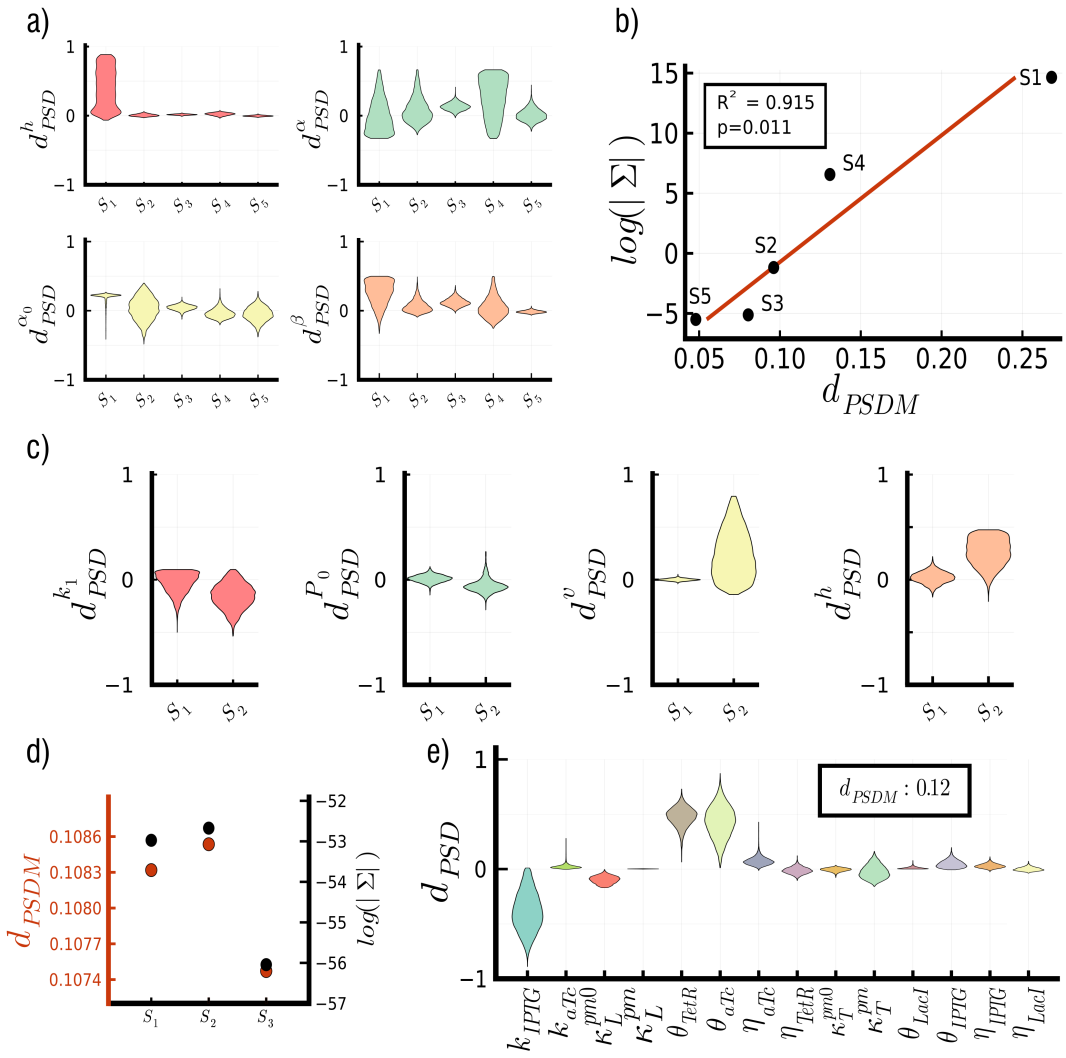


Figure 4.5: *A posteriori* experiment informative content analysis for four different *in silico* studies. a) Repressilator normalised posterior sample distance (d_{PSD}) to the parameter values selected to generate the pseudo-data for each of the five data sets considered. b) Correlation plot between the repressilator average posterior sample distance (d_{PSDM}) and a metric for the posterior covariance for each data set considered. c) Hes1 d_{PSD} when mRNA (S_1) or the total cell Hes1 protein (S_2) is considered as an observable for the inference. d) The orthogonal inducible promoter d_{PSDM} and a metric for the covariance of the posteriors obtained after inferring the model on each of the three sets of pseudo-data considered. e) The toggle switch model M_3 d_{PSD} for each model parameter after being inferred on a 24 hour pseudo-experiment where the concentration of the two external inducers change every three hours at random values between the bounds 0 to 1 mM for IPTG (0.07, 0.61, 0.58, 0.02, 0.88, 0.39, 0.64, 0.86) and the bounds 0 to 100 ng/mL for aTc (62, 87, 11, 94, 73, 88, 23, 24).

I inferred the toggle switch model M_3 on pseudo-data generated using the parameter vector Θ_{TS}^{T1} (Table 3.8, p. 68 of Chapter 3) and the parameter priors Prior_b (Table 3.7, p. 67 of Chapter 3) to assess the posterior shrinkage of marginal posteriors and their centralisation around Θ_{TS}^{T1}. The d_{PSD} metric, seen in Fig. 4.5e, showed high shrinkage and centralisation to 0 for most of the parameters, while highlighting the less identifiable ones (k_{IPTG}, θ_{TetR} and θ_{aTc}), which could be the focus of subsequent inferences. This following inferences could consider the well identified parameters as fixed values, emphasising posterior inference for the rest, or use optimal experimental design strategies (see Chapter 7 for more details). Nonetheless, while both metrics are quick to compute, the different d_{PSD} (and the marginal posteriors) for all assessed models highlights the general lack of normality for most posterior distributions. This posterior normality deviation trend suggests a potential loss of information in the metrics when quantifying parameter knowledge improvement. The supplementary Fig. S13b shows the pseudo-data and posterior prediction for the toggle switch model M_3. The results indicate the high prediction capabilities of the model when simulating the experiment used for inference despite a poor d_{PSD} for k_{IPTG}, θ_{TetR} and θ_{aTc}, suggesting that this had a low informative content for these parameters.

4.2.2 An Information Theory Approach for Information Content Analysis of Experiments

The results regarding the genetic toggle switch in this sub-section were published in ***D. G. Cabeza†, L. Bandiera†, E. Balsa-Canto and F. Menolascina, "Information content analysis reveals desirable aspects of in vivo experiments of a synthetic circuit," 2019 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), 2019, pp. 1-8, doi: 10.1109/CIBCB.2019.8791449.***

To elude posterior normality assumptions when quantifying the *a posteriori* informative content of experiments, I used an information-theoretic approach. I computed the gain in information as the difference between the multivariate prior and posterior Shannon entropy as a metric to assess how much the parameter posterior uncertainties shrunk compared to the priors. Due to an assumption of independence of the marginal priors in all cases, I used closed forms for the

entropy of all uniform and Gaussian priors considered. I fitted the sampled posteriors using Gaussian mixture models and approximated the Shannon entropy of the multivariate fitted distributions as described in Huber *et al.* 2008 [334]. More details for the methodology are described in Section 3.2.3 (p. 81) of Chapter 3.

First, I estimated the posterior entropies of the different single experiment inferences of the toggle switch model M_3 on the ten data sets retrieved from Lugagne *et al.* 2017 [85], ensuring that each approximation laid between the confidence bounds described in Huber *et al.* 2008 [334]. Fig. 4.6a shows that all these estimates (red dots) fall within bounds (red shaded area). The results indicate a lower Shannon entropy (and hence a higher gain in information) for posteriors inferred on dynamic experiments compared with the ones inferred from calibration experiments, in concordance to the results from Section 4.1.3 (p. 117) of this chapter. Information gain increased with experiments containing higher perturbation steps for the inducers, where Ec_2 resulted in the least informative (3 steps) and Ed_6 the most informative (11 steps), with an $\sim$ 42% increase in information gain. The supplementary Fig. S14a shows a comparison between the gain in information of the ten toggle switch experiments compared to the number of step changes of the inducers in these, indicating, once again, the high correlation between the two.

I then compared the gain in information using prior and posterior entropies to the FIM-based method that closes posteriors at the second moment. Fig. 4.6b indicates a high correlation between both metrics when assessing the ten experiments retrieved from Lugagne *et al.* 2017 [85]. Nonetheless, despite the high correlation between the two, there are some changes in the experimental informativeness ranking. The most notable discrepancies between the two metrics were the ranking of Ed_5 and Ec_1, and the selection of the most informative experiment. These results spotlight the common deviation from normality of inferred parameter noise, the potential detrimental effects of Gaussianity assumptions and the advantage of Bayesian statistics in such scenarios.

To extend the analysis into a different system, I compared both metrics using the inferences performed on the repressilator model (where I know the parameter values used to generate the pseudo-data), also including d_{PSDM} in the comparison. The three metrics shared a high correlation as well as the same

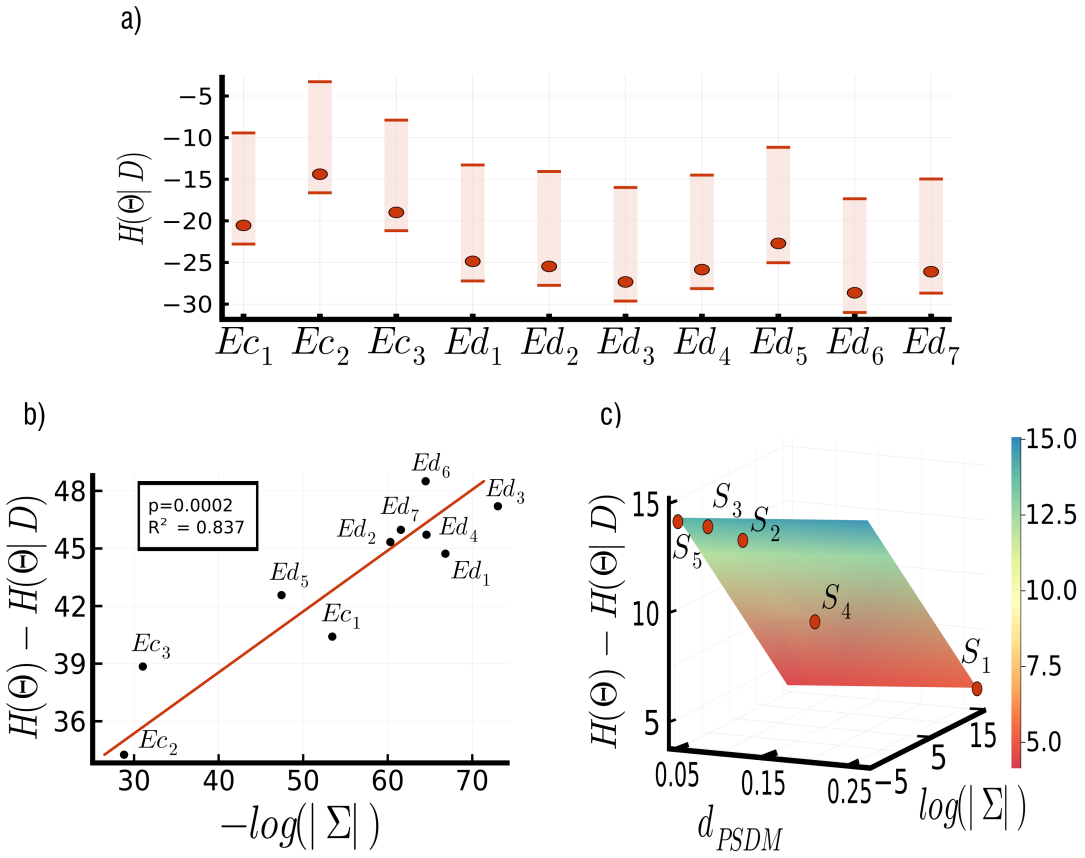


Figure 4.6: Posterior entropy approximation to assess the informative content of experiments. a) Estimated posterior entropy with upper and lower bounds for the toggle switch model M_3 after being inferred using each experiment retrieved from Lugagne *et al.* 2017 [85]. b) Correlation plot between the information gain (measured as the difference between prior and posterior entropies) and posterior distribution variance for the toggle switch model M_3 after being inferred using each experiment retrieved from Lugagne *et al.* 2017 [85]. c) Correlation plot between the information gain, posterior distribution variance and d_{PSDM} for the repressilator model inferred on the five considered data sets.

experimental ranking between them as seen in Fig. 4.6c. The supplementary Fig. S14b shows a comparison between the two information metrics (posterior entropy gain in information and a metric for the posterior covariance) when I inferred the three toggle switch models on all data sets. The results indicate, in this case, a low correlation between the two, highlighting the general absence of normality of the inferred posteriors. The supplementary Fig. S14c shows a comparison of the two metrics on the Hes1 system inferences, indicating a high correlation between the two. The supplementary Fig. S14d shows a comparison of the two metrics on the orthogonal inducible promoter inferences. Both metrics indicate, once again, a comparable gain in information between the three cases. Yet, when assuming posterior normality, the differences seemed to increase, particularly for set 3, highlighting the potential effects of this normality approximation.

4.2.3 Conclusions

Analysing parameter posteriors to quantify the amount of information gained from experiments stresses the difference in their informative content. In this section, I showed that, in computational studies, we could use d_{PSD} to examine the quality of inferred posteriors, considering uncertainty and distance to the value set to generate pseudo-data. Moreover, I can use its average, d_{PSDM}, to rank the informative content of different experiments. Since with *in vivo* experiments the real value for the parameters is usually unknown, I assessed a commonly used FIM-based method to quantify the informativeness of experiments, showing a high correlation with d_{PSDM} in all tested scenarios. Nevertheless, the metric assumes that inferred posteriors follow a multivariate Gaussian distribution, which rarely holds in these cases. For this reason, I approximated the sampled posteriors using Gaussian mixture models. I then approximated their Shannon entropy to compute the gain in information, defined as the difference between the prior and posterior entropies. The metric showed a high correlation between the previous two. However, for the genetic toggle switch, the ranking of experiments was not the same as the one for the metric that closes posteriors at their second moment. These results underline the lack of normality of posterior estimates, having a cumulative effect on systems with larger number of parameters. Altogether, analysing parameter posteriors inferred from different experiments highlights the informative difference between experimental designs. Also, this is an *a posteriori* analysis, which emphasises the need for strategies to assess the informative content of experiments before their execution, saving time and resources from experimentalists.

4.3 Prediction Uncertainty of Mathematical Models for Biological Systems and Link to Parameter Knowledge

Uncertainties on model parameters propagate to model predictions, making this relation a potential target to quantify the informative content of experiments *a priori*. In this section, I will show that while normality noise assumptions on parameters can speed up identification, it is at the expense of higher uncertainties. Moreover, these higher parameter uncertainties can pass on model predictions. I used one of the models for the genetic toggle switch as a case study to highlight uncertainty differences between frequentist and Bayesian approaches. I then will show how I can exploit the relation between parameter and prediction uncertainties to assess the informative content of potential experiments. I used the repressilator and Hes1 system to evaluate the potential of the strategy to rank experiments according to their informative content, comparing it to previously published results estimating mutual information. Finally, I used the procedure to assess the informative content of experiments for the genetic toggle switch and the orthogonal inducible promoter, systems with more complex models where I set highly uninformative priors.

4.3.1 A Model Prediction Uncertainty Comparison Between Frequentist and Bayesian Methods

The results of this sub-section were published in ***L. Bandiera†, D. Gomez-Cabeza†, J. Gilman, E. Balsa-Canto and F. Menolascina, "Optimally Designed Model Selection for Synthetic Biology," ACS Synthetic Biology, 2020, 9 (11) pp. 3134-3144, doi: 10.1021/acssynbio.0c00393***.

Following up on the multi-experimental inferences performed on three mathematical models of the genetic toggle switch (M_1, M_2 and M_3), I estimated the models' parameters with a frequentist approach, using the same data sets retrieved from Lugagne *et al.* 2017 [85]. See Section 3.2.2.1 (p. 76) of Chapter 3 for more details of the methodology used. I solved the non-linear optimisation problem using the meta-heuristic algorithm enhanced Scatter Search (eSS),

contained in the Matlab toolbox AMIGO2 [74]. I used the bounds specified in table 3.7 (p. 67 of Chapter 3) to constrain the parameter space. Note that, the frequentist approach attributes the same probability to each region of the parameter space (as if using uniform priors), something generally assumed in this cases (unless regularisation is introduced [74]), contrary with the priors generally used in the Bayesian scenarios. To compare parameter and prediction uncertainties, I followed the common assumption of normally distributed parameters, approximating variances and covariances from confidence intervals and parametric sensitivities using the FIM (see Section 3.2.5, p. 84 of Chapter 3). As for Bayesian inferences, I used the same results reported in Section 4.1.3 (p. 117) of of this chapter.

Bayesian methods to identify parameters significantly reduce model prediction uncertainties compared to frequentist ones. When comparing parameter posteriors with the Gaussian distributions approximated from frequentist parameter estimation (PE), the former resulted in lower parametric uncertainties. Fig. 4.7a shows the case for the pairs of parameters with the highest normalised standard deviation for each of the two scenarios considered and for each model considered. As seen in the bivariate scatter plots, posteriors obtained using Bayesian inference present a similar or significant lower spread (i.e. variance) compared to the ones approximated using frequentist methods. For each scenario where I consider the worst case scenario for the frequentist PE (Fig. 4.7a, x axis), the Bayesian results obtained drastically better results. Conversely, when considering the worst case scenario for the Bayesian inference (Fig. 4.7a, y axis) the two methodologies obtain comparable results. On average, Bayesian posteriors had a ~ 9.44, 111.45 and 82.38 times reduction of σ for models M_1, M_2 and M_3 respectively, compared to their frequentist counterparts estimated using the FIM. Fig. 4.7a illustrates the uncertainty reduction on bivariate plots of samples from Bayesian and frequentist distributions for the three models. For the parameter with the relative highest σ from the frequentist results, the posteriors from Bayesian inferences showed a significantly smaller uncertainty.

I observed comparable uncertainties when considering the parameter with the relative highest σ from the Bayesian results. These parametric uncertainties propagate to model predictions, making Bayesian posteriors more accurate to predict the experiments used for the inferences as seen looking at the differences between Fig. 4.7b (frequentist predictions with uncertainty) and Fig. 4.7c

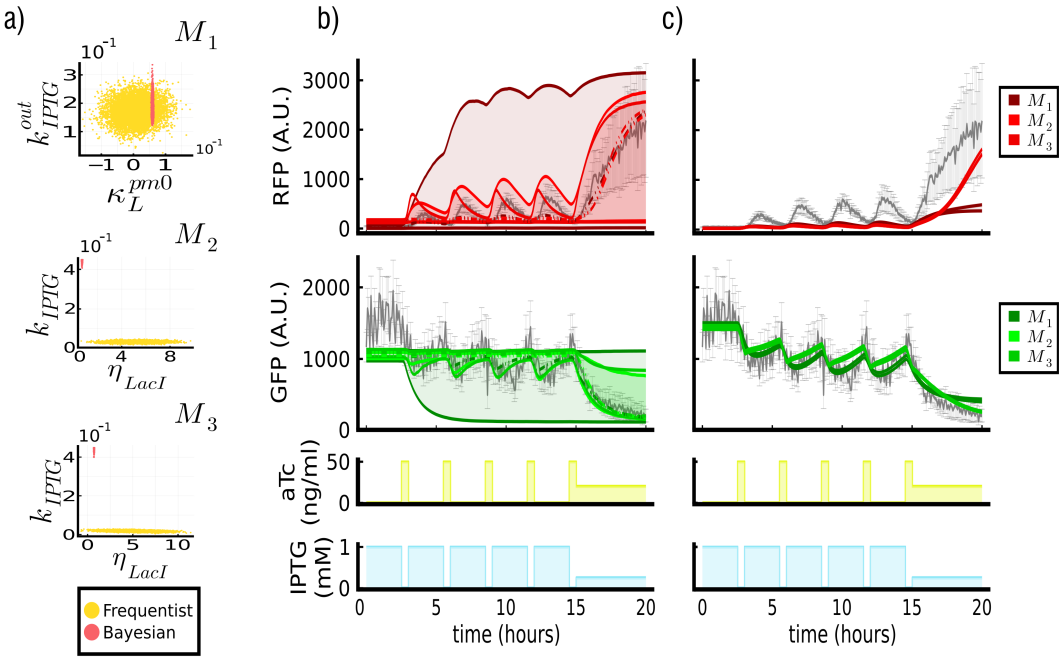


Figure 4.7: Posterior and prediction uncertainty comparison between Bayesian and frequentist methods. Bivariate scatter plots for the toggle switch models M_1, M_2 and M_3 of 8000 samples retrieved from the parameter posterior (red) after Bayesian inference on the ten data sets used from Lugagne *et al.* 2017 [85] or an approximated multivariate normal distribution (yellow) obtained after frequentist parameter estimation of the three models considering also the ten data sets. For each model, the parameter pairs displayed are the ones with the highest normalised standard deviation for the frequentist (x axis) and the Bayesian (y axis) scenarios. b) Prediction uncertainty for the three toggle switch models (99.5th and 0.5th percentiles represented as shaded areas) using 8000 samples from the approximated multivariate normal distribution (frequentist case) when simulating the experiment with the average minimum nRMSE for the three models (Ed_6). The models' simulation using the best parameter estimate is represented in dashed lines. The lower panels show the input profiles for aTc and IPTG. c) Prediction uncertainty for the three toggle switch models (99.5th and 0.5th percentiles represented as shaded areas) using 8000 posterior samples after Bayesian inference when simulating the same experiment as in panel b.

(Bayesian predictions). Model simulations using the best frequentist parameter estimates (Fig. 4.7b, dashed lines) resulted in accurate predictions for the experimental data. However, the introduction of parametric uncertainties generally reduced their predictive power as indicated by the high prediction errors observed in Fig. 4.7b (shaded areas). Meanwhile, Bayesian methods attained accurate predictions with a significantly lower simulation uncertainty as seen in Fig. 4.7c, where prediction uncertainties (shaded areas) are significantly low in comparison to the previous results. The supplementary Fig. S15 shows the frequentist models predictions for the remaining experiments used for PE to highlight that having good predictions when using the best parameter estimate but with high uncertainty when sampling the multivariate normal distributions extended to all the considered cases.

4.3.2 Prior Knowledge Propagation to Model Prediction Uncertainty

Propagation of parametric uncertainties to model predictions can be exploited to assess the informative content of experiments. As highlighted in Section 4.3.1 (p. 131) of this chapter, high (or low) parameter uncertainties affect the predictive power of models. Posterior predictions are directly related to parameter knowledge and experimental design [79], backing up this propagation phenomenon (see equation 3.35 in Section 3.2.5, p. 84 of Chapter 3). Hence, experimental designs that maximise the prediction uncertainty of a mathematical model can aid the reduction of parameter uncertainties, similarly as the FIM is used in frequentist scenarios [80]. This means that, for an experiment where the model prediction is poor (i.e. high uncertain), the data has the potential to provide more information about parameters than for one where predictions are accurate (i.e. low uncertainty).

To assess the strategy, I successfully predicted the informative content of experiments before performing inference (*a priori*) for the five experimental sets defined for the repressilator model. I applied the different parameter fold changes to the priors (Table 3.11, p. 72 of Chapter 3) for each experimental set (Section 4.1.1, p. 112 of this chapter). Then I used 10,000 prior draws to simulate the model and approximate the prior predictive distributions. I quantified prediction uncertainties using the Euclidean distance between upper and lower percentiles

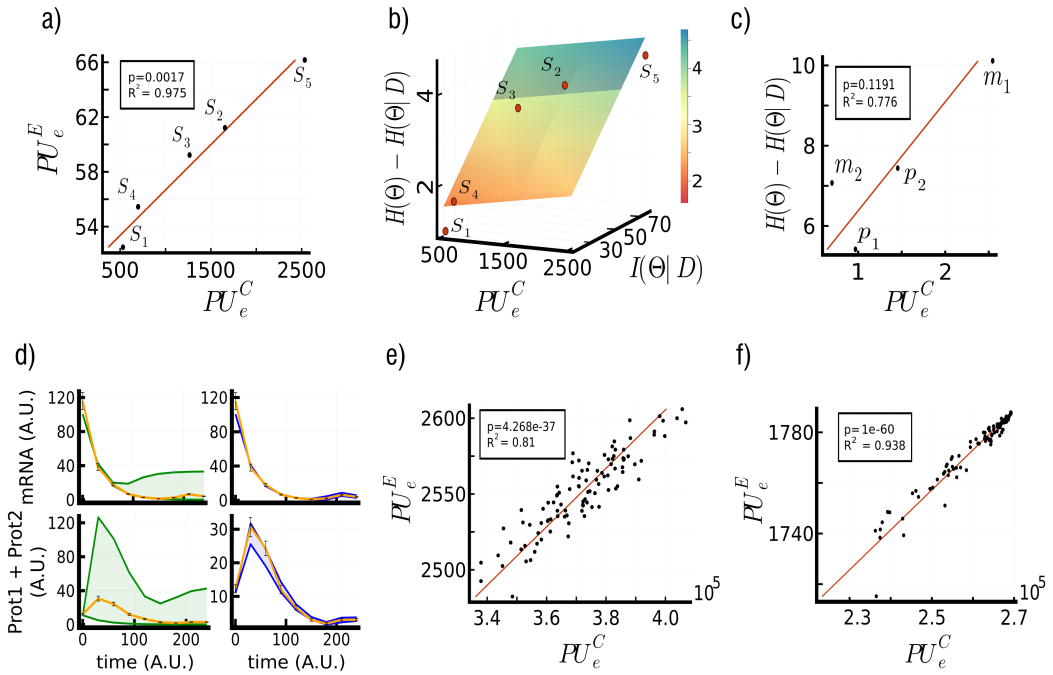


Figure 4.8: Use of prediction uncertainty to assess the *a priori* informative content of experiments. a) Correlation plot between the prediction uncertainty metric computed using the Euclidean distance between the 99.5th and 0.5th percentiles (PU_e^C) and the prediction uncertainty metric using the prediction entropy (PU_e^E) for each experimental set e considered for the repressilator. b) Correlation plot between the repressilator PU_e^C, the information gain $H(\Theta) - H(\Theta|D)$ (extracted from Liepe *et al.* 2013 [56]) and the mutual information $I(\Theta|D)$ (extracted from Liepe *et al.* 2013 [56]). c) Correlation plot between $H(\Theta) - H(\Theta|D)$ and PU_e^C for the Hes1 system. m_1 (mRNA as observable) and p_1 (total protein as observable) represent the experimental set defined in Liepe *et al.* 2013 [56], where the system's initial condition was set to $y_0 = [0.5, 3.0, 2.0]$ for mRNA, cytosolic Hes1 and nuclear Hes1. m_2 and p_2 represent an experimental set where y_0 is set to $y_0 = [100.5, 1.0, 10.0]$. d) Comparison between the prior (green) and posterior (blue) predictions (99.5th and 0.5th percentiles) and mean (orange) and standard deviation (grey) of the pseudo-data used for the inference (bottom panel) and pseudo-data representing the mRNA concentration in cells (top panel). e) Correlation plot between PU_e^C and PU_e^E for 100 pseudo-experiments for the toggle switch model M_3, where each experiment lasts 24 hours and the external inducers concentration changes every three hours to a random value selected between the allowed aTc (0 to 100 ng/mL) and IPTG bounds (0 to 1 mM). f) Correlation plot between PU_e^C and PU_e^E for 100 pseudo-experiments for the orthogonal inducible promoter model. Experimental conditions are the same, with IPTG changing between the bounds 0 to 100 μM.

(PU^C) or computing the prediction entropy for each sampling time (PU^E) as stated in Section 3.2.5 (p. 84) of Chapter 3. The latter was introduced to consider the prediction uncertainty shape, which is lost when using percentiles. Fig. 4.8a shows a comparison between the two metrics, indicating a high correlation and resulting in the same ranking of experiments. Yet, the differences between the two highlight a lack of normality of the prediction uncertainties. I then compared the experimental ranking (using PU^C) with the ones reported in Liepe *et al.* 2013 [56] when using mutual information (*a priori* assessment) and the difference between prior and posterior entropies (*a posteriori* assessment). The results from Fig. 4.8b show a high correlation between the three methods. However, PU^C can recapitulate the same experimental ranking as the *a posteriori* method compared to the mutual information metric from Liepe *et al.* 2013 [56] (discrepancy between sets S_1 and S_4).

To further investigate the uses of prior predictions to assess the informative content of experiments, I predicted the informative content of the two Hes1 system pseudo-experiments (see Section 4.1.1, p. 112 of this chapter) and set two different experiments to switch the ranking between observables. First, I used PU^C to assess the informative content of experiments considering mRNA or total Hes1 protein as observables. As reported in Liepe *et al.* 2013 [56] and Section 4.2 (p. 124) of this chapter, the metric was able to identify mRNA as the most informative observable (Fig. 4.8c, m_1). Note that I had to introduce a normalisation on predictions due to the difference in dynamic range between the two observables.

To assess the potential utility of the method, I generated a set of pseudo-data for the Hes1 transcription factor, manually changing the initial states for the system to make a scenario where observing the total cell protein would be more informative than observing mRNA according to PU^C. The *a priori* ranking of these new two sets of experiments was correct after assessing the *a posteriori* information gain as seen in Fig. 4.8c where the gain in information of p_2 is higher than for m_2. Consequently, the quality of the predictions attained was also higher (Fig. 4.8d). The supplementary Fig. S16 shows the model predictions using the posterior inferred on the second set of experiments using mRNA as an observable (a), and predictions using both posteriors when simulating the original set of experiments (b,c). These results (together with Fig. 4.8d) highlight the higher prediction capabilities of the model when using the total cell protein as an

observable for inference in this alternative experimental scenario.

Finally, I assessed the informative content of experiments when considering models of higher complexity (the genetic toggle switch and the orthogonal inducible promoter) with highly uninformative priors. I observed a high correlation between PU^C and PU^E when assessing 100 random pseudo-experiments for the genetic toggle switch model M_3 (Fig. 4.8e) and the orthogonal inducible promoter model (Fig. 4.8f). However, in both cases, model complexities and uninformative priors resulted in high prediction uncertainties for all experiments, with small variations for PU^C and PU^E as seen for their range of values in Fig. 4.8e ($3.4 \cdot 10^5$ to $4 \cdot 10^5$ and $2.5 \cdot 10^3$ to $2.6 \cdot 10^3$, respectively) and Fig. 4.8f ($2.35 \cdot 10^5$ to $2.7 \cdot 10^5$ and $1.74 \cdot 10^3$ to $1.78 \cdot 10^3$, respectively). This result suggested that, while some experiments might be slightly more informative than others, the scarce prior information hinted at the need of some preliminary experiments to update them. In doing so, PU^C and PU^E would potentially become more useful to discriminate the informative content of experiments without pointing to sub-optimal solutions, similarly to what would happen in FIM-based methods.

4.3.3 Conclusions

Assessing the informative content of experimental designs poses a challenging yet, highly rewarding labour in biology, particularly when employing Bayesian approaches. In this section, I showed that, while frequentist approaches are faster and easier to implement and obtain mean model predictions, Bayesian ones leverage a more precise analysis of inferred parametric noise. Moreover, the noise associated with parameter estimates propagates to model predictions, making Bayesian approaches more accurate when considering parametric uncertainties. I decided to explore noise propagation in Bayesian settings to assess the informative content of experimental designs focused on parameter inference. Moreover, the computational resources of this sampling-based method are lower than other reported Bayesian strategies, only scaling up by the time complexity associated with simulations and the number of samples considered.

I showed that experimental designs that increase model prediction uncertainties tend to increase the amount of information extracted since model predictions are directly related to experimental designs and parameter knowledge. Experiments

predicted with low uncertainty by a model do not provide much additional information and, therefore, can be avoided. This assumption will hold unless suspected that model predictions have low uncertainty but wrongly predict the behaviour of a system. Additionally, since this strategy relies on the amount of knowledge on the parameters of a system, its use can become limited in more complex models with highly uninformative priors. In these scenarios, experimentalists would need to acquire preliminary experimental data to update model parameter knowledge. Nonetheless, the results show an efficient method to assess the informative content of experiments in a Bayesian context, potentially saving time and resources for experimentalists. At the same time, the results highlight the importance of prior selection, the effect this has in Bayesian inference and how this influences the identification of the most informative experiment that will better update this current prior belief.

4.4 Chapter Conclusions

Bayesian statistics provides an insightful approach to parameter inference, where we update parameter knowledge in light of observing new data. In this chapter, I showed how to use Bayesian inference to estimate parameters of models of high interest in synthetic biology, even in cases of higher complexity where I introduce external time-varying inputs. Then, I used a case example, the genetic toggle switch, to spotlight that different experiments provide varying amounts of information concerning parameter uncertainties and model prediction accuracy. I then inferred three models for the same system, showing similar results for all of them, setting up the tedious problem of model selection, further explored in Chapter 5. Then, I proved in a more quantitative manner the information difference between experiments (after inference) using four different examples. Since estimating this information before experimenting is more desirable, I showed that I could use model prediction uncertainty to assess the informativeness of an experiment *a priori* concerning the current prior beliefs. I will further investigate this methodology, both computationally and experimentally, in the following Chapters 6 and 7. Thanks to the usefulness of the different tools presented in this chapter, I integrated as a part of the Julia package described in Chapter 8, BOMBs.jl, the model simulation, pseudo-data generation, Bayesian inference, entropy estimation, uncertainty quantification and information measurement of experiments strategies that I used.

Chapter 5

Optimal Experimental Design for Model Selection: Stability Analysis of a Genetic Toggle Switch

A lack of knowledge of all the specific mechanisms behind a biological system can lead to the development of multiple model candidates to describe this. Hence, in such situations, researchers are left with finding an efficient and adequate strategy to select the model that better represents the system. In this chapter, I will follow up on the results presented in Chapter 4 regarding the genetic toggle switch to investigate the model selection problem. To do this, I will first use commonly used goodness of fit metrics (technical details in Section 3.2.4, p. 83 of Chapter 3) to attempt to identify the best model between three plausible candidates (Section 5.1). Due to the traditional limitations of the strategy, I will propose an optimal experimental approach to generate better experiments to solve the model selection problem (technical details in Section 3.2.7.1, p. 88 of Chapter 3). Finally, I will analyse the stability properties of the genetic toggle switch exploiting the concepts defined in Section 2.3.7 (p. 34) of Chapter 2 to understand better the optimal input profiles obtained and stress the need for the strategy to generate optimal data to better discriminate between model candidates (Section 5.2).

5.1 Optimal Experimental Design for Model Selection

The results of this section were published in ***L. Bandiera†, D. Gomez-Cabeza†, E. Balsa-Canto and F. Menolascina, "Bayesian model selection in synthetic biology: factor levels and observation functions," 8th Conference on Foundations of Systems Biology in Engineering FOSBE 2019 (IFAC-PapersOnLine), 2019, 52 (26) pp. 24-31, doi: 10.1016/j.ifacol.2019.12.231*** *and* ***L. Bandiera†, D. Gomez-Cabeza†, J. Gilman, E. Balsa-Canto and F. Menolascina, "Optimally Designed Model Selection for Synthetic Biology," ACS Synthetic Biology, 2020, 9 (11) pp. 3134-3144, doi: 10.1021/acssynbio.0c00393****.*

Selection of the best model for a system amongst a set of equally plausible candidates is a task commonly shunned due to its difficulty [53] and lack of generic methods. In this section, I will show that, while traditional model comparison metrics are useful for model selection, these should be used with care, depending on the experimental data available. I used three different models for the genetic toggle switch to elucidate the dependence of model ranking on the metric and experiments used for the assessment. Next, I will show the advantages of optimal experimental design to generate experiments that better discriminate models. I defined the optimal experiment for model selection as the one that maximally diverges the predictions of two competing models. First, I analysed the effect of optimisation complexity and model observable selection for the system. Finally, I analysed the best results obtained, showing that the optimisation scheme successfully identifies inducer profiles that exploit stability properties of the system under study, driving model predictions as far as possible; i.e. opposite steady-states.

5.1.1 Traditional Goodness of Fit Model Comparison for the Genetic Toggle Switch

Traditional methods used to select the best candidate model for a system can become insufficient without adequate validation data due to the dependence on the experiment considered and the metric used. To evaluate this effect, I considered the three alternative models for the genetic toggle switch (M_1, M_2 and M_3) in both, Bayesian and frequentist frameworks (Section 4.3.1, p. 131 of

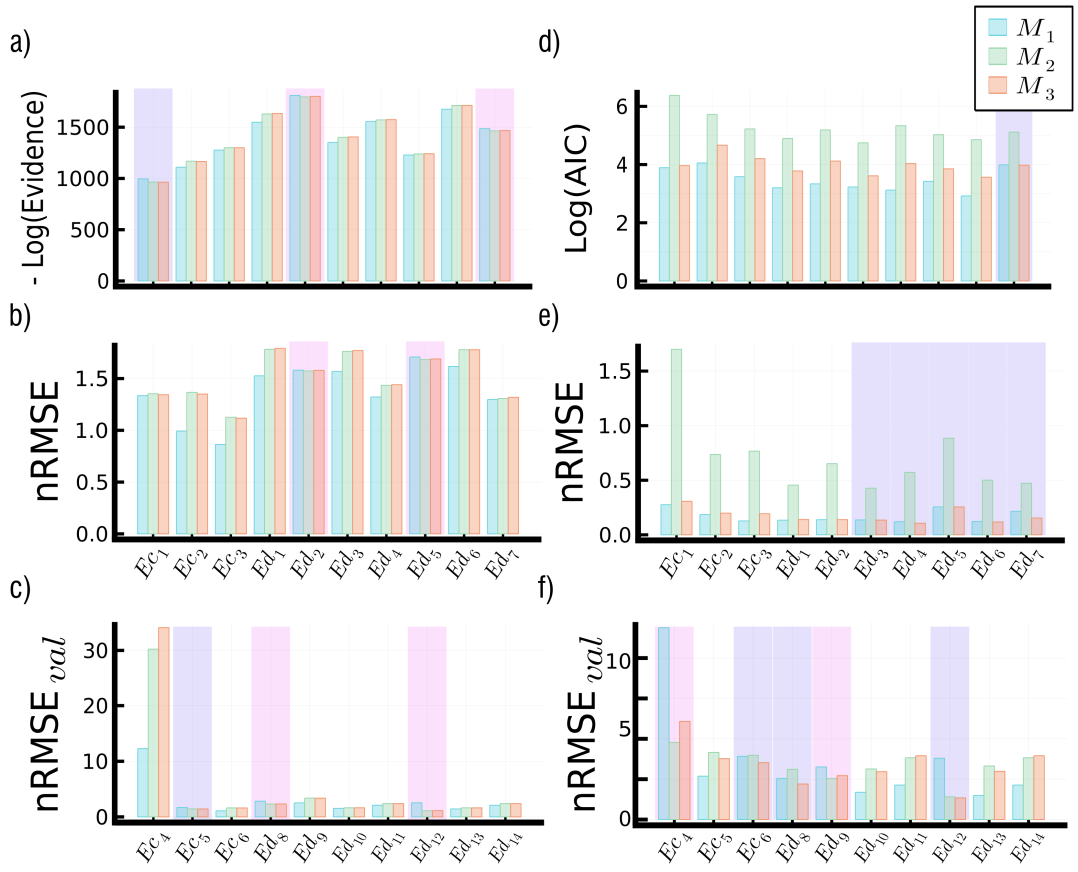


Figure 5.1: Dependence of model selection on the considered experimental scheme and metric used. Bar plots of the minus base-10 logarithm of the approximated model evidence (a) and normalised root mean square error (nRMSE) (b) for the three toggle switch models (M_1, M_2 and M_3) inferred (Bayesian case) on the ten data sets considered from Lugagne *et al.* 2017 [85]. c) nRMSE for the three inferred toggle switch models predicting ten unseen experimental sets from Lugagne *et al.* 2017 [85]. Bar plots of the base-10 logarithm of the Akaike information criteria (AIC) (d) and the nRMSE (e) for the three toggle switch models using the best parameter estimate (frequentist case) to predict the ten data sets considered from Lugagne *et al.* 2017 [85]. f) nRMSE for the three toggle switch models using the best parameter estimate predicting ten unseen experimental sets from Lugagne *et al.* 2017 [85]. Groups highlighted on light purple indicate experiments where model M_2 outperforms the other two, while groups highlighted on dark purple indicate experiments where model M_3 outperforms the other two.

Chapter 4). Note that the three models only differ in the way they describe the transport of inducers into or out of the cells (see Section 3.2.1.1, p. 64 of Chapter 3). Details of all the goodness of fit metrics used in this chapter are described in Section 3.2.4 (p. 83) of Chapter 3. The data used for parameter estimation and goodness of fit evaluation was extracted from Lugagne *et al.* 2017 [85] and defined in Table 3.6 (p. 62) of Chapter 3.

First, we approximated the model evidence ($P(D)$) using the Laplace method [269] for the three models inferred on the ten data sets retrieved from Lugagne *et al.* 2017 [85] (script generation and computations were mostly performed by Dr Lucia Bandiera, University of Edinburgh). Fig. 5.1a shows the $-Log(Evidence)$ results for each model and experiment considered, where lower values designate better models. Seven out of the ten experiments (Ec_2, Ec_3, Ed_1, Ed_3, Ed_4, Ed_5, Ed_6) point to model M_1 as the best candidate. Nonetheless, two experiments favour model M_2 (Ed_2, Ed_7) and one model M_3 (Ec_1), suggesting an inconclusive selection of the best model with 100 % confidence. Yet, according to Jeffreys scale [348], selecting model M_1 as the best one had a decisive statistical support.

In parallel, I computed the Akaike information criteria (AIC) for the three models using the frequentist parameter estimation results. I introduced this analogous study to observe if the experimental dependency in the selection was comparable in a frequentist scenario considering that I am using a different framework but the same experimental data. The results strengthen the case that this strategy heavily relies in the framework, metric and experiment selected instead of the structure of the model alone. Fig. 5.1d shows the AIC results for each model and each experiment considered, where lower $Log(AIC)$ values designate better models. Nine experiments pointed to model M_1 as the best model, with Ed_7 supporting model M_3. In all instances, model M_2 significantly fell behind the other two, with $AIC_2 - AIC_1 > 10$ for all experiments. The supplementary Fig. S17 shows the bar plots of the base-10 logarithm of the Bayesian information criteria (BIC) for the frequentist parameter estimation results. Despite the difference in the penalisation terms of AIC and BIC, both metrics obtained comparable values for all the models and experiments evaluated, suggesting, in this case, a higher dominance for the prediction distance to data terms due to a high accumulative prediction error.

Next, I used the normalised root mean square error (nRMSE) between sample predictions and time-series, also resulting in an experiment dependent ranking. Fig. 5.1b shows the nRMSE results, where lower values designate better models. In a Bayesian framework, the metric, which does not penalise models by their complexity, also identifies model M_1 as the best candidate overall, leaving two experiments favouring model M_2 (Ed_2, Ed_5). While Ed_2 still favoured M_2, in this case, Ed_6 advocated for the model M_1, and there was no experiment supporting model M_3. In a frequentist framework, half of the experiments suggested model M_1 as the best candidate (Ec_1, Ec_2, Ec_3, Ed_1, Ed_2), while the remaining five experiments pointed to model M_3 (Ed_3, Ed_4, Ed_5, Ed_6, Ed_7) as seen in Fig. 5.1e. Overall, the results showed a higher dependence on the experiment to select the best candidate model. Again, a different order and number of experiments pointed to the best model compared to the AIC ranking results. Model M_2 still was considered the worst model in all cases.

To stress the experiment dependence when using traditional metrics to select the best candidate model, I calculated the nRMSE using ten validation experiments from Lugagne *et al.* 2017 [85] (Table 3.6, p. 62 of Chapter 3). The validation experiments also contained three calibration ones (Ec_4, Ec_5, Ec_6) and seven dynamic experiments (Ed_8, Ed_9, Ed_{10}, Ed_{11}, Ed_{12}, Ed_{13}, Ed_{14}). In the Bayesian scenario, parallel to the previous results, model M_1 was selected as the best overall performing model as seen in Fig. 5.1c. However, the selection was, again, dependent on the experiment used to compute the nRMSE, leaving experiment Ec_5 in favour of M_2 and, Ed_8 and Ed_{12} in favour of M_3. In the frequentist scenario, model M_1 was again the best overall performing one, with five out of ten experiments in support as indicated in Fig. 5.1f. Two experiments selected M_2 as the best model and three experiments to M_3. The overall results hinted at the need for a better strategy to confidently select the best candidate model. The supplementary Fig. S18 shows the validation set predictions of the three models when using the best frequentist parameter estimate to visualise the difference in model predictions. The supplementary Fig. S19 shows the validation set posterior predictions of the three models to, again, visualise the difference in model predictions in the Bayesian case.

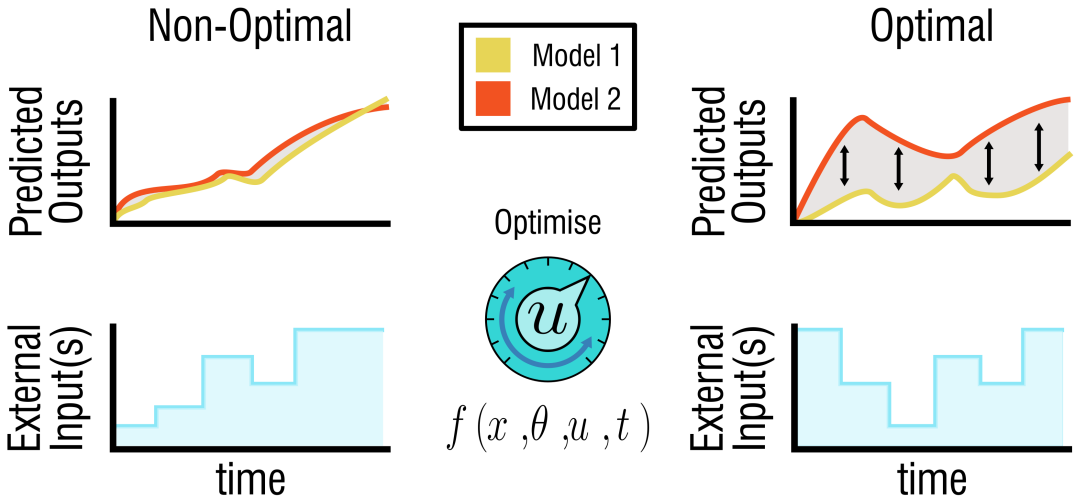


Figure 5.2: Schematic of the proposed strategy for the design of optimal experimental profiles for model selection. The optimiser explores the design space for the considered external inputs (bottom panels) to maximise the distance between predictions of two competing models (top panels). The strategy seeks an experimental scheme that maximises the predictive differences between the two models.

5.1.2 Frequentist and Bayesian Optimal Experimental Design for Model Comparison

To better discriminate between competing model candidates, I proposed the optimal design of the perturbation profile for $u(t)$ that maximises the prediction differences between two candidate models. Fig. 5.2 shows a schematic of this concept and further technical details of the methodology are described in Section 3.2.7.1 (p. 88) of Chapter 3. In the experimental design, I considered the two top-ranking toggle switch models in the Bayesian (M_1 and M_2) and frequentist (M_1 and M_3) frameworks from Section 5.1.1 (p. 141) of this chapter, since the strategy can only compare two models for each experiment. I made the decision on the basis that the resultant experimental data should be incorporated into the inference scheme to update the parameter knowledge iteratively. Looping between inference and experimental design should undergo until the desired goodness of fit metric confidently selects the best candidate model, reducing optimisations from three to one at each iteration ($\binom{3}{2}$, where $\binom{n}{k} = \frac{n!}{k!(n-k)!}$).

Following general experimental constraints of microfluidics platforms, I fixed the experimental duration T_{max} to 24 hours and the sampling time to 5 minutes. The

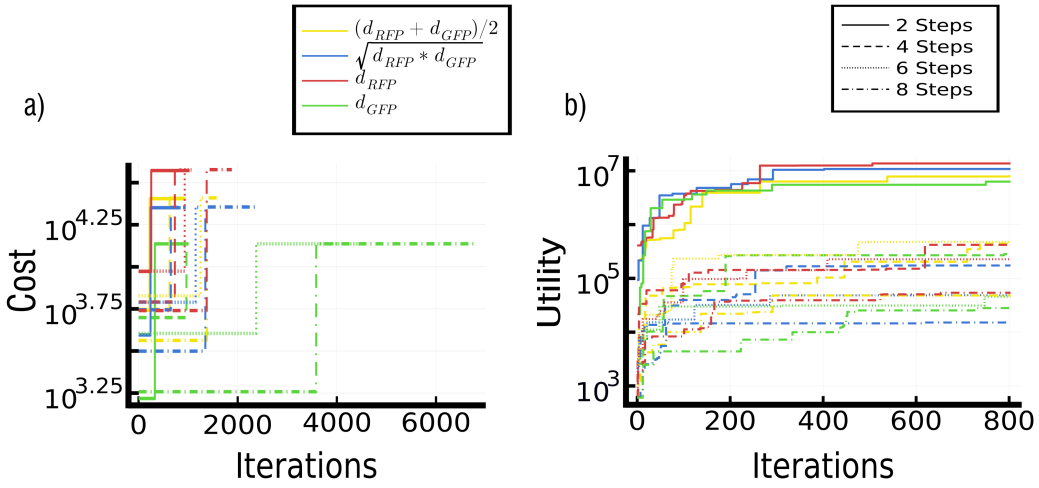


Figure 5.3: Dependence on the convergence curves for optimal experimental design for model selection on the observable(s) and the number of input perturbation steps selected. Convergence curves of the frequentist cost (a) and Bayesian utility (b) functions per optimiser iterations. Colours are assigned depending on the observable(s) considered for optimisation (average, product or single observable distances). Line shapes are assigned depending on the number of perturbation steps NS allowed in the designed experiment ($NS = 2, 4, 6, 8$).

optimisation scheme aimed to design the concentration of the two inducers (aTc and IPTG) administered in NS constant perturbation steps (with $NS = 2, 4, 6, 8$) of equal, fixed duration (step duration $= T_{max}/NS$). In a Bayesian setting, I quantified the prediction distance for the competing models (M_1 and M_2) as the Bhattacharyya distance between posterior predictions. I then used Bayesian Optimisation to explore the inputs design space to maximise the distance metric. In a frequentist setting, since prediction uncertainty was omitted, the meta-heuristic solver enhanced Scatter Search (eSS) explored the inputs design space to maximise the Euclidean distance between the predictions of the competing model (M_1 and M_3).

First, I explored the effect of the optimisation complexity and observables used on the frequentist and Bayesian optimal experimental design (OED) strategies. I increased the complexity of the problems by changing the number of input steps NS in the designed experiments. Due to the dual output nature of the system under study (the genetic toggle switch), I considered a combination of the prediction distance metrics for both fluorescent reporters (average or product of distances) and the metrics considering only RFP or GFP. For comparability and

accessibility of computational resources, I limited the total number of iterations for the meta-heuristic solvers to 800. However, in the frequentist OED, the local solver from the hybrid algorithm eSS proceeded until it reached convergence, sometimes extending the total value.

In frequentist optimisations (Fig. 5.3a), eSS efficiently explored the design space, with a subtle dependence between the prediction distance metric and NS. Higher NS resulted in slightly better cost function values, with the best candidates being for $NS = 8$. When looking at the observable used for optimisation, considering only RFP attained better results than using GFP due to the difference in dynamic range between the two. The difference in dynamic range between RFP (higher dynamic range) and GFP (lower dynamic range) had a higher impact when I considered the average prediction distances, making the product between metrics a more suitable option.

In Bayesian optimisations (Fig. 5.3b), a higher NS resulted in worst utility function values, with the best candidates being for $NS = 2$. Since for optimisations considering $NS > 2$, the best result from $NS = 2$ is present in the design space, I attributed the negative correlation between NS and the utility values to the limited resource allocation for the optimisations. Theoretically, with sufficient computational resources, the result of $NS = 2$ could be achieved in all scenarios. I observed the same effect as in the frequentist OED results on the observable used for optimisation. Consequently, I choose to focus on optimisations considering the product of the prediction distance metrics for both scenarios.

The supplementary Fig. S20 shows the predictions for the competing models for all optimisations performed in a frequentist scenario. The plots indicate how, for each case evaluated, the optimisation strategy focused on driving the models to two opposite steady-states. The supplementary Fig. S21 shows the predictions for the competing models for all optimisations performed in a Bayesian scenario. The plots indicate how the optimisation strategy generally tried to drive the system to two opposite steady-states while also accounting for model prediction uncertainty. Yet, due to computational constraints and the difficulty of the task, some evaluated scenarios (especially at an increasing number of input steps per experiment) failed to explore this option before reaching the maximum number of function evaluations allowed (i.e. 800).

The OED strategy exploited the inherent dynamic properties of the system (i.e. bistability, described extensively in Section 2.3.7, p. 34 of Chapter 2), driving apart predictions (in both frameworks) of competing models to opposite steady-states. According to the prediction distance metrics proposed, the best OED cost (frequentist) and utility (Bayesian) values were one and eight orders of magnitude higher than the most discriminatory experiment used for parameter identification (Ed_7 for the frequentist case, Ed_1 for the Bayesian case). Furthermore, the best OED cost and utility function values were one and seven orders of magnitude higher than the most discriminatory experiment in the validation data set (Ed_{12} for both cases). When looking at the model predictions for the frequentist (Fig. 5.4a) and Bayesian (Fig. 5.4b) OED designs, the optimisers found experimental profiles that drove the competing models to different steady-states (opposite dominance state for RFP-LacI and GFP-TetR). I detected the same result in all optimisations performed, considering different values for NS and distance metrics assessing different observables or combinations of these.

I then analysed the response phase-planes of the different models using nullclines to further understand the optimisation results. I computed the nullclines for the two observables by fixing aTc and IPTG to the values of the first and last steps of the optimised experimental profiles for both frequentist (Fig. 5.4c,d) and Bayesian (Fig. 5.4e,f) results. Indeed, in both cases, the optimisers leveraged on values for the inducers that made the nullclines intersect at opposite regions for RFP and GFP Consequently, each model pointed to a different reporter as the dominant one at steady-state. Note that, in the first step for model M_1 in the frequentist results (Fig. 5.4c, left panel), the nullclines intersect at three different points, indicating a bistable behaviour of the system. Since for all optimisations I set the system to start at the GFP dominant state (overnight using only IPTG), the system remained in the GFP dominant steady-state while model M_3 predicted the system to be at the RFP dominant steady-state (Fig. 5.4d, left panel). The change in input values produces a saddle-node bifurcation in model M_1, pointing that the system will always be GFP dominant at steady-state (Fig. 5.4c, right panel) while model M_3 still predicts the same output (Fig. 5.4d, right panel). Similarly, in the Bayesian case, the nullclines of model M_1 intersect at high GFP value at steady-state for both conditions assessed (Fig. 5.4e), while for model M_2 this happens at high RFP values (Fig. 5.4f).

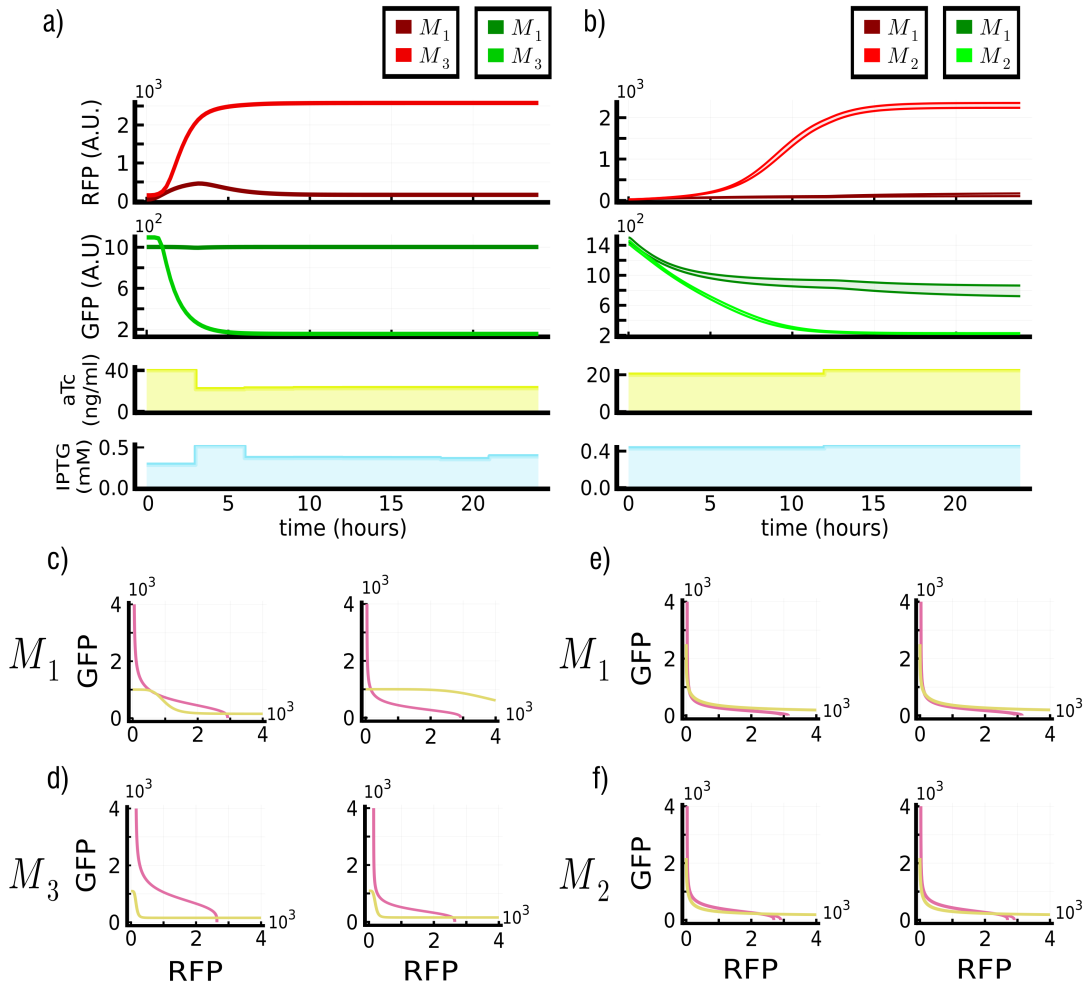


Figure 5.4: Optimal experimental design for model selection identifies an experimental scheme that drives the two competing models for the genetic toggle switch to opposite steady-states. Model predictions of the competing models when applying the optimal input designed in frequentist (a) and Bayesian (b) frameworks. a) Simulation of the competing models M_1 and M_3 using the optimal parameter estimate on the best input designed ($NS = 8$, where NS is the number of input changes in an experiment) from all optimisations considered ($NS = 2, 4, 6, 8$). b) Posterior predictions of the competing models M_1 and M_2 on the best input designed ($NS = 2$) from all optimisations considered ($NS = 2, 4, 6, 8$). Nullclines for the competing models M_1 (c) and M_3 (d) when fixing the inducer concentrations to the first (left) and last (right) input values from the frequentist optimal experimental design. Nullclines confidence region (99.5th and 0.5th percentiles) for the competing models M_1 (e) and M_2 (f) when fixing the inducer concentrations to the first (left) and last (right) input values from the Bayesian optimal experimental design.

5.1.3 Conclusions

Optimal experimental design is a powerful tool revealing the best additional experiments needed to confidently select the best model for a system amongst a set of *a priori* equally plausible ones. In this section, I used three different mathematical models for the genetic toggle switch to show that traditional model ranking strategies can become dependable on the metrics and experiments used for the analysis, in both frequentist and Bayesian frameworks. I showed that, with the currently available information, commonly used goodness of fit metrics (model evidence, AIC, BIC, nRMSE) were unable to confidently select the best candidate model. Dependency on the experiment used to assess the ranking (in both training and validation sets) indicated the need for additional data sets to better inform the decision. For this reason, I proposed an optimal experimental design scheme to determine the best input for an experiment so that the prediction for competing models diverge as much as possible. Independently of the framework used (frequentist or Bayesian), the proposed method successfully designed experimental profiles that exploited the intrinsic properties of the system, driving the competing models to opposite steady-states. Nonetheless, I highlight the need for a case by case prior analysis depending on the system.

For systems with multiple observables, I identified that the difference in dynamic range could impose dominance on one of the observables. I observed that the product between the prediction distance metrics better compensates for the effect. However, if possible, normalisation could be introduced. Furthermore, the complexity of the optimisation should be considered to determine the allocation of computational resources for the task. In such a way, I estimate that the formulation of the models used is another factor to consider, especially in Bayesian settings where, due to the need for sampling-based methods, the computational time for a single function evaluation of the process escalates with model complexity and the number of samples selected. Altogether, our proposed OED for model selection methodology successfully identifies the experiments with the highest discriminatory power. Such experiments should be incorporated into the identification data set, cycling over iterations of parameter updates and OEDs until goodness of fit metrics can confidently select the best candidate model.

5.2 Stability Analysis of the Genetic Toggle Switch developed in *E. coli*

The results of this section were published in ***L. Bandiera†, D. Gomez-Cabeza†, J. Gilman, E. Balsa-Canto and F. Menolascina, "Optimally Designed Model Selection for Synthetic Biology," ACS Synthetic Biology, 2020, 9 (11) pp. 3134-3144, doi: 10.1021/acssynbio.0c00393***.

Bistability (discussed extensively in Section 2.3.7, p. 34 of Chapter 2) plays a key role in cell cycle and signalling pathways, making these systems a point of interest in systems biology to develop analytical mathematical methods to understand them [6]. In this section, I will expand on the OED for model selection results to better understand how different models for the genetic toggle switch infer the bistability (described extensively in Section 2.3.7, p. 34 of Chapter 2) properties of the system from experimental data. First, I computed the bistable properties of the two competing models for the frequentist (M_1, M_3) and Bayesian (M_1, M_2) frameworks at different concentrations for aTc and IPTG. I then compared the results for each pair of competing models, identifying the different concentrations of the inducers that maximise the differences between predictions, i.e. models diverge to opposite steady-states. Next, I showed how I could exploit the bistability analysis to design better experiments for model selection. Finally, I experimentally assessed a scenario where there was a prediction contradiction between competing models in both frequentist and Bayesian scenarios, stressing the need for optimal experiments when pursuing model selection.

5.2.1 Stability Properties of Models for the Genetic Toggle Switch at Different Inducer Concentrations

To further understand the scope of the OED for model selection results and the discrepancies between stability properties from competing models for the genetic toggle switch, I assessed the effect that different combinations of the two inducers exert on the inferred models. First, I discretised the values of aTc and IPTG at 1 % between the allowed bounds, obtaining 101 different equally spaced points for each inducer. For each potential combination of the inducers ($101 \cdot 101 = 10201$), I computed the GFP and RFP nullclines for each competing model from Section

5.1.2 (p. 145) of this chapter by analytically setting the time derivative of the states to 0, in both frequentist and Bayesian scenarios. Next, for each model and each inducer's combination, I distinguished between bistability (nullclines intersect at three points) and monostability (nullclines intersect at one point) points, splitting the latter by GFP or RFP dominance at steady-state. Further details of the analisis is reported in Section 3.2.6 (p. 86) of Chapter 3. The result was a set of stability matrices enclosing the stability properties of the competing models M_1 and M_3 for the frequentist results (Fig. 5.5a, right), and models M_1 and M_2 for the Bayesian results (Fig. 5.5b, left). Finally, to better discriminate divergences between competing models, I superimposed the stability matrices, obtaining the combined matrices for the competing models of the frequentist (Fig. 5.5a, left) and Bayesian (Fig. 5.5b, right) frameworks.

The stability analysis uncovered the different combinations of inducers that drove competing models to opposite steady-states, highlighting how OED leveraged these combinations. In Fig. 5.5a and Fig. 5.5b, I highlighted in blue the inducers concentrations that launched competing models to different steady-states. The black squares on the regions represent the area containing the values for aTc and IPTG obtained in the best optimal designs (Fig. 5.4a and Fig. 5.4b). Frequentist and Bayesian optimisations explored the design space to finally find and exploit the combination of inducers that forked the stability properties between models. Subsequently, I extracted two random combinations of the inducers for each distinct region in the matrices (three for the bistable region for the Bayesian results). Then I computed the respective cost (Euclidean distance) and utility (Bhattacharyya distance) function values.

Optimal designs scored better than the randomly selected points, indicating an advantage of using dynamic experimental designs since the stability analysis only represents steady-state dynamics. However, points sampled in the regions where competing models diverge to opposite steady-states significantly outperformed the rest in frequentist (Fig. 5.5c) and Bayesian (Fig. 5.5e) frameworks. Hence, validating that such input combinations presented a higher value to strengthen model selection at this stage of the process. In the frequentist results, points in the bistable area for model M_1 attain equivalently high cost function values. Since all experimental designs assume that, before the experiment, cells are in the GFP-TetR dominant state, the bistable area for model M_1 in the frequentist results acts as a GFP dominant monostable region.

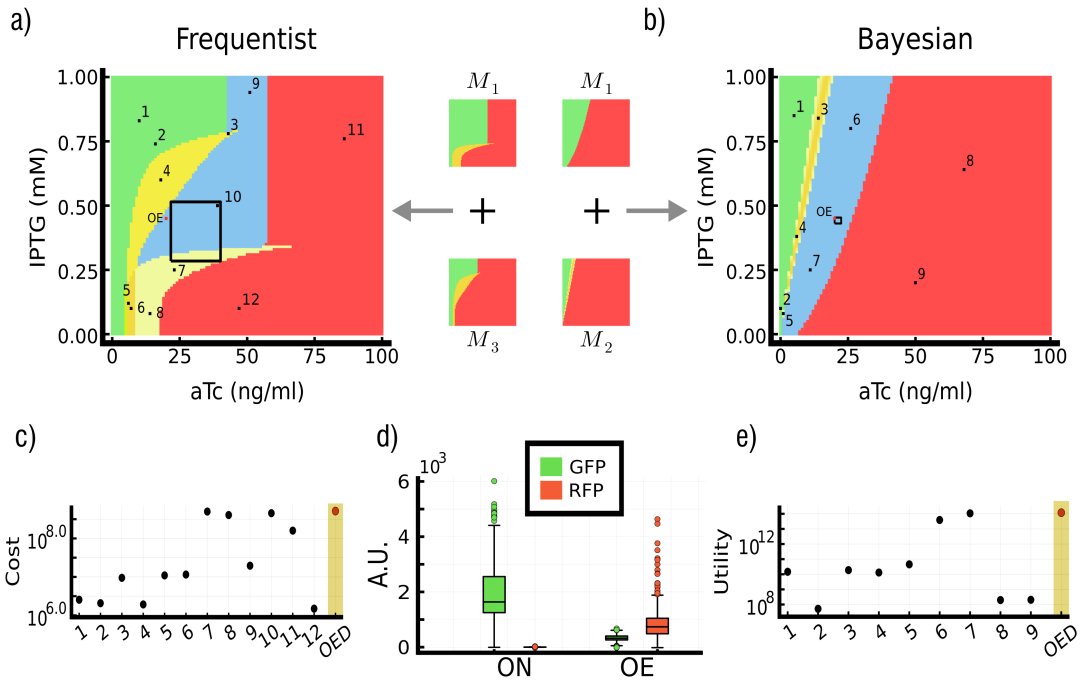


Figure 5.5: Stability analysis of competing models for the genetic toggle switch in frequentist and Bayesian frameworks. a) Heat map of the combined stability matrices of M_1 and M_3 (right) computed using the best frequentist parameter estimate as a function of aTc and IPTG. Colour green and red denotes monostability for both models and the dominant fluorescent reporter at steady-state (GFP and RFP, respectively). Yellow colours indicate bistability for one or both models. Colour blue indicates the discrepancy region between monostable states where GFP is the dominant reporter at steady-state for M_1 and RFP for M_2. b) Heat map of the combined stability matrices of M_1 and M_2 (left) computed using 400 draws of the inferred Bayesian posterior following the same inducer concentrations and colour notations as a). The blue region indicates the discrepancy region between monostable regions where GFP is the dominant reporter at steady-state of M_1 and RFP for M_2. c) Frequentist OED for model selection cost function values for single-step experiments using randomly equally distributed inducer combinations indicated in a) compared to the cost function value for the outperforming best OED result from Section 5.1.2. d) Experimental results of the average cell fluorescence at steady-state (three replicates) after an overnight using maximum IPTG (ON) and the following overnight using an optimal combination (aTc=20 ng/mL, IPTG=0.45 mM) of inducers (OE). e) Bayesian OED for model selection utility function values for single-step experiments using randomly equally distributed inducer combinations indicated in b) compared to the utility function value for the outperforming best OED result from Section 5.1.2.

Experimental data indicates that the best model pointed by the goodness of fit metrics (model M_1) errs on the prediction of stability properties. First, I cultured in LB media the toggle switch *E. coli* strain from Lugagne *et al.* 2017 [85] (see Section 3.2.8, p. 93 of Chapter 3 for further cell culturing details) overnight (ON) with maximum IPTG to ensure that cells were in a GFP-TetR dominant state. Next, I centrifuged and replaced the media (three iterations) with fresh LB media containing the aTc and IPTG concentrations from the first optimal design step from the Bayesian OED results (20 ng/mL and 0.45 mM, respectively). Then, I diluted the resultant culture (1:500 dilution) and let cells grow overnight again (OE). Finally, I imaged ON and OE cultures using fluorescence microscopy, extracting average cell GFP and RFP values of three biological replicates, represented in Fig. 5.5d.

After exposure to the optimally designed input, cells show a steady-state shift, going from GFP (ON) to RFP dominant (OE). The results indicated that, when assessing the stability properties of the system for the specific input combination used in both frequentist and Bayesian frameworks, model M_1 mistakenly inferred the properties of the system. Model M_1 was the best overall model assigned by diverse goodness of fit metrics (Section 5.1.1, 141 of this chapter). Yet, this results highlight that the inferred model is completely wrong in predicting crucial stability properties of the genetic system.The model is in principle capable to assess the correct stability with a different parameterisation. So, this results indicate an insufficient (or potentially inappropriate) amount of data observed to be confident about the fitted model rather than indicating that this one is structurally wrong. Hence, the results stress the need for additional optimal experiments to enforce a better-informed decision. I will further explore the prediction of steady-states, bistability and their use in validation and design of inference constraints in Section 7.2.3 (p. 225) of Chapter 6.

5.2.2 Conclusions

An in-depth analysis of the stability properties of inferred models of bistable systems can guide experimentalists to better design experiments to discern between equally plausible models. In this section, I showed that, when assessing the stability properties of competing models for the genetic toggle switch, critical differences in steady-state predictions arose. Considering that I inferred all

models using the same experimental data, the results suggested that these were not enough, at least for one model, to extrapolate the general behaviour of the system under different experimental conditions. Different concentrations of aTc and IPTG induced saddle-node bifurcations (collision of two equilibrium points) of the nullclines phase plane, resulting in the models predicting if the system behaved as a bistable or monostable (with information on the dominant state) one. While competing models shared some steady-state predictions, I observed significant differences for some inducer concentrations. I showed that these discrepancies between steady-state predictions could become crucial when comparing mathematical models for bistable systems. Nonetheless, this type of analysis excludes information about system dynamics, a constraint easily avoided in optimal experimental design.

Finally, to elucidate the usefulness of optimal experiments regarding model selection, I experimentally assessed a condition that drove any pair of competing models to opposite steady-states. The results showed that model M_1 erred in predicting the steady-state of the system. The result becomes relevant considering that all the goodness of fit metrics used (model evidence, AIC, BIC, nRMSE) selected the model as the best one. The results indicate the need for additional (optimal) experiments to better inform a decision on the model selection problem. Note that I did not assess the same experimental condition starting the system in the RFP-LacI dominant state due to issues on aTc adaptation during overnights. The biological system could behave as a bistable one for the combination of inputs applied instead of a monostable one as predicted by models M_3 (frequentist) and M_2 (Bayesian). Competing models could not be identified as better models, only to have better predictions than model M_1 considering the observed data.

5.3 Chapter Conclusions

OED poses an enticing strategy to generate better data and reduce the experimental burden to solve the model selection problem. In this chapter, I showed how traditional model selection approaches depend on the metric, data and methodology used, deeming them problematic at the early stages of the process. While the situation could improve by ordaining better data to improve parameter estimates for the system (further studied in the following Chapter 7), the results stress the need for a better approach to solving the problem. Hence, I proposed to use OED to generate data focused on solving the model selection problem. The strategy aims to find the experimental scheme that maximises the differences (i.e. maximising the distance between model predictions) between two competing models by challenging these at each step of the process. The results highlight the strategy's success by exploiting inherent properties of the system (i.e. bistability) to maximise the differences between model predictions. I then further investigated the stability properties of the system, proving that the optimisation scheme was exploiting this to design its optimal input profiles by finding regions of the input space that would drive two competing models to two different steady states. A quick experimental assessment of one of these points in the input space highlighted the issue of traditional approaches to the problem at the early stages of the process by indicating that the best overall model was the only one being wrong at predicting a crucial steady-state. Hence, this final results only strengthen the narrative that more, or at least better, experiments are required to better inform the final decision of which model better describes the genetic toggle switch. For this aim, researchers would benefit from an automated platform to efficiently generate and process this data, as I will report in the following Chapter 6. Due to the promising results of the OED strategy for model selection presented, I included a generalised version of the methodology as a part of the Julia package described in Chapter 8, BOMBs.jl.

Chapter 6

Fluorescence Microscopy for Microfluidics in Synthetic Biology: Image Processing and Platform Automation

Computational studies (as shown in Chapters 4 and 5) are crucial for developing mathematical models in biological sciences. Yet, their end goal is to apply them to characterise systems in *in vivo* scenarios. For this reason, researchers need efficient platforms to generate, process and analyse data with ideally automated processes to minimise the experimental burden. In this chapter, I will show how to develop and implement such platforms using fluorescence microscopy (see Section 3.2.10, p. 95 of Chapter 3) and microfluidics (see Section 3.2.11, p. 96 of Chapter 3) to study biological processes in *E. coli* (see Section 3.1.4, p. 59 of Chapter 3) and *S. cerevisiae* (see Section 3.1.5, p. 60 of Chapter 3). First, I will define an image processing platform to process data generated in such scenarios (Section 6.1). Here, I will show how to use this to accurately adjust the combination of different media presented to cells in microfluidic chips and use deep learning to segment cells accurately and quickly from images to develop an automated image processing platform. Then, I will show how to expand this platform to aid process automation (Section 6.2). Here, I leverage the fast in-real-time data processing platform to automatically update model parameters and input designs within one experiment by dividing these into sub-sections. To exemplify the feasibility of these and potential future control applications, I will use two real examples (yeast and bacteria) to test a platform based on frequentist statistics and another in Bayesian ones.

6.1 Microscopy Image Processing for Microfluidic Experiments

Earlier versions of the mixing ratio calibration and the image processing platform for yeast microfluidic experiments were published in ***L. Bandiera, D. Gomez-Cabeza, E. Balsa-Canto and F. Menolascina, "A Cyber-Physical Platform for Model Calibration," In Synthetic Gene Circuits, 2021, pp. 241-265. Humana, New York, NY., doi: 10.1007/978-1-0716-1032-9_12.***

Accurate and fast data processing in microfluidics is a crucial element to allow real-time control of biological circuits (control engineering) [349] or inform potential updates of administered cell inducers in real-time [86] (on-line OED). In this section, I will present an image processing platform to process microscopy images from microfluidic experiments, considering the orthogonal inducible promoter (yeast) and the genetic toggle switch (bacteria). First, I showed the calibration procedure of media mixing ratios for devices containing a Dial-A Wave junction [170]. The junction allowed us to control and change the concentration of inducers to administer to cells at a given time in microfluidic experiments. Afterwards, I showed how I adapted a convolutional neural network algorithm of recent increasing interest, U-Net [345], to segment microscopy images of yeast and bacteria experiments. U-Net provided us with fast and highly accurate segmentations. Finally, I incorporated the segmentation algorithm into an automated on-line image processing platform. I used the platform to process microscopy images of dynamic yeast and bacteria microfluidic experiments. All microscope components and settings are detailes in Section 3.2.10 (p. 95) of Chapter 3.

6.1.1 Calibration of the Dial-A Wave Junction to Assess Media Mixing Ratio in Microfluidic Chips

Calibration of the Dial-A Wave (DAW) junction permitted a controlled dynamic input administration in microfluidic experiments. We added a fluorescent dye, sulfo-cyanine5 carboxylic acid (non-reactive dye), to one of the two syringes

attached to the linear actuators. Since we used this to administer media into the microfluidic device, it allowed us to quantify the mixing ratio between medium of the two different syringes from the linear actuators. The DAW junction allowed mixing between both media in yeast (Fig. 6.1a) and bacteria (Fig. 6.1b) microfluidic chips (fabrication of these detailed in Section 3.2.11, p. 96 of Chapter 3) by changing the height of the actuators holding the syringes thanks to hydrostatic pressure [170]. Media from both syringes (Fig. 6.1a and Fig. 6.1b, two channels on the left) was directed to the DAW junction, where a ratio of each media entered the central channel (Fig. 6.1a and Fig. 6.1b, middle channel on the right) for mixing while the rest was redirected to a waste port (Fig. 6.1a and Fig. 6.1b, upper and lower channel on the right). See Fig. 3.2 of Chapter 3 (sections A) for a full picture of both DAW junctions in the schematic of the device. I build up changes and updates in the calibration platform developed by past members of the group (i.e. base code for actuator control using the Matlab MODBUS communication interface).

Following the methodology described in Ferry *et al.* 2011 [170], I quantified the mixing ratio by measuring the dye fluorescence intensity at the end of the staggered herringbone mixer channel for yeast (Fig. 6.1c) and bacteria (Fig. 6.1d) microfluidic chips. This chip sections correspond to sections B of Fig. 3.2 of Chapter 3. First, we manually adjusted the height of the syringe containing media with the inducers and fluorescent dye (S_1) and the one containing only media (S_2) so to fill the mixing channel only with media from S_2 (0 % mixing ratio). Then, I added a ten minute wait to ensure a stable fluorescent signal during calibration, avoiding any potential local dilutions of the media containing the dye when connecting the tubing to the devices.

We (past members of the group and I) automated the movement of the actuators to gradually increase the height of S_1 and decrease the height of S_2 to the point where the mixing channel contained only media from S_1 (100 % mixing ratio), acquiring a fluorescent image (cy5 channel) after each movement. I removed the gradual move back to the original syringe position due to accumulation of the fluorescent dye in the PDMS walls, which distorted the final results. The automated platform repeated the procedure three times, as represented in the syringes height to time pattern represented in Fig. 6.1e. Next, the platform computed the minimum and maximum fluorescence values to normalise the average region of interest (ROI) fluorescence from the images, obtaining a mixing

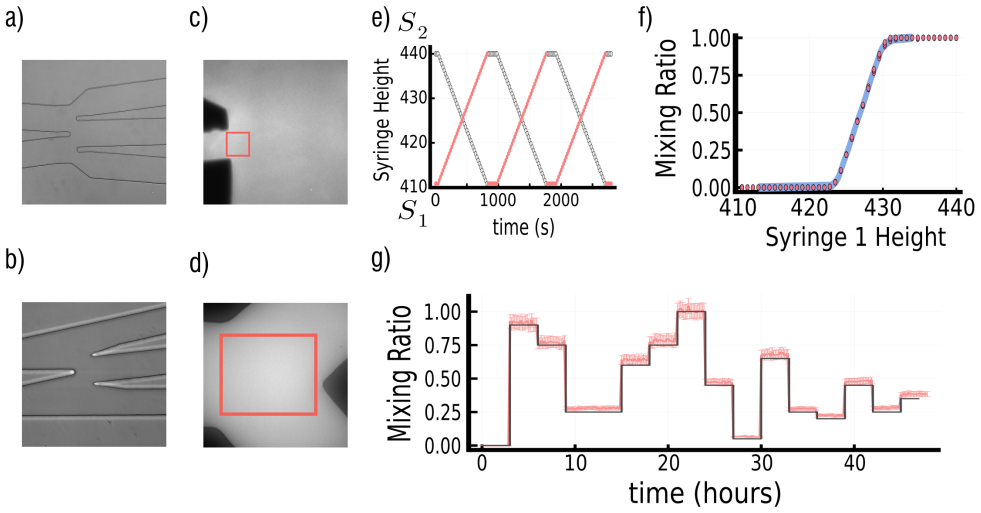


Figure 6.1: Calibration of the Dial-A-Wave (DAW) junction for dynamic experiments. a) DIC image at 20X magnification of the DAW junction of the yeast microfluidic chip MFD005$_a$. b) DIC image at 40X magnification of the DAW junction of the bacteria microfluidic chip. End of the staggered herringbone mixer channel in the yeast (c) and bacteria (d) microfluidic chips. The images were acquired using the cy5 filter to image media containing sulfo-cyanine5 carboxylic acid at 40X (c) and 100X (d) magnifications. The red square indicates the ROI for the calibration procedure. e) Height of the two syringes attached to the linear actuators (with respect to the microscope stage) over time during the progress of a calibration procedure. f) Calibration results of a run in the bacteria microfluidic chip relating mixing ratio of the two syringes to a specific syringe height (the second syringe has the complementary heights within the allowed range). Pink dots represent the measured mixing ratios at the end of the staggered herringbone mixing channel (d). The blue line is the correspondent polynomial (order 9) fit. g) Normalised sulfo-cyanine5 carboxylic acid signal (pink) throughout a 48 hour experiment compared to the desired mixing ratio specified for the experiment (black) using the bacteria microfluidic device.

ratio in percentage. Finally, with the relationship between syringe height and mixing ratio, the platform fitted a high order polynomial to the data (see Section 3.2.11.4, p. 104 of Chapter 3 for more details) to obtain a function relating the two variables as represented in Fig. 6.1f, where the dots represent the experimental data acquired and the line the fitted polynomial. The fitting allowed us to extrapolate unmeasured mixing ratios to apply in an experiment. Fig. 6.1g shows the target (black) and measured (pink) mixing ratios on an experiment performed in the bacteria microfluidic chip. The experiment lasted 48 hours, with changes of the mixing ratio every three hours to randomly selected values, without significant calibration prediction degradation over time (2.5 % average error on target prediction). Due to image acquisition noise, I only allowed mixing ratios to vary at a 5 % discretisation level.

6.1.2 Convolutional Neural Networks (U-Net) for Microscopy Image Segmentation

Convolutional neural networks (CNN) provided a fast and accurate segmentation of microscopy images for bacteria and yeast experiments. I used the CNN U-Net [345] to perform cell segmentation of microscopy images (see more details on the implementations and images used in Section 3.2.11.5, p. 105 of Chapter 3.2.11.5). First, I manually annotated 35 ROIs from different differential interference contrast (DIC) images of experiments considering the yeast microfluidic device. The ROIs contained different levels of cell densities (from 1 to $\approx$ 2000 cells per image) to account for all the possible events in an experiment, i.e. start from one to five budding cells at the beginning, to the cell trap full by the end of the experiment. I performed manual image annotation to generate the ground truths using the software ImageJ [350]. Next, I trained the ImageJ implementation of U-Net [346] for $1.5 \cdot 10^5$ iterations, using a learning rate of 10^{-5}, a validation interval of 150 iterations and keeping 10 images for validation (28.5 % of the total data set). I performed training and usage of the CNN on a remote server using ssh protocol. I employed transfer learning (i.e. using knowledge acquired from a different but related problem) to minimise the computational time of the task, using as the initial set of model weights the ones provided by the U-Net ImageJ plugin [346].

I assessed the quality of the network training by checking the convergence curve

of the cross-entropy loss function and the intersection over union (IoU) metric [346], which scored values above 85 %. Finally, to have an implementation of the software to run in local machines, I adapted the code from Lugagne *et al.* 2020 [347] for our purposes. Due to differences in both U-Net implementations, I re-trained the local network for 30 epochs (batch size = 4) using 2000 segmented images from the ImageJ network implementation as my ground truth. I randomly split the images in training (80 %) and validation (20 %) sets for the process. Then, I tested the resultant network on 400 unseen images, reaching an average IoU of 88.6 %.

U-Net outperformed a state of the art segmentation algorithm, i.e. Cellstar [351], in both accuracy and computational speed. After the first network training (ImageJ U-Net), I compared the segmentation performance to a previously used segmentation algorithm in the research group, Cellstar. First, I selected a ROI from a DIC image full of yeast cells (Fig. 6.2a), representing one of the most complex segmentation scenarios in an experiment. Then, I manually annotated the ROI to acquire the ground truth mask (Fig. 6.2b) to later compare it to the segmentation algorithms results. Finally, I segmented the image using Cellstar (Fig. 6.2c) and U-Net (Fig. 6.2d). The results show the higher performance of U-Net, with more than 10 % improvement in IoU. Additionally, U-Net achieved a 10 % higher pixel classification global accuracy and better identified cell contours. In addition to the higher segmentation accuracy, U-Net showed a faster segmentation time due to its GPU usage compared to Cellstar. The latter relied on CPU computations, resulting in a direct correlation between segmentation time and image complexity assessed by the number of individual cells present in the frame.

I exploited transfer learning using previously trained CNN model weights (from our yeast experiments and Lugagne *et al.* 2020 [347]) to extend the segmentation strategy to bacteria DIC and fluorescence images. I expanded the image segmentation scheme to process bacteria images from the genetic toggle switch [85]. In bacteria frames, I also classified image complexity using cell density. First, I used the set of U-Net weights obtained from yeast images to perform transfer learning and re-training the network with 15 DIC images containing *E. coli* cells with low cellular density (Fig. 6.2e). While the results suggested the need for a better training data set, cell movement between frame acquisitions at low cell densities highlighted the issue of using such segmentations.

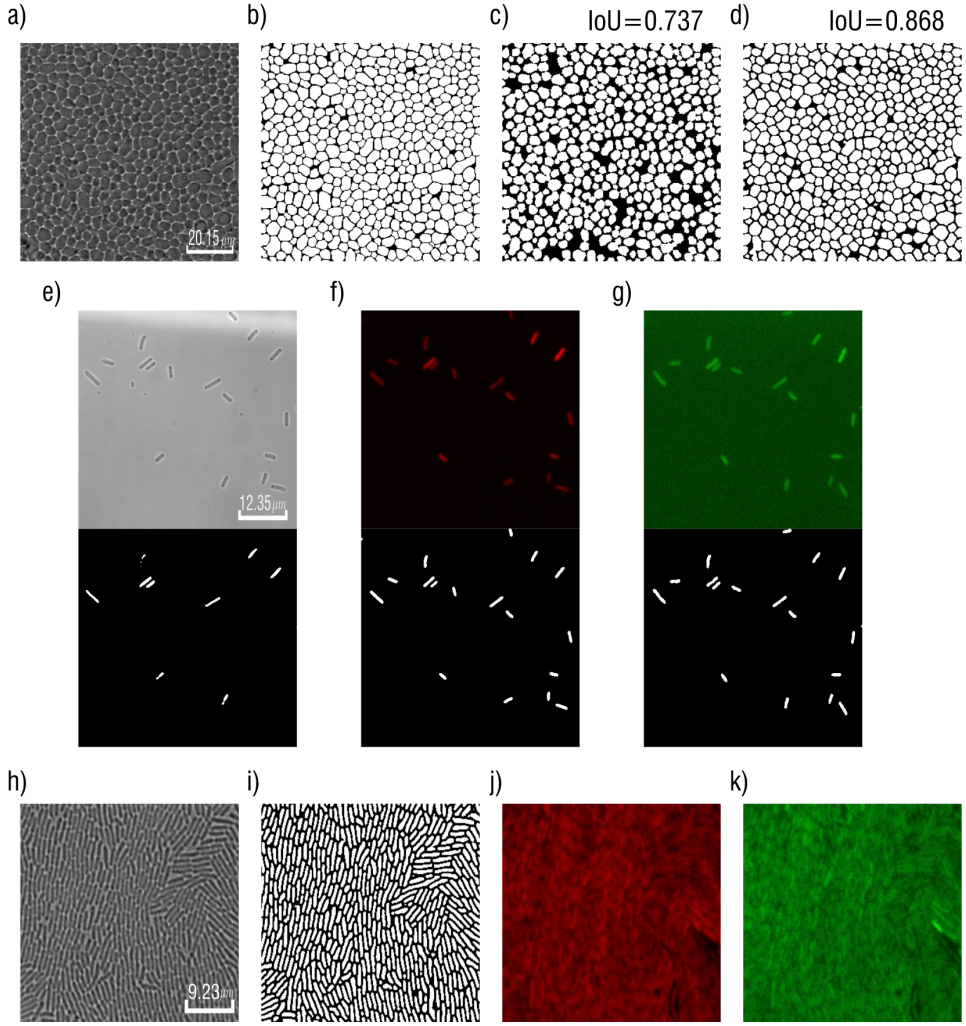


Figure 6.2: Convolutional neural networks (CNN) for image segmentation of yeast and bacteria microfluidic experiments. DIC image section of a microfluidics experiment containing yeast cells (a), compared to the ground truth manual segmentation of the section (b), an automated segmentation using the software CellStar (c) and an automated segmentation using the CNN U-Net (d). Comparison between the same imaged time-frame (with a 0.5 to 1 second delay for filter switch) in DIC (e), RFP (f) and GFP (g) channels (top panels) and the correspondent image segmentation (bottom panels) using the CNN U-Net of a microfluidic experiment containing bacteria cells (low cell density per trap). DIC image section (h) of a microfluidics experiment using bacteria (high cell density per trap) compared to its segmentation mask (i) using the CNN U-Net, and the same time-frame imaged in the RFP (j) and GFP (k) channels.

Consequently, I annotated 26 fluorescence images and retrained the network. I used the strategy thanks to a constant fluorescence presence in the GFP and RFP channels during experiments (basal gene expression).

The CNN achieved an IoU above 80 % when segmenting ROIs from RFP (Fig. 6.2f) and GFP (Fig. 6.2g) channels, circumventing the issue of cell movement at lower cellular densities. Cell movement significantly decreased at higher trap densities (Fig. 6.2h) in the microfluidic chips. I attributed this phenomenon to the reduction of free space between cells and biofilm formation. Hence, segmentation of DIC images became relevant again, reducing the total number of segmentations to half during an experiment. Additionally, cell edges became harder to distinguish in fluorescent channels compared to DIC images. I used the U-Net model and set of weights from Lugagne *et al.* 2020 [347] to segment high cell-density *E. coli* images (Fig. 6.2i) to extract RFP (Fig. 6.2j) and GFP (Fig. 6.2k) fluorescence signals during experiments.

6.1.3 Image Processing Automation for Microfluidic Experiments

To extract quickly and accurately the average cell fluorescence from reporter proteins in microfluidic experiments, I automated the image processing of microscopy images in real-time. I used a custom-made code (developed by a member of the Swain Laboratory, University of Edinburgh) using μmanager to automate the acquisition of microscope images for the selected channels, device positions and equally spaced number of time points. After each time point acquisition from the microscope, in any microfluidic experiments considering the yeast or bacteria chip design, the microscope control program checked for any pre-programmed changes in the height of the linear actuators, and adjusted these to the desired positions. See pictures of the microscope platform and actuators in Fig. 3.1 of Chapter 3.

In parallel, the image processing platform that I designed processed each acquired frame (see methodological details in Section 3.2.11.5, p. 105 of Chapter 3). A general schematic of the functionality and process decision of the platform is represented in Fig. 6.3a. First, the platform reduced all images to the ROIs previously selected by the user. Then, the platform processed background images

to extract the mean and standard deviation of the sulfo-cyanine5 carboxylic acid signal (Section 6.1.1, p. 158 of this chapter). Next, the platform used U-Net (Section 6.1.2, p. 161 of this chapter) to segment all images containing cells and post-processed the resultant masks to index individual elements (segmented cells). The platform extracted the average single-cell fluorescence to compute the cell fluorescence mean and standard deviation for the time frame processed for all the fluorescent channels selected. Summary plots were generated and updated in real-time in the local machine and uploaded to GitHub pages, where a web page displayed the plots and the last time frames acquired. The process repeated itself for each new time point acquisition until all the data for the experiment was processed, at which point, a comma-separated values (.csv) file was generated.

I used the on-line image processing platform to process *in vivo* data for the orthogonal inducible promoter developed in *S. cerevisiae*. Section 3.2.11.3 (p. 101) of Chapter 3 contains the methodological details to set up the microfluidics experiment for yeast cells. Before any experiment, the user had to select the main ROI, including the cell trap with at least one budding cell and a second position also in the cell trap but as far from the main ROI, trying not to include any cells. We needed to acquire the second position inside the cell trap to consider a region with the same height to obtain comparable pixel values and avoid any potential effect of media autofluorescence in any channel.

The platform used the primary position to segment cells (U-Net) and extracted the average cell Citrine fluorescence values during the experiment. The automated code generated some background masks (inverse of the U-Net results) to compute the average background fluorescence in the Citrine channel. The platform slightly dilated each element in the background mask (cells) to account for edge segmentation mistakes and only used the background values if more than 60 % of the ROI pixels corresponded to background pixels. The automated code also segmented the second position in the possible event that some cells would start growing in the ROI. Then, the platform extracted the average background pixel values from the Citrine channel and the average dye fluorescence from the Cy5 channel. If too many cells were present in the main ROI, the platform used the background values from the second one. When the second position becomes crowded with cells, the platform uses the average background pixel values from all the past time points processed.

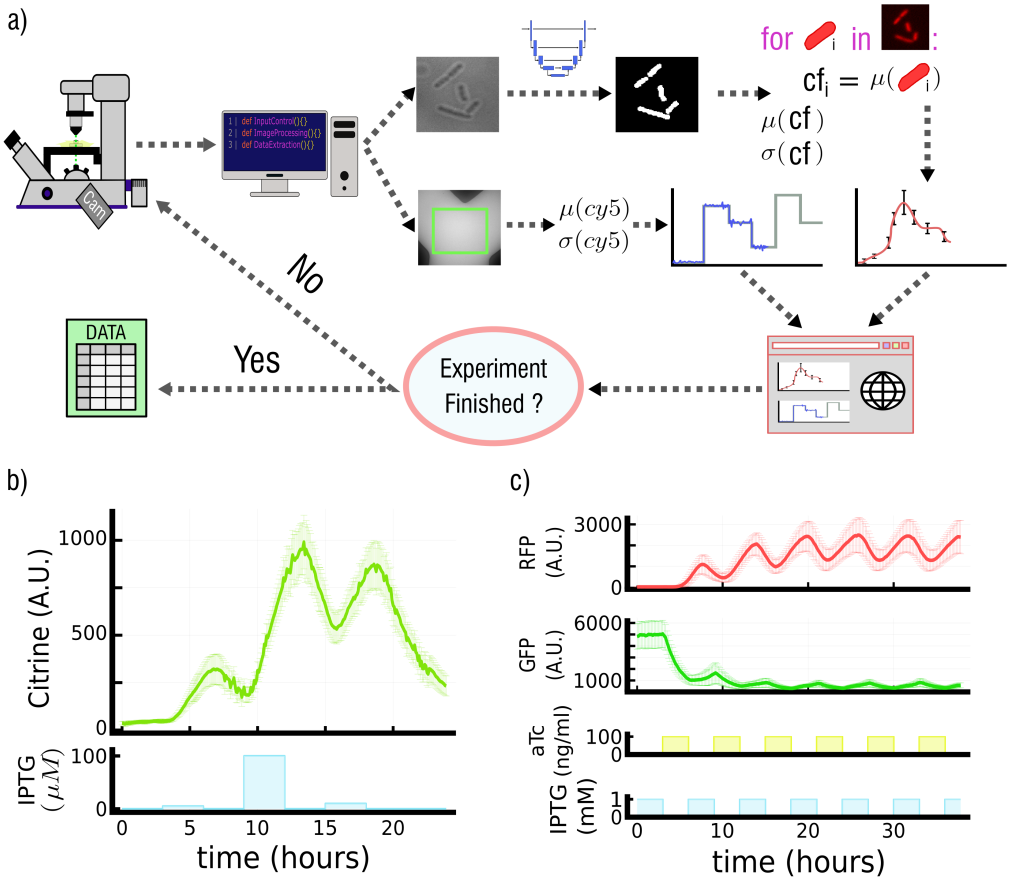


Figure 6.3: Automated image processing of microfluidic experiments in real-time. a) Schematic of the online image processing platform for microfluidic experiments. b) Example of a 24 hour experiment for the orthogonal inducible promoter where images were processed online while the experiment was proceeding. c) Example of a 38 hour experiment for the genetic toggle switch where images were processed online while the experiment was proceeding.

Next, the platform computed the average individual single-cell fluorescence and used this to compute the cell mean and standard deviation Citrine fluorescence for the frame. Fig. 6.3b shows an example of an *in vivo* microfluidic experiment for the orthogonal inducible promoter processed in real-time. The experiment lasted 24 hours, with changing concentrations of IPTG every three hours (IPTG = [0,5,0,100,0,10,0,0] μM) and a sampling time of 5 minutes.

I expanded the on-line image processing platform to process *in vivo* data for the genetic toggle switch developed in *E. coli*. Section 3.2.11.2 (p. 99) of Chapter 3 contains the methodological details to set up the microfluidics experiment for bacteria cells. Due to the device design, the user could select up to 10 different cell traps to image in an experiment. Because of time constraints and to avoid delays in the image processing, I only processed on-line three cell traps in each experiment, with the final data including the average fluorescence of the cells for the three traps (technical replicates). The user could also select an additional position to image the end of the staggered herringbone mixing channel to image the average dye fluorescence from the Cy5 channel.

The platform used the cell trap positions to segment cells (U-Net) and extracted the average cell RFP and GFP fluorescence values during the experiment. The automated code generated some background masks (inverse of the U-Net results) to compute the average background fluorescence in RFP and GFP channels. The platform slightly dilated each element in the background masks (cells) to account for edge segmentation mistakes. If less than 60 % of the ROIs pixels corresponded to background pixels, the platform uses the average background pixel values from all the past time points processed for each cell trap. The platform computed the average background pixel values in the cell traps to avoid effects from media autofluorescence due to the rest of the device features having different wall heights. This background fluorescence processing occurred during the initial 10 hour wait used to let cells stabilise and occupy the whole cell trap. The supplementary Fig. S22 shows the graphical user interface for the image processing platform to highlight the ease of its use for all lab users.

At the beginning of the experiment, since only a few cells were usually present in the traps, the platform segmented the fluorescence channels to account for cell movement between channel acquisitions. This movement can be observed in Fig. 6.2 in changes of individual cell positions for the time frame between the DIC

(Fig. 6.2e), RFP (Fig. 6.2f) and GFP (Fig. 6.2g) acquisitions. Then, after each processed time point and considered cell trap, the algorithm assessed the single-cell distances to the three nearest cells (k-nearest neighbours with $k = 3$). The platform switched to segment DIC images if the average k-nearest neighbours distance was smaller than 27 pixels. We selected the threshold cell distance by computing the average k-nearest neighbours for three separate traps of three different experiments and assessing visually the time-frame at which cells stopped moving and started growing in group formations (biofilm). For each experiment, the platform waited 10 hours for cells to stabilise in the microfluidic chip and divide to fill the cell traps. The platform only used the data from the first 10 hours to compute the average RFP and GFP channel background fluorescence for the rest of the experiment.

Fig. 6.3c shows an example of an *in vivo* microfluidic experiment for the genetic toggle switch processed in real-time. The experiment lasted 38 hours (10 initial stabilising hours not shown) with switching to fixed concentrations of aTc and IPTG every three hours and a sampling time of 10 minutes.

6.1.4 Conclusions

Convolutional neural networks, recurrent image processing procedures and prior biological knowledge can aid the design of fast, accurate and reliable automated platforms to process microfluidic experiments in real-time. In this section, I showed how to calibrate the mixing ratios of microfluidic devices containing a Dial-A Wave junction. The results show an accurate mixing ratio extrapolated from fluorescence data (sulfo-cyanine5 carboxylic acid) without significant degradation over 48 hour periods. The only limiting step of the method was the time delay between a change in mixing ratio and a stable fluorescence signal at the end of the staggered herringbone mixing channel. In our set-up, for both microfluidic devices considered (yeast and bacteria), this time delay was $\sim$30 seconds, scaling the calibration procedure to $\sim$50 minutes. In cases where the delay could become longer (different device designs or media pressure settings), an experimentalist could evaluate fewer height positions or reduce the number of calibration cycles if time is a constraint.

I then used a convolutional neural network of growing popularity (U-Net) to quickly and accurately segment images from microfluidic experiments. I trained

the CNN with a low number of ground truth images, leveraging on transfer learning, hence, reducing the work of manually annotating images. U-Net was faster and more accurate than other segmentation algorithms (i.e. Cellstar), making the CNN the best candidate for an automated image processing platform. However, the computational speed of U-Net is tied to GPU usage, making other algorithms more suitable in the absence of a sufficiently powerful CUDA supported GPU. Finally, I integrated U-Net into my image processing platform for microfluidic experiments. The platform processes microscopy images as they get acquired; segmenting the desired ROIs, extracting average cell fluorescence values, tracking the mixing ratio of the Dial-A Wave junction, and updating remote plots for users to track the progress of experiments. I build the platform as general as possible, needing minimal modifications to process images of different systems, strains, cell types or microfluidic chip designs (e.g. mammalian cells). The only time-consuming requirement when adapting the platform to other cells types would be the re-training of U-Net. Nonetheless, the platform can currently process microscopy images of experiments using *S. cerevisiae* (orthogonal inducible promoter) and *E. coli* (genetic toggle switch).

6.2 An Experimental Platform for On-Line Optimal Experimental Design for Model Calibration

Optimal experimental design (OED) allows for the generation of maximally informative data for traditionally laborious and expensive biological experiments, minimising time and resources allocated to the task. Yet, following the general trend of automation, periodic model and experimental design updates before this finishes allows for better and faster identification of model parameters [86]. This strategy, which has the potential to reduce the number of experiments directed at parameter identification, is commonly known as on-line OED. In this section, I will describe two microfluidics experimental platforms designed to perform on-line OED using frequentist or Bayesian statistics. First, I will describe the general architecture of the two platforms, which heavily rely on real-time image processing (Section 6.1, p. 158 of this chapter). Thanks to having real-time image processing of microscopy images from our microfluidics setup and information about experimental details (e.g. sampling time or initial input design), the platforms can estimate model parameters and update input designs during an experiment. Next, I will show the core functionality of the frequentist platform using the orthogonal inducible promoter developed in *S. cerevisiae* as an example. I will show the simple use of the platform and highlight the different details needed from users to execute such experiments, illustrating two *in vivo* examples of the process. Finally, I will present the Bayesian platform in the same manner and level of detail. In this case, I used the genetic toggle switch developed in *E. coli* as an example, finishing the section with an *in vivo* example of a Bayesian on-line OED executed in our microfluidics platform.

6.2.1 Architecture of a Platform for On-Line Optimal Experiments

I exploited the real-time image processing platform to implement an experimental platform, relying on microfluidics, for on-line optimal experimental design for parameter inference. The methodological details for OED for parameter inference are extensively detailed in Sections 3.2.5 (p. 84) and 3.2.7 (p. 87) of Chapter 3. Fig. 6.4 shows a schematic of the general process the platform follows The

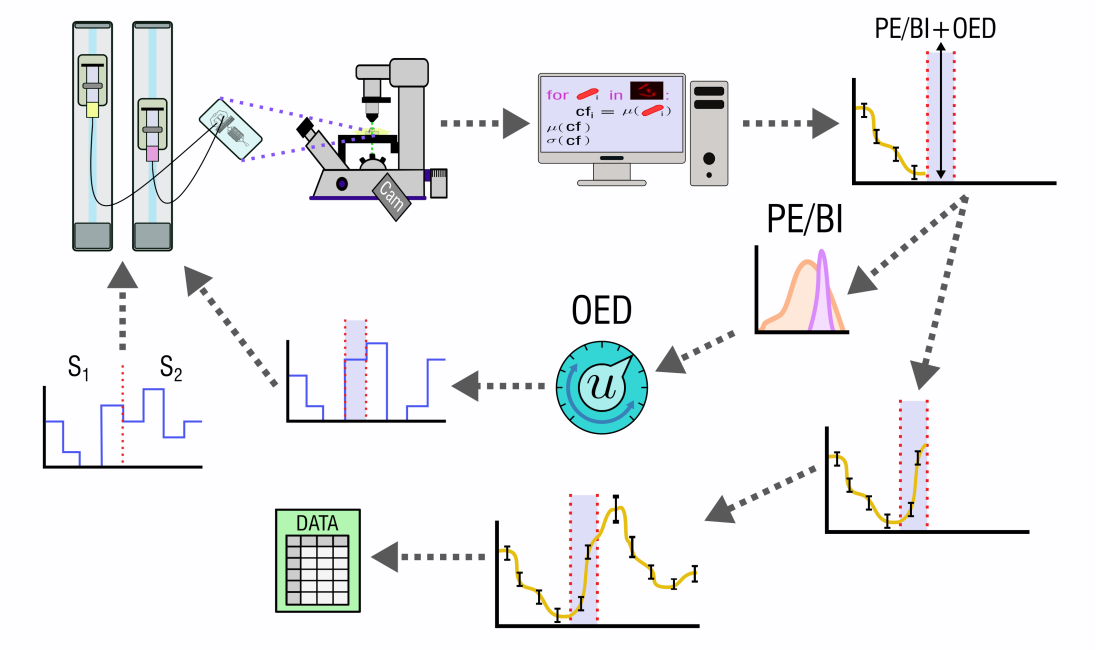


Figure 6.4: *In vivo* on-line optimal experimental design for parameter inference. Schematic of the steps contained in the on-line OED platform for microfluidic experiments. The linear actuators (top left) control the concentration of inducers administered to the cells. After acquiring and processing data for a fraction of the experiment (S_1), the platform estimates model parameters using frequentist parameter estimation (PE) or Bayesian inference and optimally redesigns (OED) the input profile for the following experiment fraction (S_2).

methodological details of this platform are detailed in Section 3.2.11.6 (p. 107) of Chapter 3. First, the user needs to set up a microfluidics experiment, providing an input profile for the linear actuators to follow. Additionally, the online OED platform requires information about the number of equally spaced experiment subdivisions and the desired allocation time for computations (PE/BI and OED) before the ending of each sub-experiment (except the last sub-experiment) and a file containing the model structure for the system. When the experiment starts, the linear actuators provide the predetermined input to the cells at each given time. Meanwhile, the microscope platform acquires images for the predefined imaging channels and chip positions and the image processing platform processes the images in real-time.

On-line computations happen before the end of a sub-experiment. Hence, once the platform processes enough data, this would trigger the first step of the process, parameter identification. The data required for the trigger are the time

points for the sub-experiment minus the ones acquired during computations (predefined). Once the platform infers the given model from the processed experimental data, the program starts the optimisation scheme to redesign the input profile for the following sub-experiment. Once OED finishes, the platform immediately updates the orders for the linear actuators so that these will administer the optimal input design during the upcoming sub-section of the experiment. Note that, while optimisations happen, the imaging platform still acquires and processes data.

If the user intends to perform frequentist on-line OED, the platform uses the Matlab toolbox AMIGO2 [74] to perform parameter estimation (PE) and OED (minimisation of a metric from the FIM). Alternatively, if the user intends to perform Bayesian on-line OED, the platform uses Stan [83] to perform Bayesian inference (BI) and a Julia implementation of the Bayesian OED algorithm described in Section 7.1.2 (p. 191) of Chapter 7. The platform repeats the process for as many sub-experiments as defined. All platforms stop once all the data is acquired and processed, generating a summary file.

Interaction between the independent acquisition, image processing and on-line OED platforms proved essential for the success of experiments. I placed error-handling functions on all the platforms to account for unexpected failures. I built the image processing platform to work independently from the other two to avoid the complete stop of an experiment if the image processing was to fail. Most failures were already taken into account by exception functions, developed over time as these arose (e.g. missing time frames). However, some difficult to automatically fix failures, such as unexpected movement of the device, and hence, incorrect placement of the predefined ROIs, still occurred, although infrequently. Therefore, I developed the platform to perform parameter inference and OED independent of the other two. While after the development and testing of the code, we did not experience any platform failures, the platform isolation ensured that experiments would progress if this failed. I note that the malfunction of any of the two platforms would invalidate the optimality of the resultant experiment. However, image post-processing would allow generating a potential replicate or validation experiment, minimising time and resource losses.

Additionally, while the user predefined the time allocation for all computations (PE/BI and OED), some unexpected computer memory allocation failures could

extend the time of these, especially in Bayesian scenarios. For example, Stan does not allow a maximum time for MCMCs chains to run. For this reason I enforce an initial maximum likelihood estimation of the parameters to start inferences at a good initial guess. Then, I reduce the amount of warm-up samples (between 100 and 500) and allow a maximum of 100 samples to run per chain (and consequently parallel core), hence reducing the computational time for Bayesian inference. I assume an average maximum inference time of three hours according to tests performed with experimental data for the genetic toggle switch. I designed the platform to identify failures and time delays and take them into account during OED, where optimisation will only happen for time steps unimaged, considering the real time allocated for the optimisation. If the platform encountered time delays, the linear actuators would only consider unimaged time steps, ignoring the first(s) input steps from the new design and generating a file at the end of the experiment containing the experimentally administered input profile for users. Again, while the resultant experiment from such failures would be sub-optimal, this could still be used as a validation experiment, avoiding time and resource losses.

Consequently, to minimise user interaction with the platform, I designed the on-line OED platform to automatically extract the maximum amount of information from the other two by generating temporary files in every critical process step. The temporary files contained information about the image acquisition (i.e. sampling time, number of positions and channels imaged, length of the experiment and input profile) and processing (i.e. processed fluorescence data) platforms. This automation reduced the necessary information introduced by the user before the start of an experiment to the optimisation details and the inducer concentrations used for overnight cultures.

6.2.2 *In vivo* Frequentist On-Line Optimal Experimental Design Using the Orthogonal Inducible Promoter Developed in *S. cerevisiae*

I designed the frequentist on-line OED platform with maximum flexibility, usage simplicity and user adaptability. Fig. 6.5 shows the graphical user interface displayed to the user at the beginning of an experiment with the different tuneable platform settings. The user can type in the field “Identifier” a string to

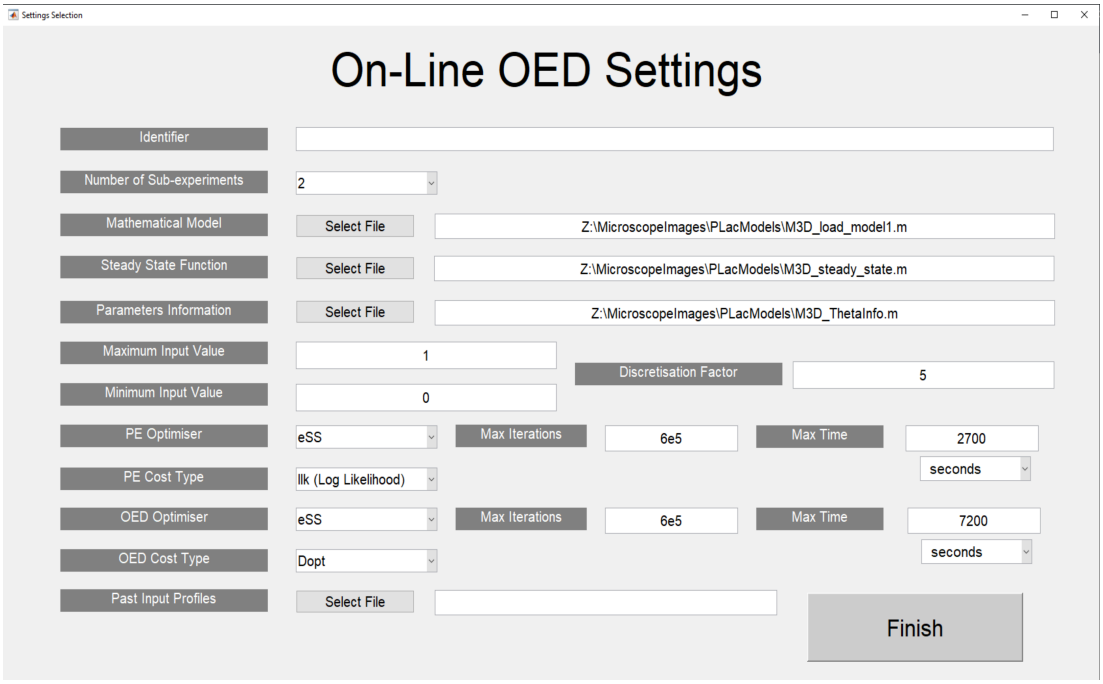


Figure 6.5: Frequentist on-line optimal experimental design platform. Image of the graphical user interface (GUI) with the default settings for the frequentist on-line OED platform in our microfluidics setup.

be added to any temporary files generated by the platform. The number of sub-experiments option allows the user to select a value between two to ten. The following three fields require the path to Matlab functions containing information about the mathematical model describing the system. The field "Mathematical Model" requires a function returning the model structure for AMIGO2 [74]. The field "Steady State Function" requires a function that returns the analytical steady-state for the mathematical model so this can be updated during parameter estimation for each parameter vector evaluated. The field "Parameters Information" requires a function that returns a parameter vector estimate and parameter bounds in the structure needed for AMIGO2.

The maximum and minimum input values let the user adjust the inducer bounds used for the optimisation. The "Discretisation factor" field allows input discretisation during optimisations, given in percentage between the inducer bounds. For parameter estimation, the user can select the different global optimisers (eSS, DE, SRES) and cost function (least-squares or log-likelihood) included in AMIGO2. Additionally the user can choose the maximum time (default set to 45 minutes) and the number of iterations (default set to $6 \cdot 10^5$

iterations) allowed for the optimisation. The platform runs in parallel as many PE optimisations as cores the computer has, selecting the best estimate as to the one with the lowest cost function.

For optimal experimental design, the user can choose different MINLP optimisers (eSS, GA, NOMAD, VNS) and the FIM metric to use for the optimisation (D-opt, E-opt). Additionally, the user can select the maximum time (default set to two hours) and the number of iterations (default set to $6 \cdot 10^5$ iterations) allowed for the optimisation. The platform adds a delay of 15 minutes to the total computations time (PE and OED) to ensure the finalisation of optimisations by the time the re-designed steps start. The last field, "Past Input Profiles", allows the user to introduce a function returning experimental data (mean and standard deviations), sampling times and input designs from past experiments, so new computations for sequential OED include this information.

I successfully executed frequentist on-line OED experiments for the synthetic orthogonal inducible promoter developed in *S. cerevisiae* in a microfluidics setup following the protocols described in Sections 3.2.9 (p. 95) and 3.2.11.3 (p. 101) of Chapter 3. We set experiments to last 24 hours, with a sampling time of 5 minutes (288 total samples), including two sub-experiments of 12 hours each (Fig. 6.6). We defined constant steps of three hours for IPTG and a total computational time of three hours (45 minutes for PE, 2 hours for OED and 15 minutes delay) done during the last input step of the first sub experiment. Experiments started normally, where I set IPTG to [0,5,0,100] μM (Fig. 6.6a) and [0,40,80,25] μM (Fig. 6.6b) for the first experiment sub-section and to 0 μM for the whole second sub-experiment as a control. Once the platform acquired and processed 108 sampling points (144 for the total sub-experiment minus 36 for the computational time), the code triggered parameter estimation. With the new parameter estimates, the platform used AMIGO2 to perform optimal experimental design using D-opt (Fig. 6.6a) or E-opt (Fig. 6.6b) to re-design the second sub-experiment. Then, the platform sent the new design to the linear actuators' controller, ensuring that these executed the optimal input design. Once the experiment finished, we updated parameter estimates using all the data from the experiment obtaining the final parameter estimate for the sequence.

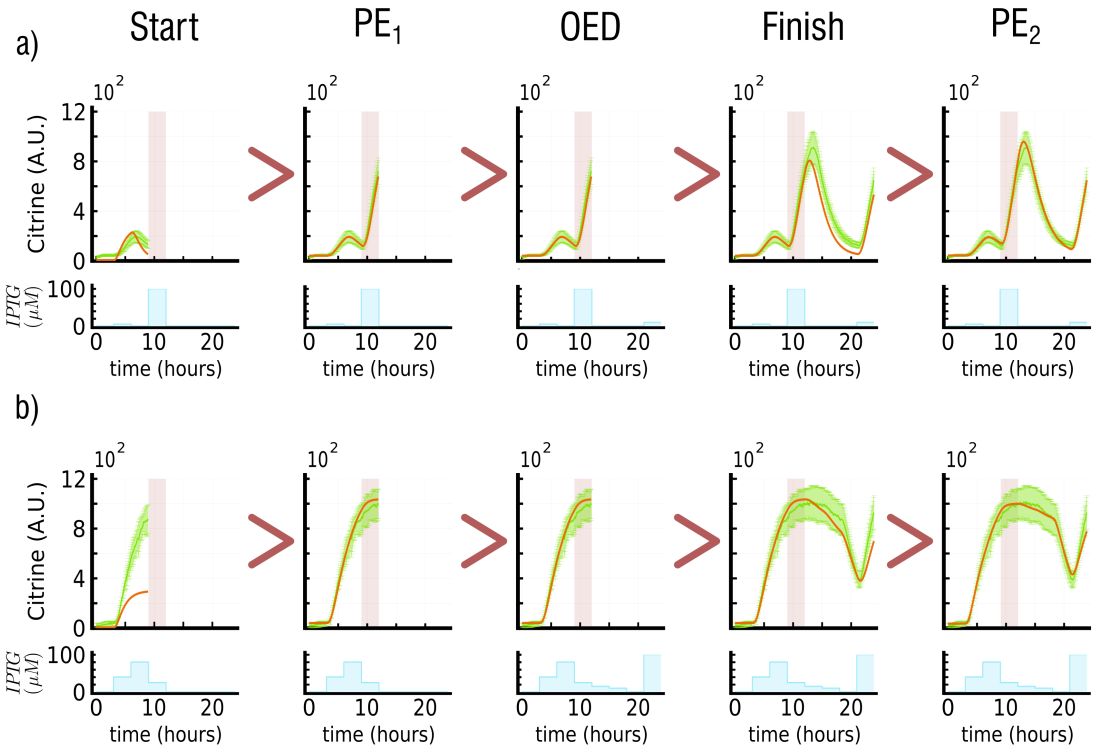


Figure 6.6: *In vivo* Frequentist on-line optimal experimental design for the orthogonal inducible promoter developed in *S. cerevisiae*. Microfluidics experimental data (green) and simulations using the current best parameter estimates (orange) of two different experimental profiles (lower panels) for IPTG (a, b). The input profile introduced before starting the experiment only considered the concentrations of IPTG for the first half of the experiment, setting the inducer to 0 μM for the entire second half as a control. IPTG concentrations changed every three hours of constant steps, with IPTG_1 updating from [0,5,0,100,0,0,0,0] μM to [0,5,0,100,0,0,0,0,10] μM (a) and IPTG_2 updating from [0,40,80,25,0,0,0,0] μM to [0,40,80,25,15,10,0,100] μM (b). We allocated three hours (low opacity red area) for PE_1 (45 minutes) and OED (2 hours) and performed a second PE_2 after the platform acquired and processed all data.

6.2.3 *In vivo* Bayesian On-Line Optimal Experimental Design Using the Genetic Toggle Switch Developed in *E. coli*

After testing the functionality of the frequentist on-line OED, I modified the code to build a Bayesian counterpart. Due to the high computational complexity of the problem in a Bayesian setting, I changed all the platform scripts to perform simulations, MLE, OED, data extraction and file handling to use the Julia programming language. The platform used the probabilistic programming language Stan to perform Bayesian inference. Since the image processing platform runs in Matlab, the on-line OED platform accessed all the saved data files using MAT.jl. Following the same architecture of the frequentist platform, I designed the Bayesian one to include the same degrees of simplicity, flexibility and user adaptability. Fig. 6.7 shows the graphical user interface (GUI) displayed at the beginning of an experiment with the different tuneable Bayesian platform settings.

In the platform GUI, users can type in the field "Identifier" a string to be added to any temporary files generated by the platform. The number of sub-experiments slide bar allows the user to select a value between two to ten. However, I only tested cases with two sub-experiments due to the high computational demand of the task. The following field ("Model Dictionary") requires a Julia script returning a dictionary with information about the mathematical model to use, external inducers, solver specifications, estimated initial state values, observables specification, parameters to infer, priors and parameter bounds (if any). I follow the dictionary structure and syntax set by my Julia package, BOMBs.jl (package presented in Chapter 8). Next, users can define the maximum and minimum concentrations for up to two inducers and their desired discretisation factor for the optimisation section. The platform interprets the order for the inducers as the one set by the user inside the Julia dictionary. This information will be used by the platform during the optimisation step. Note that, for experimental designs such as the ones used for the genetic toggle switch (Section 7.2, p. 206 of Chapter 7), the platform will only use the maximum inducer concentration values since these are fixed for the whole duration of the experiment.

Since Bayesian statistics usually entail high computational costs, the platform

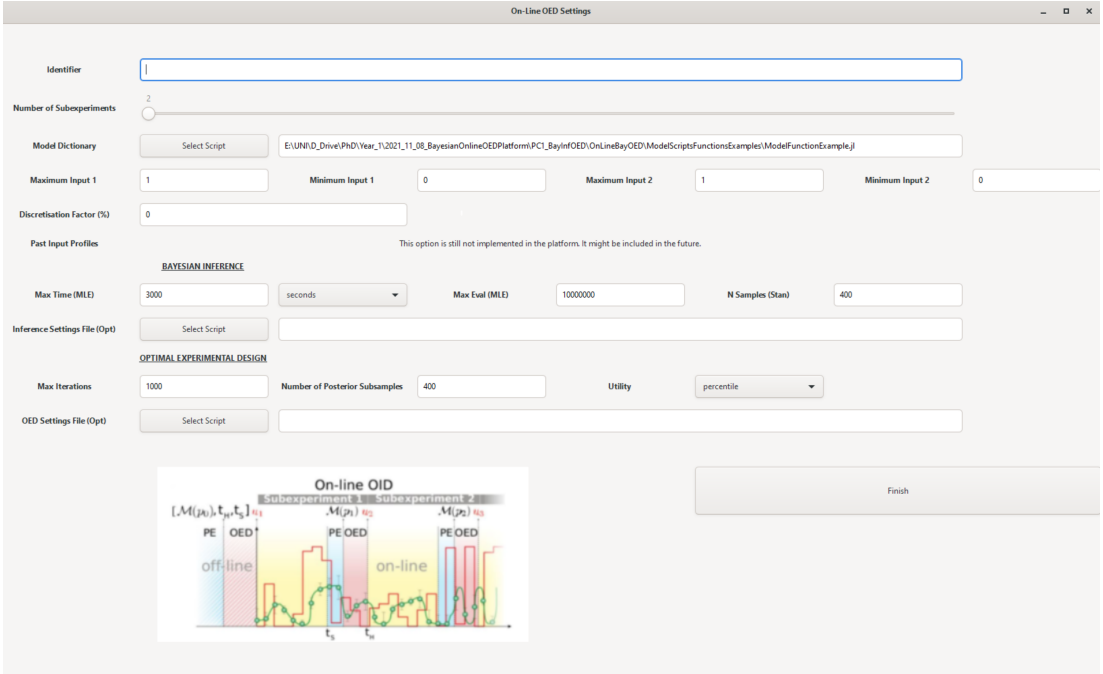


Figure 6.7: Bayesian on-line optimal experimental design platform. Image of the graphical user interface (GUI) with the default settings for the Bayesian on-line OED platform in our microfluidics setup.

includes a preliminary maximum likelihood estimation (MLE) step before a low-sample inference. The MLE ensures that the MCMC chains start at a good initial parameter estimate, reducing the required warm-up samples (set to 250 as default). Users can define the desired maximum time allowed for the task and the maximum number of function evaluations. The platform uses the probabilistic programming language Stan to perform Bayesian inference. Users can select the total final number of posterior samples in the field "N Samples (Stan)". Additionally, I designed the platform to easily modify desired Stan settings. Users can introduce in the "Inference Settings File (Opt)" field a Julia script containing a function that returns a dictionary with the different settings to change. Some examples are the number of warm-up samples, the number of chains (set to 4 by default) or the target average proposal acceptance probability. I followed the dictionary structure and syntax set by my Julia package, BOMBs.jl (package presented in Chapter 8). The package assumes a maximum inference time of three hours (set after multiple test runs). However, inferences might run for longer due to the absence of a time stop trigger in Stan.

I implemented the platform to follow the Bayesian OED strategy described in

section 3.2.7 (p. 87) of Chapter 3 (and further used in 7.1 (p. 186) and 7.2 (p. 206) of Chapter 7). Users can define the maximum number of iterations for optimiser (Bayesian optimisation), the number of parameter samples to use for the simulation and the metric to assess prediction uncertainty (percentiles or entropy). The platform assumes a maximum of one hour for the optimisation. However, if the previous inference expands for longer than the allocated time, this additional inference time will be subtracted from the optimisation one. Hence, to prevent the platform from neglecting the optimisation step, the platform enforces a minimum of ten minutes for the OED section. If the allocated time is inferior to ten minutes, the platform uses additional computational time extracted from the beginning of the sub-experiment to design, checked in one-hour windows. This computational time expansion implies that the beginning of the sub-experiment (used for delayed computations) will follow the original design. The platform adds a delay of 15 minutes to the total computations time (maximum likelihood estimation, Bayesian inference and OED) to account for potential short delays caused by script compilations and operations. The platform accounts for experimental designs following the constraints described in Section 7.2 (p. 206) of Chapter 7 (genetic toggle switch experiments). The optimisation scheme only needs to find the optimum switches between the two inducers in three-hour windows since their concentrations become set by the initial design. For this reason, the platform evaluates all potential experimental designs (2^n designs, where a state of present or absent is evaluated in n remaining intervals of 3 hours) as long as the total amount to assess is less than 500. Otherwise, the platform will follow an optimisation scheme using Bayesian optimisation. Finally, users can introduce in the "OED Settings File (Opt)" field a Julia script containing a function that returns a dictionary with different optimisation settings to modify. I follow the dictionary structure and syntax set by my Julia package, BOMBs.jl (package presented in Chapter 8).

I successfully tested the platform by performing a Bayesian on-line OED microfluidics experiment for the synthetic genetic toggle switch developed in *E. coli* following the protocols described in Sections 3.2.8 (p. 93) and 3.2.11.2 (p. 99) of Chapter 3 and using the mathematical model M_3 described in Section 3.2.1.1 (p. 64) of Chapter 3. Following the same experimental constraints used for all microfluidic experiments regarding the genetic toggle switch (Section 7.2, p. 206) of Chapter 7, experiments lasted 38 hours, with a sampling time of 10 minutes (228 total samples), including two sub-experiments of 19 hours each (Fig. 6.8). I

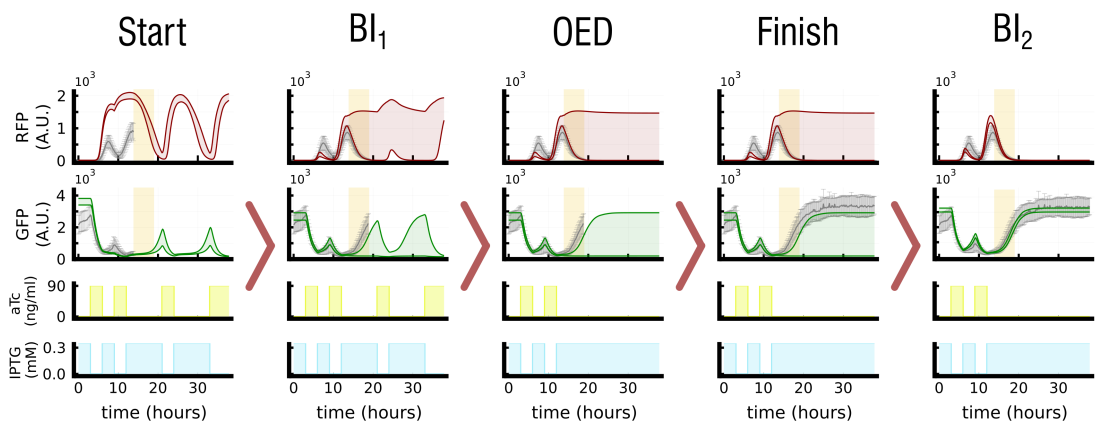


Figure 6.8: *In vivo* Bayesian on-line optimal experimental design for the genetic toggle switch developed in *E. coli*. Microfluidics experimental data (grey) and simulations for the system observables (RFP-LacI and GFP-TetR) using samples from the current parameter posterior. The input profile divides the experiment into three-hour steps where only one of the two inducers is present at a given interval. The initial design randomly selects the presence of aTc (90 ng/ml) or IPTG (35 mM) for a given time step interval. The experiment was divided into two fractions of equal time. The first half was used to collect data, perform Bayesian inference (BI_1) for an initial update of the parameter priors and compute the optimal experimental design for the second half of the experiment considering the current parameter knowledge update (OED). The time allocation for the computations is indicated by the low opacity orange areas. Then, the model was inferred using the whole experiment (BI_2) once all experimental data was collected and processed.

divided the experiment into three-hour steps in which only one of the two inducers (aTc or IPTG) got exposed to the cells at a constant concentration. Then, I randomly selected the inducer concentrations and their presence at a given step. I set the maximum computational time to 5 hours, divided between MLE (45 minutes), Bayesian inference (3 hours), OED (1 hour) and additional time to account for delays (15 minutes). The platform divided the experiment into two 19-hour sub-experiments and accordingly started computations 5 hours before the end of the first one, once all the required data (84 time points) was processed. After the trigger, the platform inferred the model parameters using Stan (Fig. 6.8 BI_1) and assessed the optimal design for the second sub-experiment (Fig. 6.8 OED). Since the concentration of aTc and IPTG remained constant for the duration of the experiment, the platform evaluated all

potential 3-hour step switch combinations ($2^7 = 128$) and selected the one that provided maximum prediction uncertainty. Next, the platform dispatched the new optimal design for the second sub-experiment to the linear actuators' controller for the remaining part of the experiment. Once the platform collected and processed all the data, I inferred the model parameters using the entire data set and the originally defined prior (Fig. 6.8 BI_2). I used all experimental data for the final inference due to the reduced sample number of the transitory BI_1, which is only used for a temporal update to increase the informative content of the experiment.

6.2.4 Conclusions

On-line OED is a promising strategy with the potential to streamline parameter identification by leveraging data processing automation. In this section, I showed the implementation of two on-line OED platforms (using frequentist or Bayesian statistics) integrated with our microfluidics set-up. Thanks to the in-real-time image processing of the acquired microfluidics data, I automated the inference of processed data and the design of potential experimental sections *in vivo*. I showed how a prospective on-line OED platform needs to work together with a data-processing one. The platform needs to know the number of experimental sub-divisions and the allocated computational time. With this information, the platform can trigger parameter identification and OED once the image processing platform processes enough data for a particular experimental section. Such computations can follow frequentist or Bayesian statistics. For this reason, I designed a platform for each strategy. I built such platforms to provide as maximum flexibility as possible, allowing users to easily change the mathematical model to use and adjust parameter identification and OED settings. The frequentist on-line OED platform performs parameter estimation by minimising the mean squared error or log-likelihood between data and model simulations and OED using a FIM-based optimality criterion. The platform runs in Matlab, and I use the toolbox AMIGO2 [74] to perform all computations. The Bayesian on-line OED platform performs Bayesian inference using the probabilistic programming language Stan [83] and OED by maximising the prediction uncertainty of a model. The platform runs in Julia leveraging my package BOMBs.jl (presented in Chapter 8). After highlighting the architecture and features of each platform, I demonstrated their functionality by showing *in vivo* microfluidic experiments using the orthogonal inducible promoter developed in *S. cerevisiae* (frequentist

platform) and the genetic toggle switch developed in *E. coli* (Bayesian platform).

6.3 Chapter Conclusions

Fast, reliable and automated data processing platforms are essential to understand biological processes, especially when studying systems at a (sub-)cellular level. Furthermore, one can build layers of complexity on top of these, such as performing process control, to further understand and use these biological systems. In this chapter, I presented a platform to process fluorescence microscopy images for microfluidic experiments focused on studying yeast and bacteria cells. The results show how current technical advancements in machine learning (i.e. convolutional neural networks) are ideal for efficiently handling a traditionally slow-to-solve and complex problem, cell segmentation. Hence, I developed a fully automated image processing platform thanks to this deep learning technique. I then used the platform to process yeast and bacteria images in real-time in microfluidic experiments, processing these as soon as they were acquired. Thanks to the advantages (e.g. fast, accurate, easy to use, easy to modify) of the platform, I then used it for all the experiments performed and reported in the following Chapter 7. Moreover, I showed how all these techniques could be transferred to other scenarios, such as the calibration of media mixing in microfluidic devices thanks to fluorescent dyes. Since automated data processing is one of the major bottlenecks to control systems or performing on-line optimal experimental design, I leverage the image processing platform to implement the latter experimentally. First, I showed how I implemented this new on-line OED platform using frequentist and Bayesian statistics. Then I used both platforms to successfully perform on-line OED, demonstrating the experimental feasibility of automatically updating model parameter estimates and input designs while a microfluidic experiment is still ongoing. These promising results allowed me to use the Bayesian platform to study the advantages of on-line Bayesian OED *in vivo* in Section 7.2.2 (p. 217) of the following Chapter 7.

Chapter 7

Bayesian Optimal Experimental Design for Parameter Inference: A Computational and Experimental Study

Parameter inference is a crucial step in model development, yet, inadequate experimental data can hinder it or even render the task impractical. Optimal experimental design poses an enticing solution to generate highly informative data to identify model parameters while minimising the time and resources used by experimentalists. In this chapter, I will expand on the research regarding prediction uncertainty presented in Chapter 4 (see technical details in Section 3.2.5, p. 84 of Chapter 3) to propose a Bayesian OED strategy (see Section 3.2.7.3, p. 92 of Chapter 3 for the methodology details). This Bayesian OED approach aims to find the experiments that maximise the prediction uncertainty of models. Hence, I seek to find the data that our models know less about, potentially the one we can extract more information from. In the first main section of this chapter, I will introduce a computational (i.e. *in silico*) study for this strategy, starting with a simple example as a proof of concept (Section 7.1.1, p. 186). Then, I will build upon these results to successfully expand the analysis using a model of higher complexity (Section 7.1.2, p. 191). Next, I will further analyse the potentially detrimental effects of restrictive experimental constraints on the OED strategy (Section 7.1.3, p. 198).

Thanks to the promising results I obtained, I decided to translate my Bayesian OED work into the context that it is supposed to aid. In the second main section

of this chapter, I will leverage my findings to implement the methodology in a real scenario (i.e. *in vivo*), thanks to the platforms presented in Chapter 6. Therefore, I used fluorescence microscopy and microfluidics to generate data for a synthetic system, the genetic toggle switch developed in *E. coli*. First, I will build up on the computational study done in Section 7.1.2 (p. 191) and draw a parallel experimental one comparing random and optimal dynamic input designs (Section 7.2.1, p. 207). Then, I will expand the work by adding on-line OED to the analysis (Section 7.2.2, p. 217) and see if this presents an advantage to parameter inference or to avoid automatically sub-optimal initial designs generated due to experimental constraints. I will finish the chapter by analysing the stability properties of the system under study (similarly as done in Section 5.2, p. 151 of Chapter 5) and introduce this information as a type of constraint when performing Bayesian inference (Section 7.2.3, p. 225).

7.1 *In silico* Bayesian Optimal Experimental Design for Parameter Inference Exploiting Prediction Uncertainty

Bayesian optimal experimental design (OED) is a promising, yet challenging, strategy to design highly informative experiments directed to parameter inference. In this section, I will show that since parameter uncertainties propagate to model predictions, I can exploit this relationship combined with optimisation algorithms to design *a priori* maximally informative experiments for the task. First, I will test the Bayesian OED strategy using a simple case study. I will use the negative feedback loop system described in Section 2.3 (p. 21) of Chapter 2, fixing five of the nine parameters to reduce the complexity of the problem. With a set mathematical model and four parameters to infer, I will show how a sequential OED strategy outperforms intuition-driven experiments. Next, I will increase the complexity of the validations using a mathematical model for the genetic toggle switch (M_3). I will show how, in the absence of restrictive constraints, my Bayesian OED strategy designs highly informative experiments, outperforming a dynamic random (i.e. multiple time-varying concentrations of external inducers within one experiment are randomly selected) experimental design. To ensure the validity of the *in silico* results, I used two sets of parameters to generate the experimental pseudo-data. I will show how, in both instances, the experiments designed by my strategy generate less uncertain parameter posteriors closer to the values set to simulate the pseudo-data. Finally, I will assess a more restrictive experimental design scenario with a higher number of constraints. The results show that while, due to these constraints, my strategy generates low informative designs (steady-state exploration), an on-line update of model parameters and the experimental design (on-line OED) overcomes the issues generating maximally informative data.

7.1.1 Bayesian Optimal Experimental Design for Parameter Inference: Proof of Concept

Due to the potential use of prediction uncertainty to quantify *a priori* the informative content of experiments seen in Section 4.3 (p. 131) of Chapter 4, I coupled the methodology to an optimiser (Bayesian Optimisation) to design input

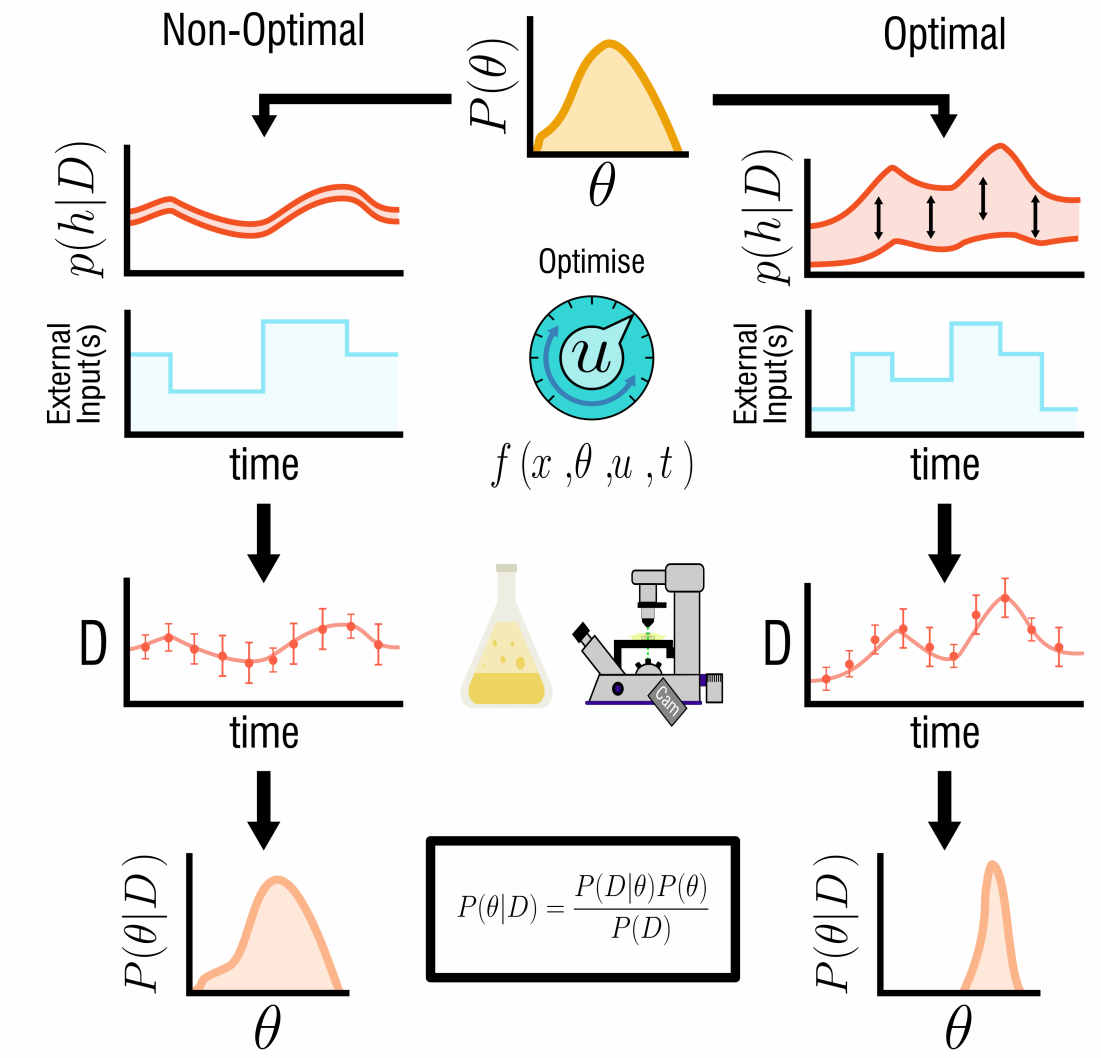


Figure 7.1: Schematic of the proposed strategy for the design of optimal experimental profiles for parameter inference. The optimiser explores the design space for the considered external inputs to maximise the prediction uncertainty ($p(h|D)$) of the model (top panels), starting from the parameter prior knowledge $P(\theta)$. Afterwards, experimental data following the designed inputs is generated (middle panels) and used for Bayesian inference (bottom panels), improving parameter identification compared to non-optimal experiments. The strategy seeks an experimental scheme that the current parameter knowledge predicts poorly.

profiles that maximise the informative content of experiments for parameter inference. The OED strategy explores the design input space for u to find the stimulus that maximises prediction uncertainty with respect to the current parameter knowledge (see technical details in Section 3.2.7, p. 87 of Chapter 3). According to the relationship between experimental design, parameter knowledge and prediction uncertainty [79], the optimal experiment will potentially better estimate parameters, minimising their uncertainty compared to non-optimal experiments with high prediction certainty after inference. Fig. 7.1 shows a schematic for this proposed methodology to optimise experiments for model inference.

First, as a proof of concept, I used the inducible negative feedback loop toy model (equations 2.14, p. 32 from Section 2.3 (p. 21) of Chapter 2, represented in Fig. 7.2a), to assess the Bayesian OED strategy potential. To simplify the problem, I fixed all model parameters except n_1, K_{M2}, d_2 and k_m, reducing the inference problem to these four parameters. I defined the parameter priors as $\mathcal{N}(3,1)$ (truncated at bounds 1 and 5) for n_1, $\mathcal{N}(15.5,7.25)$ (truncated at bounds 1 and 30) for K_{M2}, $\mathcal{N}(0.505,0.2475)$ (truncated at bounds 0.01 and 1) for d_2 and $\mathcal{N}(3,1)$ (truncated at bounds 1 and 5) for k_m.

I then defined two sets of pseudo-experiments with different starting conditions for the state variables (mRNA = 10 A.U. and P = 100 A.U. for experiment 1; mRNA = 500 A.U. and P = 1500 A.U. for experiment 2) generated with the nominal parameter values ($n_1 = 3$, $K_{M2} = 10$, $d_2 = 0.8$ and $k_m = 3$). The profile patterns for the external inducer U followed an intuition-driven design, changing every 20 units of time (maximum time was set to 60 A.U.). The input profiles then followed [0,100,0] A.U. for experiment 1 and [100,0,100] for experiment 2. I used 10 % heteroscedastic additive Gaussian noise (a low error as observed in our microfluidic experiments, see Section 3.2.1.4, p. 73 of Chapter 3 for more details) to generate the different pseudo-data traces. A multi-experimental inference resulted in good identification of all the parameters except for K_{M2}, with close to null improvement compared to the defined prior as shown in Fig. 7.2b. The gain in information ($H(\theta) - H(\theta|D)$ as defined in Section 3.2.3, p. 81 of Chapter 3 and extensively used in Chapter 4) was 9.79.

To assess the Bayesian OED strategy, I optimally designed the two experiments using the same state initial conditions and time switches for the inducer U. I

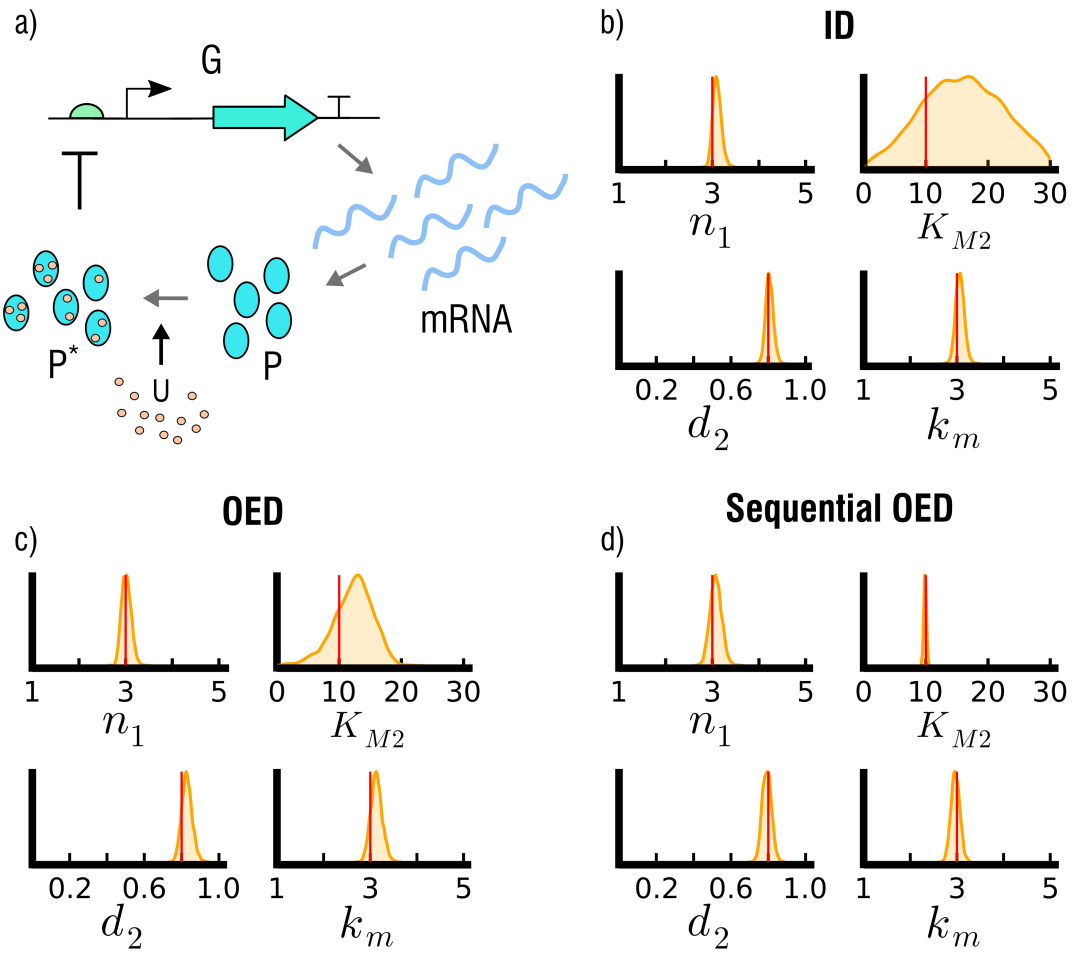


Figure 7.2: Bayesian optimal experimental design prof of concept. a) Schematic of the inducible negative feedback loop toy model defined by equations 2.14. Posterior marginal posteriors for inferences using two intuition-driven (ID) experiments (b), two optimally designed experiments (c) and two sequential optimally designed ones (d). The parameter values arbitrarily fixed for all the simulations are: $\alpha = 1$, $V_m = 200$, $K_M = 20$, $d_1 = 0.5$, $n_2 = 2$.

arbitrarily set the maximum inducer U to 100 A.U. for the optimisation scheme. Results show similar inference results for n_1, d_2 and k_m and a better identification for K_{M2} when using optimal experiments rather than intuition-driven ones as shown in Fig. 7.2c where the parameter presents a lower uncertainty compared to the previous case. The gain in information was 10.35, 1.06 times higher than the obtained with intuition-driven experiments. Finally, I tested a sequential OED scheme to evaluate a better strategy for experimental design. I inferred the model using the first optimal experiment defined from the previous test (mRNA = 10 A.U. and P = 100 A.U.) and used the updated model to design the second optimal experiment. I then inferred the model using as parameter priors the posteriors from the first inference. The gain in information was 12.30, 1.26 times higher than the obtained with intuition-driven experiments. As represented in Fig. 7.2d, in this situation, all the parameters got well identified, including K_{M2}.

In both OED strategies, the increase of gain in information seemed small due to considering the multivariate posterior. Three of the four parameters (n_1, d_2 and k_m) were highly and similarly identified in the three scenarios. Yet, the remaining parameter (K_{M2}) was poorly identified in the intuition-driven case, with an information gain 32 times higher when using OED and 157 times higher when using OED with sequential designs. Since, in this example, most of the gain in information was focused on one parameter, this gets masked when considering it using the full multivariate posteriors. Hence, these results indicate the value of Bayesian OED and sequential designs to maximise the informative content of experiments.

The supplementary Fig. S23 shows the parameter priors used for the analysis (a) and the five different pseudo-experiments used compared to the model prior predictions (b-f) so that readers can compare these to all the inferred posteriors and the model predictions when using them. The supplementary Fig. S24 shows the five different pseudo-experiments used compared to the model predictions using the parameter posterior obtained from intuition-driven experiments. These indicate a low uncertainty and high accuracy, except for the last sequential optimal experiment. The supplementary Fig. S25 shows the five different pseudo-experiments used compared to the model predictions using the parameter posterior obtained from optimal experiments. The plots highlight a general improvement for all predictions, pointing to the advantage of OED. The supplementary Fig. S26 shows the five different pseudo-experiments used

compared to the model predictions using the parameter posterior obtained from sequential optimal experiments. The results indicate a high accuracy and low uncertainty for all the evaluated cases, highlighting the advantage of the Bayesian OED strategy to improve the informative content of experiments concerning current parameter knowledge.

7.1.2 *In silico* Validation of Bayesian Optimal Experimental Design for Parameter Inference Using the Genetic Toggle Switch

I used a model for the genetic toggle switch (M_3, described in Section 3.2.1.1, p. 64 of Chapter 3) to further validate the OED methodology in an *in silico* study. I defined two sets of parameter vectors, Θ_{TS}^{T1} and Θ_{TS}^{T2} (Table 3.8, p. 68 of Chapter 3), and used them to generate independent sets of pseudo-data traces for the observables of the system (RFP-LacI, GFP-TetR) using 10 % heteroscedastic additive Gaussian noise (a low error as observed in our microfluidic experiments for the genetic toggle switch, see Section 3.2.1.4, p. 73 of Chapter 3 for more details) and a regular sampling time of 5 minutes. First, I generated one 24 hour pseudo-experiment for each parameter vector and let the concentration of aTc and IPTG change every three hours to randomly selected values between the allowed bounds (0-100 ng/mL and 0-1 mM, respectively). I defined this random experiment for an initial update (Iteration 1) of the highly uninformative priors ($Prior_b$ from Table 3.7, p. 67 of Chapter 3) as suggested by my prediction uncertainty analysis preliminary results (see Fig. 4.8e of Chapter 4). Then, I generated two more random experiments using the same experimental constraints as for the initial pseudo-experiment for each parameter vector. I used the pseudo-data to sequentially update the model parameters (Iterations 2 and 3) and observe the convergence to the parameter values used to generate the pseudo-data.

I then used my OED strategy to design two optimal experiments applying the same experimental constraints as for the initial pseudo-experiment, for each parameter vector, to sequentially update the model parameters (Iterations 2 and 3). I designed the first optimal experiment using samples from the posterior obtained after inference using the initial random experiment. Consequently, I used the samples from the posterior inferred (technical details of the process in

Section 3.2.2.2, p. 77 of Chapter 3) using the initial random experiment and the first optimal pseudo-experiment to design the second OED experiment. The supplementary Figs. S27 and S28 show the traces for the pseudo-experiments used in this section (initial experiments, randomly designed experiments and optimally designed experiments) generated using the parameter vectors Θ_{TS}^{T1} and Θ_{TS}^{T2}, respectively, and the inducers input profiles.

The *in silico* analysis showed that my Bayesian OED methodology resulted in experiments that reduced prior uncertainties while better identifying the parameters used to generate the pseudo-data. After inference on an initial data set (Iteration 1), I sequentially inferred the model on two random (Rand) or optimal (OED) experiments and assessed the posteriors using the same methods employed throughout Chapter 4. First, I used the posterior sample distance (d_{PSD}, see Section 3.2.2.3, p. 79 of Chapter 3) to the parameter vector used to generate the pseudo-data to quantify the inferred posterior improvement when using the two types of experimental designs. Fig. 7.3 shows the violin plots for the d_{PSD} assesses using each sample of each parameter posterior obtained for each experiment.

When using Θ_{TS}^{T1} to generate the pseudo-data (d_{PSD} results represented in Fig. 7.3a), I observed a better and faster convergence of parameter posteriors to the values used to simulate the pseudo-data while achieving lower uncertainties. Parameter k_{IPTG}, θ_{TetR} and θ_{aTc} were the most challenging to identify on both experimental design strategies (i.e. Rand and OED). Nonetheless, iterations using optimal experiments managed to shrink the parameter posteriors while centring them closer to the parameter values used to generate the pseudo-data. Inferences using random experiments failed in achieving both objectives with the amount of data provided. While for the remaining parameters, posteriors got centred around the values from Θ_{TS}^{T1} in all experimental designs. Nonetheless, with the same amount of data, OED achieved lower parametric uncertainties reflected in a final d_{PSDM} score ($d_{PSDM} = 0.05017$), almost one half smaller when compared to the one obtained using random experiments ($d_{PSDM} = 0.09623$).

When using Θ_{TS}^{T2} to simulate the pseudo-data (d_{PSD} results represented in Fig. 7.3b), both experimental design approaches reached posterior convergence to the parameter vector used to generate the pseudo-data, with optimal experiments reaching a higher reduction on parametric uncertainties. I noted that the results

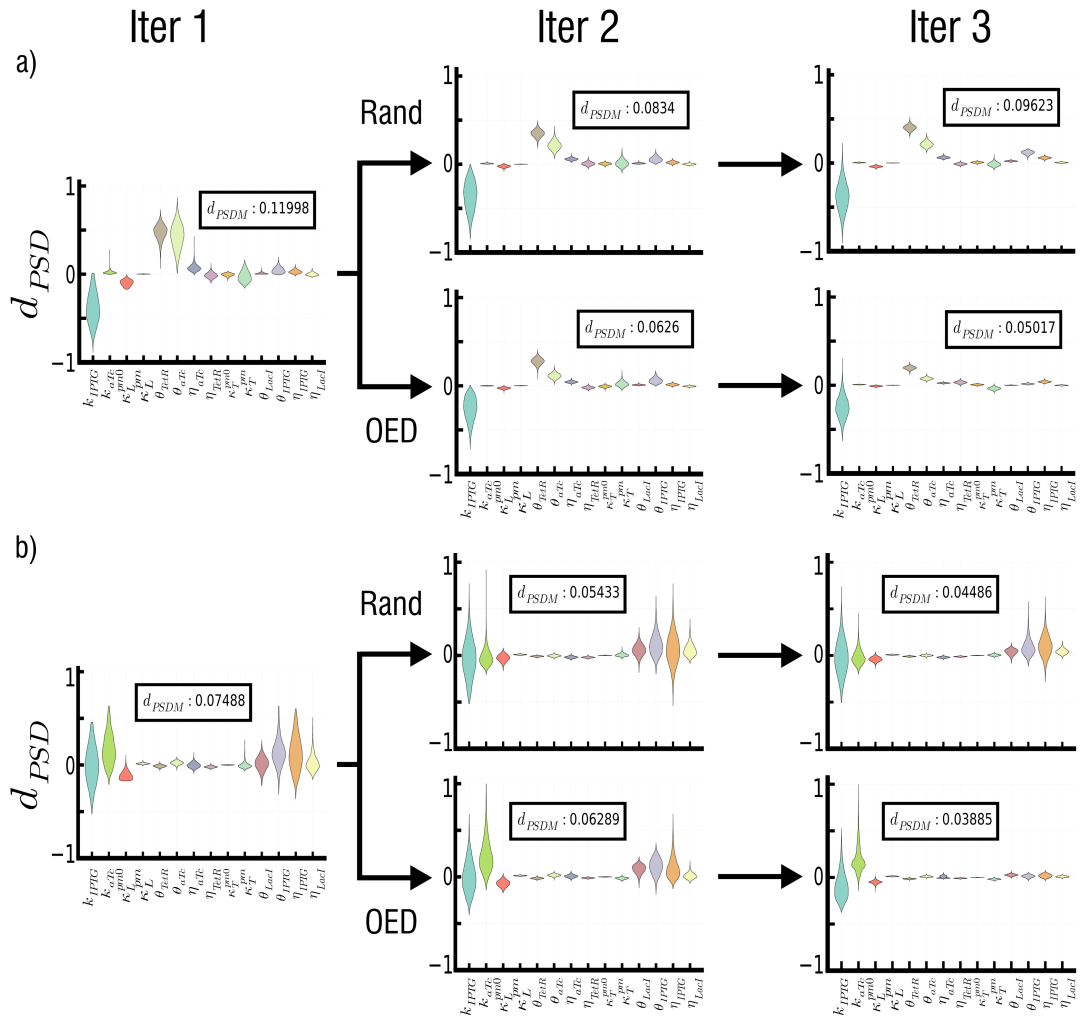


Figure 7.3: Bayesian optimal experimental design improves convergence to the true parameters. Posterior sample distance (d_{PSD}) to the parameter vector Θ_{TS}^{T1} (a) and Θ_{TS}^{T2} (b) used to generate the pseudo-data (Table 3.8, p. 68 of Chapter 3) for toggle switch model M_3, after inference on a random pseudo-experiment (Iter 1), and two consecutive rounds of inference (Iter 2 and Iter 3) using random (Rand) or optimally designed experiments (OED). All pseudo-data assumes a maximum experimental time of 24 hours, a sampling time of 5 minutes, and changes on the concentration for the two inducers every 3 hours with a 10 % heteroscedastic noise level for RFP and GFP.

of the second inference (Iteration 2) are similar between both experimental approaches, making the random design better in terms of d_{PSDM}. However, for the third inference iteration, optimal experiments reduced posterior uncertainties for θ_{LacI}, θ_{IPTG}, η_{IPTG} and η_{LacI}, with a $d_{PSDM} \sim 5$ times smaller on average for these parameters when compared to the inference results for the last random experiment. The results again highlight the better d_{PSDM} score for optimal experiments ($d_{PSDM} = 0.03885$) compared to random experiments ($d_{PSDM} = 0.04486$), and hence, the advantage of using OED for parameter inference.

The Bayesian OED strategy improved parameter identification in computational studies compared to a random design of experiments, yielding an average 0.1 additional fold-change increase in information gain. I quantified the information gain of an experiment as the difference between prior and posterior entropies ($H(\Theta) - H(\Theta|D)$) as defined in Section 4.2 (p. 124) of Chapter 4 (represented in Fig. 7.4a and Fig. 7.4b). I additionally computed, for comparison reasons, the variance/covariance decrease ($-log(|\Sigma|)$) when assuming posterior normality as a faster metric to evaluate (represented in Fig. 7.4c and Fig. 7.4d). While d_{PSDM} also considers the posterior sample distance to the parameters used to simulate the pseudo-data, these metrics become more relevant in *in vivo* studies where the true parameter values are unknown. With the gain in information and variance/covariance decrease, I only quantified the posterior uncertainty reduction after each considered inference.

When using Θ_{TS}^{T1} to generate the pseudo-data, the gain in information (Fig. 7.4a) and variance/covariance decrease (Fig. 7.4c) were better for OED experiments than random ones. Each iteration considering optimal experiments reached a higher fold increase in both metrics when compared to random experiments. Additionally, the first optimal experiment (Iteration 2) attained a better value for both metrics than all random experiments together (Iteration 3). The results are in concordance with the ones obtained using d_{PSDM}.

When using Θ_{TS}^{T2} to generate the pseudo-data, the gain in information (Fig. 7.4b) and variance/covariance decrease (Fig. 7.4d) also reached higher fold increases when using optimal experiments. Particularly, the optimal experiment in Iteration 3 had the highest fold increase in information of all the experiments. The results achieved from the two parallel *in silico* studies highlighted the advantage of Bayesian OED for parameter inference at each step of the process.

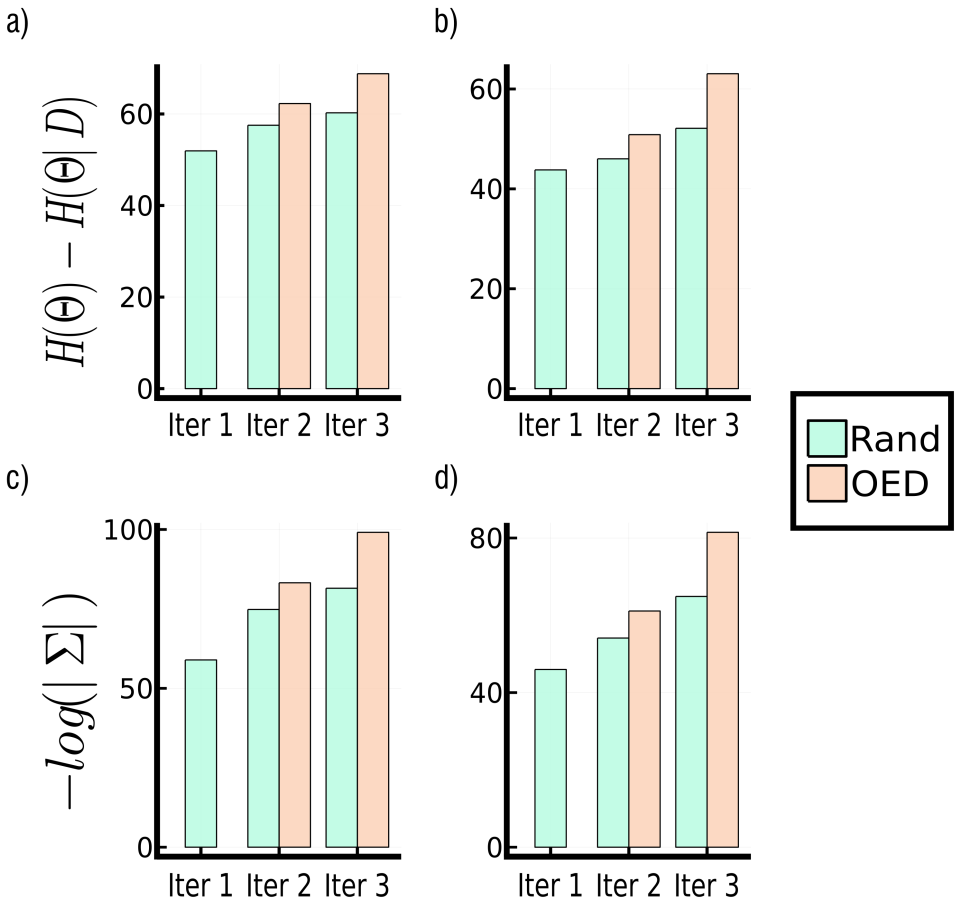


Figure 7.4: Bayesian optimal experimental design reduces parameter uncertainties. Gain in information, measured as $H(\Theta) - H(\Theta|D)$, for the toggle switch model M_3, after inference on a random experiment (Iter 1), and two consecutive rounds of inference (Iter 2 and Iter 3) using random (Rand) or optimally designed experiments (OED) when employing the parameter vectors Θ_{TS}^{T1} (a) and Θ_{TS}^{T2} (b) to generate the pseudo-data. Improvement of a metric summarising parameter variances and covariances, measured as $-log(|\Sigma|)$, when using the parameter vectors Θ_{TS}^{T1} (c) and Θ_{TS}^{T2} (d) to generate the pseudo-data.

Optimal experimental design also accelerated convergence to the true stability properties of the toggle switch system. As a conclusive *in silico* assessment of the advantages of Bayesian OED, I sought to compare the stability properties of the model used to simulate the pseudo-data with the final inferred models. First, I computed the stability matrices (see Section 5.2, p.151 of Chapter 5 for the introduction to these matrices and Section 2.3.7, p. 34 of Chapter 2 for an explanation of bistability) for the toggle switch model using the parameter vectors used to generate the pseudo-data (True), Θ_{TS}^{T1} (Fig. 7.5a) and Θ_{TS}^{T2} (Fig. 7.5b), as ground truths. Then, I computed the stability matrices using the posterior mean for the last inferences (Iteration 3) using random (Rand) and optimal (OED) experiments to compare them to the ground truths. I used as comparison metrics accuracy (correct stability prediction over the total amount of predictions) and IoU (as used in image segmentation studied in Chapter 6) of the overall matrices and each of the three different regions (i.e. bistability and RFP or GFP dominant monostability).

The results showed higher accuracy and IoU for the models inferred on optimal pseudo-data. When I used Θ_{TS}^{T1} to generate the pseudo-data, OED achieved a $\sim 2\%$ and $\sim 3\%$ increase in the overall accuracy and IoU, respectively, compared to random experimental design as represented in Fig. 7.5c. Note that, while the improvement of the metric seems low, the matrices contain a high number of pixels to classify, where each one of this is a critical property of the system under study. Hence, even small metric increases represent a high improvement in model predictions. Yet, when looking at the similarity of each region, the outperforming results of OED became more evident. While the metrics point to similar values for the RFP dominant monostable region, optimal experiments achieved a $\sim 5\%$ similarity increase for the bistable region and a $\sim 25\%$ for the GFP dominant monostable region.

I observed similar tendencies in the results that used Θ_{TS}^{T2} to generate the pseudo-data as represented in Fig. 7.5d. Optimal experiments achieved better accuracy and IoU scores when considering the whole matrices and each independent region. The best improvement in predictions concerned the GFP dominant monostable region. The models inferred on random experiments attained less than 3 % accuracy and 0.03 IoU, while optimal experiments achieved an accuracy of 100 % and an IoU of ~ 0.78.

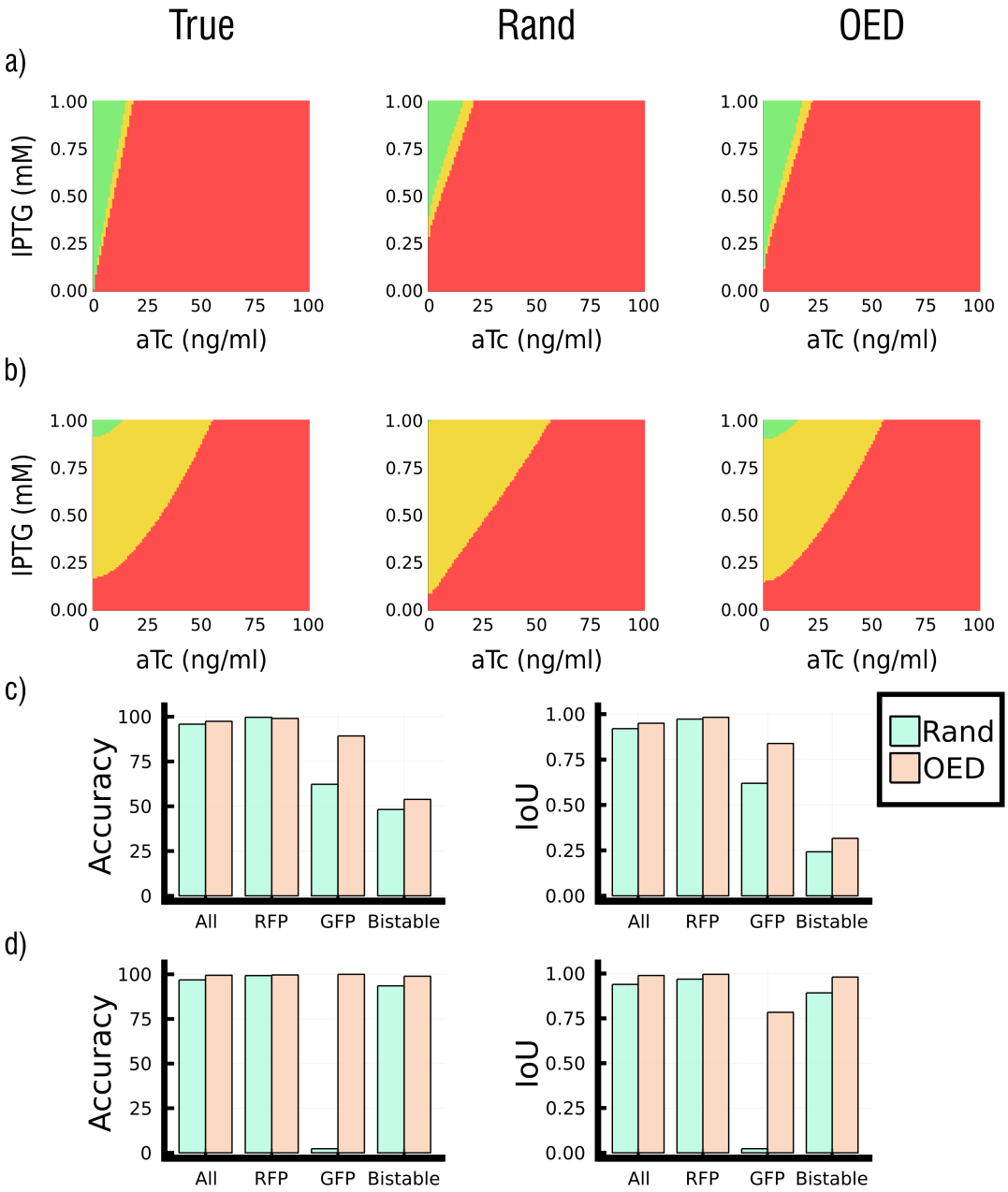


Figure 7.5: Bayesian optimal experimental design aids inference of the stability properties of the genetic toggle switch. Heat maps of the model M_3 stability matrices generated using Θ_{TS}^{T1} (a, left), and Θ_{TS}^{T2} (b, left), the mean posterior parameter values after inference on random experiments (a,b, middle) and the mean posterior parameter values after inference on optimal experiments (a,b, right) as a function of aTc and IPTG. Colour green and red denotes monostability and the dominant fluorescent reporter at steady-state (GFP and RFP, respectively) while yellow colours indicate bistability. Prediction accuracy and IoU when considering the full matrices (All) or the individual regions (RFP, GFP, Bistable) of the results from a) (c), and b) (d).

7.1.3 *In silico* On-line Bayesian Optimal Experimental Design

I used the same model for the genetic toggle switch (M_3, described in Section 3.2.1.1, p. 64 of Chapter 3) to assess the potential of employing the Bayesian OED strategy in an on-line setting (i.e. sequential and automated parameter and input design updates within an experiment). In this case, I only used the parameter vector Θ_{TS}^{T1} (Table 3.8, p. 68 of Chapter 3) to generate the pseudo-data traces for the observables of the system (RFP-LacI, GFP-TetR). Additionally, I implemented the experimental constraints from our microfluidics platform for a more direct comparison with potential experiments (see Section 3.2.11, p. 96 of Chapter 3 for more details). These constraints consider 38-hour long experiments, with constant input steps of three hours where only one of the two inducers is present at each interval at the same concentration for the whole duration of this. Hence, I reduced the experimental design to identify one fixed concentration of aTc and IPTG, and the three-hour interval that each one is present (more details in Section 7.2, p. 206 of this chapter). For additional similarity to *in vivo* experiments, I increased the sampling time to regular intervals of 10 minutes and the pseudo-data heteroscedastic additive Gaussian noise to 20 % (a moderate error as observed in my microfluidic experiments for the genetic toggle switch, see Section 3.2.1.4, p. 73 of Chapter 3 for more details). To reduce the computational time and assess the experimental design strategy with medium informative priors, I used the posterior from the initial experiment (Iteration 1) from the previous section (Section 7.1.2) as my starting prior ($Prior_c$).

I sequentially designed (within the predefined constraints) and inferred four pseudo-experiments using three experimental design strategies: random input design, off-line OED and on-line OED. I followed a sequential inference scheme, where the posterior obtained after inference on one data set became the prior for the next (see Section 3.2.2.2, p. 77 of Chapter 3 for technical details on Bayesian inference). For the random design, I randomly selected a concentration for each inducer between the allowed bounds and the three-hour interval at which each one is present. For off-line OED, I followed a similar approach, optimally computing the concentration of the inducers and the three-hour intervals at which each one appears. Since I followed a sequential inference scheme, each new

experimental design depends on the inference results of the previous iteration. For the fourth experiment, I identified the most uncertain parameter concerning the initially set priors (i.e. smallest $H(\theta) - H(\theta|D)$) and fixed the rest to their maximum likelihood estimate for the OED step. This approach allowed the experimental design to focus on just one highly uncertain parameter. Note that, for this last inference, I still inferred all parameters since I only intended the fixing for experimental design.

I implemented the on-line OED strategy following experimental and computational constraints from our microfluidics setup (see Sections 3.2.11, p. 96 and 3.2.11.6, p. 107 of Chapter 3). At each iteration, I first optimally designed the whole experiment (note that the design for the first iteration is the same as the off-line OED). Then, I divided the experiment into two sub-sections of equal duration (19 hours each) and allocated five hours for all computations (Bayesian inference and OED). Hence, I used the data from the first sub-experiment while neglecting the last five hours to infer the model. In a real scenario, the data for these five hours would be acquired and processed while computations proceed. I considered this inference an intermediate one, starting with an initial MLE to initialise the MCMC chains at a good point in parameter space, reducing the number of warm-up samples (250) and inferring only the required post-warm-up ones for OED (400). Next, I optimally designed the second sub-experiment, generated the pseudo-data and inferred the model as performed in the previous section, with a total of 8000 post-warm-up samples. I then used the posterior distribution to optimally design the next experiment, following the sequential inference scheme. As with the off-line OED strategy, I designed the last experiment by fixing all parameters to their maximum likelihood estimate except the most uncertain one concerning the starting prior. The supplementary Fig. S29 shows the pseudo-data traces for all the random and optimal (off-line and on-line) experiments used in this subsection and the inducers input profiles.

While the new experimental constraints had a detrimental effect on off-line OED, on-line OED overcame the issues generating highly informative data as seen in Fig. 7.6. For the sake of comparability, all the metrics used are subtracted to the same one computed using the prior $Prior_c$ ($H(\theta) - H(\theta|D)$, $log(|\Sigma_{p(\theta)}|) - log(|\Sigma_{p(\theta|D)}|)$ and $d_{PSDM}^{p(\theta)} - d_{PSDM}^{p(\theta|D)}$), where higher values for the three indicate better results (i.e. more information gained). Using the same prior in each iteration allowed me to investigate the cumulative gain in information at

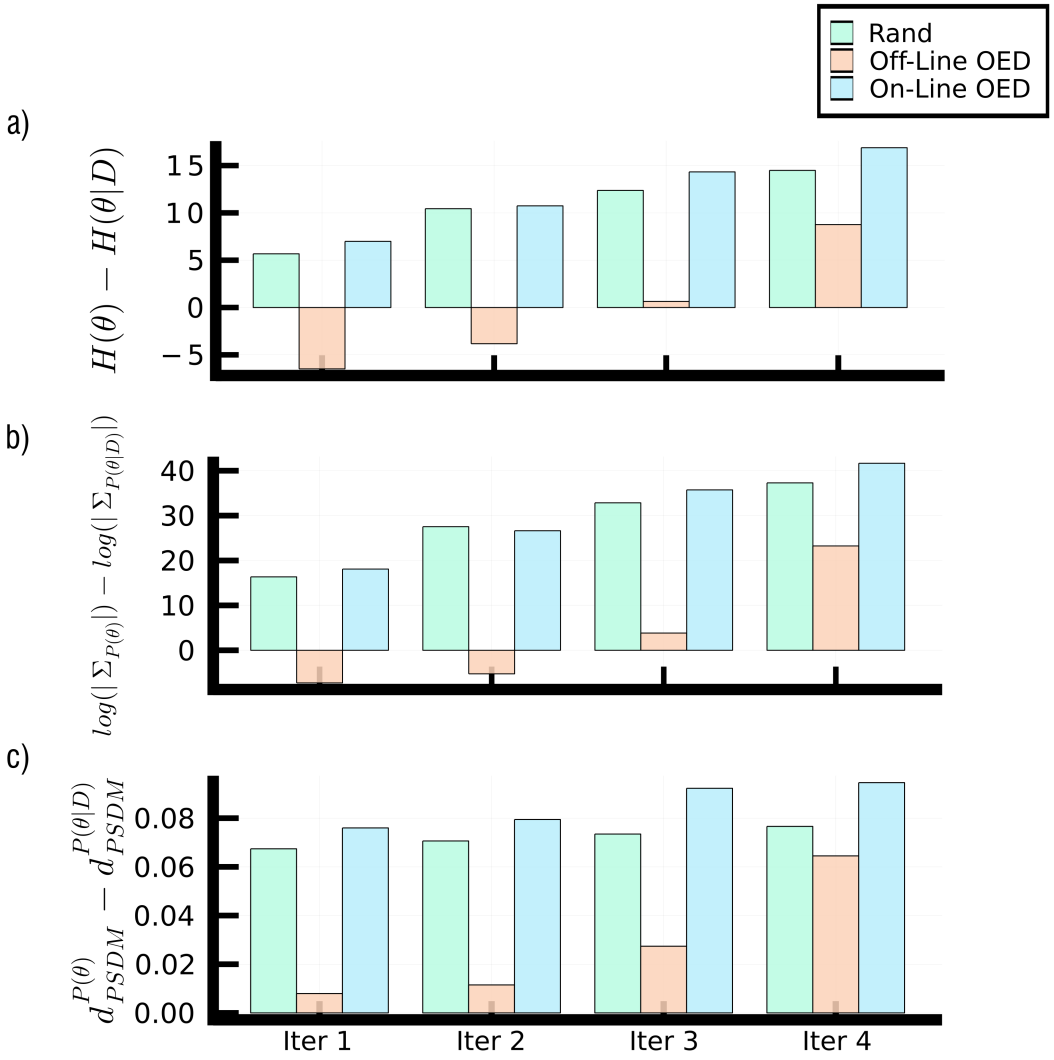


Figure 7.6: On-line OED overcomes heavily constrained experimental setups generating highly informative data. *A posteriori* gain in information measured as the difference between prior ($P(\theta)$) and posterior ($P(\theta|D)$) entropy (a), logarithm of the determinant of covariances (b) and mean posterior sample distance to true value (c) for the toggle switch model M_3. Inferences employed four sequential experiments following random, off-line optimal and on-line optimal experimental designs and using the parameter vector Θ^{T1}_{TS} to generate the pseudo-data.

each sequence step. When applying the difference between prior and posterior entropies (seen in Fig. 7.6a), on-line OED outperformed the other two design approaches. Yet, off-line OED attained the worst results, with an apparent increase in parameter uncertainty after inference for iterations 1 and 2.

Since the first half of the off-line and on-line OEDs was the same for iteration 1, I further analysed the results to understand the increase in information gain achieved by an on-line approach. For the first two experiments, off-line OEDs resulted in constant input profiles. These two experiments applied a sustained IPTG concentration (0.23 and 0.16 mM, respectively) with no switch to aTc at any given time. However, such input design drove the model to two opposite steady-states simultaneously for different prior samples. Fig. 7.7 represents this phenomenon in a single step experiment, where for a fraction of the posterior samples, model predictions indicate high levels of RFP and low levels of GFP, and the opposite for other ones. Such dichotomy within samples of a same posterior could be acceptable for stochastic models when looking at a cellular resolution. Yet, we are dealing with deterministic models where the population should be at one same steady-state. Model simulations predicted the maximum and minimum fluorescence concentration for both reporters (RFP and GFP) using samples from the same parameter distributions. Consequently, due to the experimental constraints and the bi-stable nature of the toggle switch system, my Bayesian OED strategy identified steady-state experiments without dynamics as the most informative ones. While these experiments provided important information about the system, the absence of any dynamics made them poorly informative compared to the other two design strategies. On-line OED overcame this experimental constraint issue, using the first half of the experiment to inform the model about the steady-state and the second half to apply dynamics to the system. Such an approach made on-line OED the best experimental design strategy at each step of the process according to the metric.

Assuming posterior normality rendered similar results when assessing the cumulative informative content of experiments. To evaluate the gain in information in such conditions, I used the difference between prior and posterior uncertainty computed as the logarithm of the determinant of the covariance matrix (seen in Fig. 7.6b). The results point again to on-line OED as the best design strategy. However, the random experimental design had a higher gain in information in iteration 2. In the same manner, off-line OED attained a lower

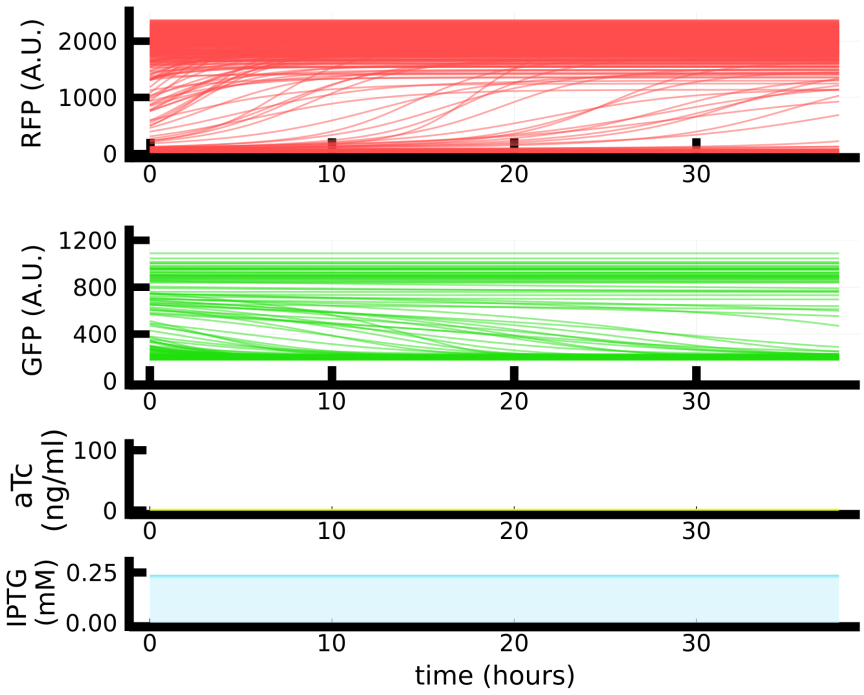


Figure 7.7: Constant inputs can drive a model for the genetic toggle switch to predict two steady-states simultaneously. Prediction of the first off-line OED experiment from the *in silico* validation of the Bayesian on-line OED strategy. For the same prior distribution, some samples indicate an RFP dominant steady-state, while others point to a GFP dominant one.

uncertainty increase than when estimating entropies. Such effects can be explained by the average lack of normality of the inferred posteriors, highlighting the advantage of probabilistic methods when assessing parameter uncertainties as seen in Chapter 4. Nonetheless, the results of both metrics achieved a high correlation for each type of experimental design, with r^2 values above 0.98.

The posterior sample distance to the parameter vector used to simulate the pseudo-data shows concordance with the previous two metrics. To assess the cumulative gain in information and posterior distance to Θ_{TS}^{T1}, I computed the difference between prior and posterior d_{PSDM} (technical details in Section 3.2.2.3, p. 79 of Chapter 3), shown in Fig. 7.6c. The results indicate that the posteriors inferred from on-line OED get closer to Θ_{TS}^{T1} at each iteration on top of obtaining lower parametric uncertainties. As expected from the previous results, off-line OED attains the worst scores. However, the metric increase at each step of the process indicates that the experimental designs achieve posteriors closer to Θ_{TS}^{T1}.

While such posteriors increased in parameter uncertainties in iterations 1 and 2, this is still getting closer to the parameter vector used to simulate the pseudo-data. Nonetheless, the metric shows a high correlation with the previous two, with the lowest r^2 value of 0.93. The results emphasise the advantages of on-line OED to overcome extensive experimental constraints and generate highly informative data resulting in better parameter identification and uncertainty reduction.

The supplementary Fig. S30 shows the d_{PSD} metric to Θ_{TS}^{T1} for each parameter and each posterior inferred in this section. These results show that both random and on-line optimal experiments achieve a high uncertainty reduction along the process. However, the latter attains better centralisation to the parameter values used to generate the pseudo-data (i.e. $d_{PSDM} = 0$). The off-line optimal experiments resulted in a slower convergence, particularly for the two sub-optimal non-dynamic experiments (iterations 1 and 2). The supplementary Fig. S31a shows the heat maps of the stability matrices generated using Θ_{TS}^{T1} and the posterior mean from the last inferences (iteration 4) for each experimental design considered. As expected by the posteriors analysis, the on-line OED strategy obtained a closer stability matrix than the one generated using Θ_{TS}^{T1}, followed by random and off-line optimal experimental designs. The supplementary Fig. S31b shows the accuracy c comparisons between the matrix generated using Θ_{TS}^{T1} and the ones generated using the posterior means. The supplementary Fig. S31c shows the same comparison but uses the IoU as a metric. These results quantify the observations extracted from panel b regarding the overall matrices and each stability region obtained.

7.1.4 Conclusions

Optimal experimental design makes the arduous task of inferring model parameters from scarce and noisy data more efficient and cost-effective. Yet, while Bayesian statistics are becoming more popular to infer model parameters, Bayesian OED still presents a substantial challenge due to its high computational cost, even in easy scenarios. In this section, I showed *in silico* that we could use model prediction uncertainty (as shown in Chapter 4) to design highly informative experiments, establishing an efficient strategy for Bayesian OED. First, as a proof of concept, I used a simple example, an inducible negative feedback loop system with only four free parameters, to illustrate the strategy

potential. The results show that Bayesian OED generated data capable of accurately inferring the model parameters outperforming an intuition-driven strategy. Furthermore, a sequential approach to OED showed an increased efficiency, highly decreasing parameter posterior uncertainties.

Next, I used a mathematical model for the genetic toggle switch (M_3) to assess the strategy performance in a more complex scenario. I set up a lowly constrained hypothetical experimental setup and compared a random experimental design to Bayesian OED. I defined two parameter sets to generate the pseudo-data. Then, I assessed how well inferences using the two experimental design types (random or optimal) performed at identifying the predefined parameter values and the amount of uncertainty associated with the inferred posteriors. In both cases, Bayesian OED outperformed the random experimental design strategy highlighting its potential. Thanks to the promising results obtained, I included a generalised version of the Bayesian OED strategy in my Julia package, BOMBs.jl, described in Chapter 8.

Finally, I tested the strategy in a more realistic yet highly constrained experimental scenario mimicking our microfluidics setup. In this case, Bayesian OED performed poorly, resulting in experimental designs without input dynamics due to the setup constraints. The reason for such results was that, for some IPTG concentrations, the model predicted both steady-states for the two fluorescent reporters simultaneously. This prediction resulted in the most uncertain one, but the absence of system dynamics made the inferred posteriors highly uncertain. To overcome the issue, I introduced an on-line Bayesian OED strategy. The first half of the designed experiments could still be a constant input profile without dynamics. But, since I updated the model parameters halfway, the second half of the experiment could explore input dynamics. These on-line optimal experiments were the most informative of all the tested ones. Hence, I showed that Bayesian OED generates highly informative experimental data, easing the parameter inference task. However, highly constrained experimental setups or unknown model behaviours might produce sub-optimal designs. For this reason, I deem it helpful to perform a preliminary investigation (e.g. by simulating the system multiple times at multiple experimental conditions and performing parameter estimation rounds with pseudo-data) and analysis (e.g. practical identifiability and sensitivity) of the potential optimal experiments. This initial exploration allows the researcher to assess the need for strategy

modifications or the introduction of on-line OED for a more automated approach.

7.2 *In vivo* Implementation of Bayesian Optimal Experimental Design for Parameter Inference Exploiting Prediction Uncertainty

OED is a powerful tool to reduce the time and resources allocated to experiments directed at parameter inference while improving (or at least maintaining) the quality of the results. Yet, the experimental design strategy has seen limited applicability in biological sciences, restricted to computational studies for the most part. In this section, I will follow up on the computational studies done in the previous section and use Bayesian OED to infer the parameters of a model for the genetic toggle switch using microfluidics experimental data. First, I will infer sequentially the same model using experiments that followed a random or optimal experimental design. Due to the effects of the experimental constraints, I assessed the optimisation results and complemented them with a model-based experimental design to significantly improve the results. Then I will determine the *a posteriori* informative content of the experiments using an information theoretic approach. Since model predictions are more important than reducing parameter uncertainty, I will then assess them using a validation set of randomly designed microfluidic experiments. I also implemented an on-line Bayesian OED platform within our microfluidics setup. I will investigate the informative content of an experiment of the sort and the predictive capabilities of the inferred model. The promising results highlight the potential of an automatic and periodic update of model parameters and experimental designs to improve inference results, particularly in highly constrained experimental setups. Finally, I will examine the stability properties of the genetic toggle switch under diverse concentrations for the two external inducers. I will also illustrate how to introduce such conditions as inference constraints and evaluate the prediction uncertainty strategy to optimally select the best ones.

7.2.1 *In Vivo* Validation of Bayesian Optimal Experimental Design for Parameter Inference Using the Genetic Toggle Switch

I applied the Bayesian OED strategy tested in the previous section (see Section 3.2.7.3, p. 92 of Chapter 3) in a real scenario using microfluidics time-series to infer the parameters for the model M_3 (described in Section 3.2.1.1, p. 64 of Chapter 3) describing the genetic toggle switch system. I processed all the microscopy data generated in this chapter as described in Section 3.2.11.5 (p. 105) of Chapter 3 and using the platform described in Section 6.1 (p. 158) of Chapter 6. I set the maximum experimental duration to 38 hours, where the concentration of the inducers was constant in varying intervals of three hours. This setting allowed for a maximum of 13 changes for the two inducers during an experiment (with the final step being restricted to two hours). I constrained the first step of the experiment to have the inducer concentrations used for the overnight incubations. To avoid potential aTc adaptation, I only used IPTG at different concentrations between the set bounds for the overnight incubations. Due to experimental constraints, I only allowed the presence of one (or none) of the inducers per step. Hence, the experiments exposed the cells to a predefined concentration of aTc (between 0-100 ng/mL) or IPTG (between 0-1 mM) for each time interval (i.e. step). Sections 3.1.4 (p. 59) and 3.2.8 (p. 93) of Chapter 3 contains the technical information for the genetic toggle switch and cell growth conditions; and Section 3.2.11 (p. 96) of Chapter 3 the necessary information to perform microfluidic experiments for this system.

I generated a set of experiments split into three different iterations following intuition-driven, random or optimal experimental designs, similarly as done in Section 7.1.2 (p. 191) of this chapter. First, I performed an initial intuition-driven experiment for an initial update of the highly uninformative priors ($Prior_b$ from Table 3.7, p. 67 of Chapter 3). This initial experiment followed a periodic switch pattern every three hours between maximum IPTG (1 mM) and aTc (100 ng/mL) concentrations (Fig. 7.8 Iter 1). From this point, the following experiments used either random (Fig. 7.8 Iter 2 and 3, left) or else optimal (Fig. 7.8 Iter 2 and 3, right) input designs. For the random experimental design, I randomly selected a fixed concentration for aTc and IPTG to be used during the experiments and then randomly selected which of the two inducers would be present at each three-hour interval. Optimal designs followed the same

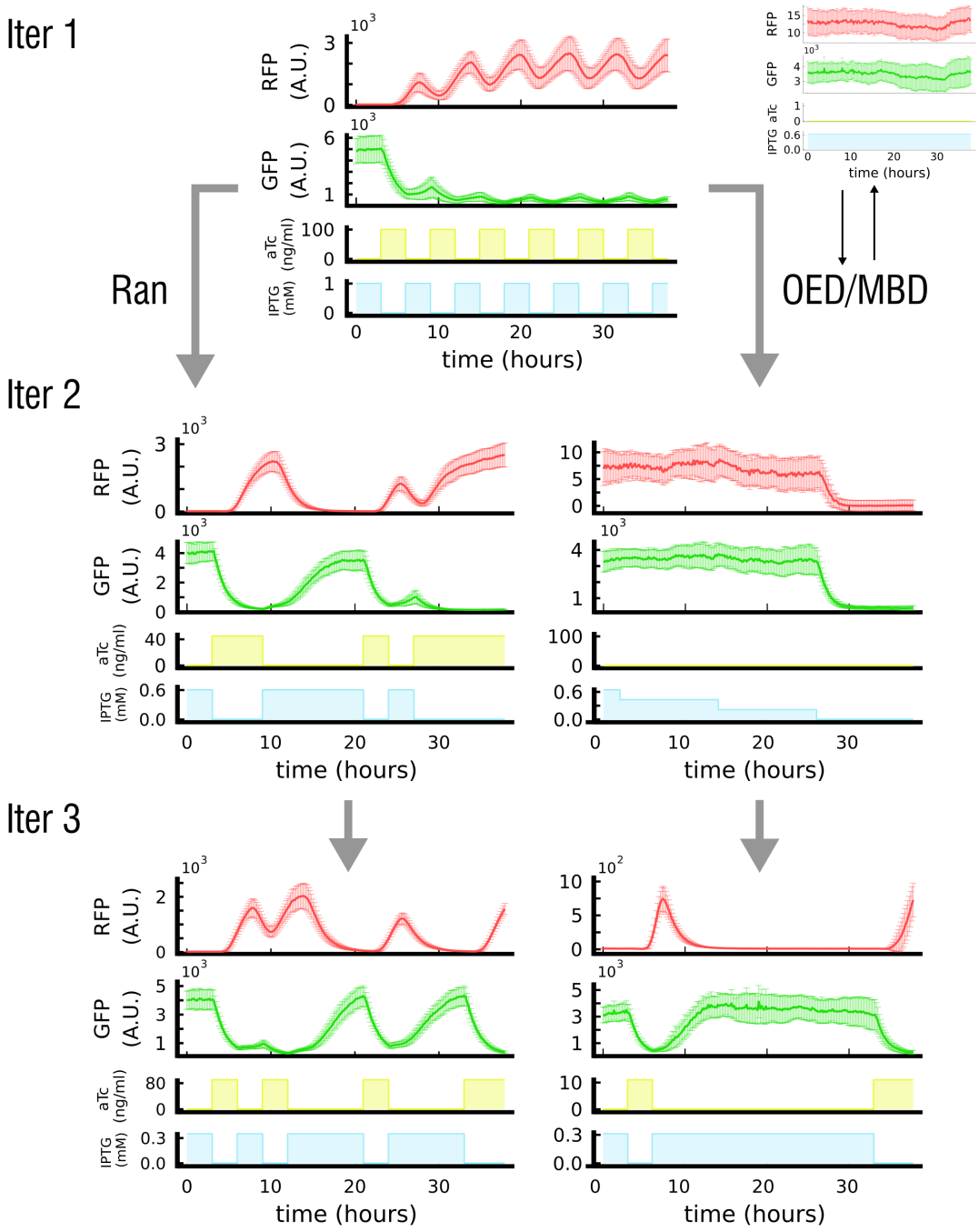


Figure 7.8: Microfluidics experimental data for the genetic toggle switch used for parameter inference. Microfluidics initial experiment (Iter 1) following a pulse input pattern switching between aTc and IPTG at their maximum levels. The following experiments for the next inference rounds (Iter 2 and Iter 3) followed a random input design (Ran, left) or an optimal experimental design (OED, right) combined with a model-based experimental design (MBD) for the experiment in iteration 2 after analysis of the OED results (top-right corner).

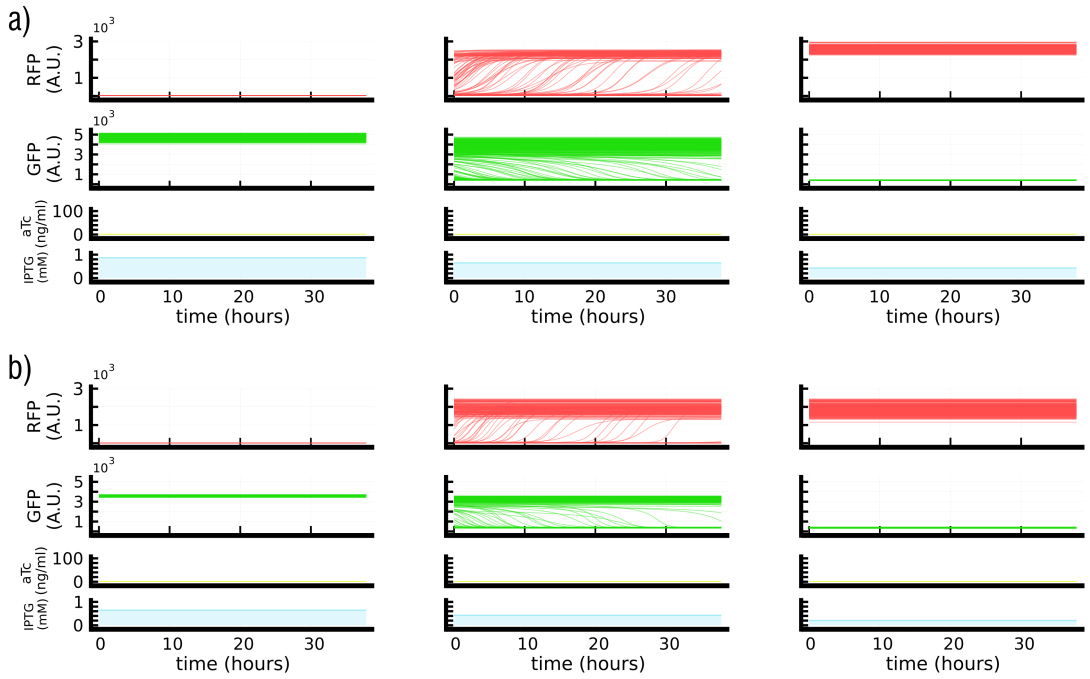


Figure 7.9: Rationale behind the model-based experimental design update of the first OED. a) Model predictions after inference using the initial pulse-like experiment (Iter 1) of a single-step experiment using 0.86 mM IPTG (left), 0.64 mM IPTG (middle) corresponding to the OED results and 0.42 mM IPTG (right). b) Model predictions after inference using the first optimally designed experiment (Iter 2) of a single-step experiment using 0.64 mM IPTG (left), 0.42 mM IPTG (middle) corresponding to the next OED results and 0.20 mM IPTG (right). Following the OED results, aTc was set to 0 ng/mL for all experiments and simulations.

experimental constraints while using my Bayesian OED strategy (Section 3.2.7.3, p. 92 of Chapter 3) to identify the optimum inducer concentrations and switching steps. I employed Bayesian optimisation with a maximum of 4000 utility function evaluations to solve the non-linear optimisation problem. Due to evident overfitting from the initial inference (Iter 1), I performed multi-experimental inferences using the experiments from iterations 1 and 2 and the initial uninformative prior ($Prior_b$ from Table 3.7, p. 67 of Chapter 3). I followed the same procedure to obtain the posteriors of iteration 3. Additionally, I also performed single-experiment inferences using only the experiments from iteration 3 while using the posteriors from iteration 2 as priors.

After analysing the optimisation results from the first OED, I employed a

model-based experimental design to improve the informative content of the design and reduce iterative evaluations of steady-states for the system. The optimal design (Fig. 7.8 top-right corner) was a single-step experiment dividing the model prediction into two different steady-states for parameter samples drawn from the inferred posterior. The experimental design described a constant concentration of 0.64 mM IPTG and 0 ng/ml aTc for 38 hours. While some parameter samples predicted a GFP-TetR dominant state (predicted by all samples at increasing concentrations of IPTG), others predicted an RFP-LacI dominant state (predicted by all samples at decreasing concentrations of IPTG). This apparent transition point between steady-states is represented in Fig. 7.9a, where simulations of the model using samples of the same posterior predict both steady-states for the optimal design (middle) and singular steady-states for increasing (left) or decreasing (right) concentrations of the inducer. Predictions for some parameter samples point to one steady-state but with high delays for the transition between the two. Note that, in this scenario, since I aim to maximise the model prediction uncertainty, a single-step experiment containing the two system steady-states will always present a higher utility function value compared to dynamic experiments where there are transitions between the two. I hypothesised that inferring the model using such an experiment would displace the steady-states transition point to a lower concentration of IPTG as observed in the Bayesian OED *in silico* study (Section 7.1.3, p. 198 of this chapter). This effect would restrict any future optimal designs following the same pattern until I evaluated a range of concentrations for the inducer, including its absence. To validate the hypothesis and back up the decision to combine the OED results with a model-based experimental design, I performed the experiment in our microfluidics platform.

Despite the usual lower informative content of steady-state experiments compared to dynamic ones [86], I performed the experiment, inferred the model and executed OED to design the following experiment. The results of the second round of OED validated my hypothesis by identifying as optimum a single-step experiment with a constant concentration of 0.42 mM IPTG and 0 ng/ml aTc. This experimental design, again, defined the transition point between the two steady-states of the system as represented in Fig. 7.9b, where the phenomenon happens at the newly design experiment (middle) but not when using higher (left) of lower (right) concentrations for the inducer. For this reason, I modified the first OED for a four-step experiment evaluating the steady-state response of

the system at four decreasing equidistant IPTG concentrations (0.64, 0.4288, 0.2112 and 0 mM). I selected as IPTG bounds the value obtained in my first OED and the absence of the inducer (Fig. 7.8 Iter 2, right).

OED combined with a model-based design resulted in experiments with higher informative content measured according to the reduction in posterior uncertainty. After the first initial inference using the pulse-like data set, I identified model overfitting when predicting unseen experiments. For this reason, independently of the experimental design defined in the following iterations, I performed at each step a multi-experimental inference, always using the same starting uninformative priors ($Prior_b$ from Table 3.7, p. 67 of Chapter 3). I approximated the gain in information (Section 4.2.2, p. 127 of Chapter 4) as the difference between prior and posterior entropies and as the minus logarithm of the determinant of the covariance matrix (assumption of normality). For comparability purposes and to determine the overall cumulative gain in information, I used $Prior_b$ from Table 3.7 (p. 67 of Chapter 3) to compute $H(P)$ in all instances. Both metrics (Fig. 7.10a and Fig. 7.10b, respectively) show a higher informative content of optimal experiments when compared to randomly designed ones. The first optimal/model-based designed experiment resulted in a gain in information 1.72 times higher than the one obtained in iteration 1 when estimating entropy differences. The $-log(|\Sigma|)$ was 2.4 times higher when assuming normality of the posteriors. This gain in information achieved by the optimal experiment in iteration 2 was higher than the overall (i.e. posterior obtained in iteration 3) one attained using randomly designed experiments. However, the gain in information from the optimal design in iteration 3 had a lower fold increase. After analysing the model predictions using both posteriors, I identified that the parameter uncertainty shrinkage between iterations 2 and 3 focused on excluding small regions of the parameter space that made model predictions point to two distant steady-states.

Since highly uninformative priors can lead to slower inferences with convergence issues [352], [353], I also inferred the experiments from iteration 3 using the posteriors obtained in the previous iteration as priors. Such sequential inference is a usual approach in Bayesian inference schemes, where, in the light of new unseen data, the current priors are updated to new posteriors. The apparent absence of overfitting from the estimated parameters in iteration 2 allowed faster ($\approx$ 21 times quicker) and more robust inferences. The increase in computational

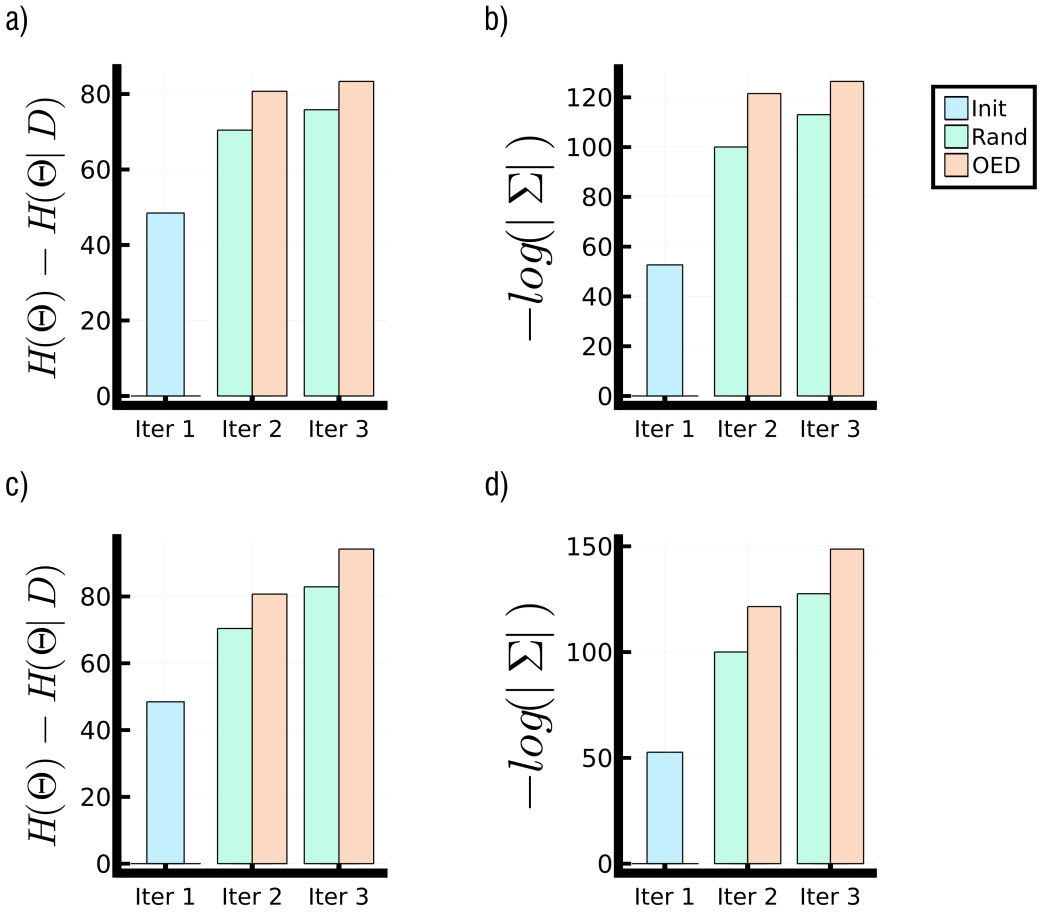


Figure 7.10: Information gain comparison between randomly designed and optimally designed microfluidic experiments. Gain in information, measured as $H(\Theta) - H(\Theta|D)$ and $-log(|\Sigma|)$, of randomly (Rand) and optimally (OED) designed experiments using the uninformative starting priors for each multi-experimental inference (a and b, respectively) or using the posterior from iteration 2 for the single-experiment inference to obtain the posteriors of iteration 3 (c and d, respectively). Blue bars indicate the gain in information after the inference using only the initial pulse-like experiment (Init).

speed (from days to hours) was also partly attributed to the reduction of experimental data used in these single-experiment inferences. Consequently, I observed a higher information gain, measured in both the difference between prior and posterior entropies (Fig. 7.10c) and as the minus logarithm of the covariance matrix (Fig. 7.10d), independently of the design scheme used. Nonetheless, the optimally designed experiments still displayed a higher parameter uncertainty reduction, illustrating the advantage of Bayesian OED when inferring model parameters from noisy experimental data.

Next, I performed three additional validation experiments to assess the predictive abilities of the inferred models toward unseen data. The first validation experiment (Val 1) followed the same random input design implemented for the inference data sets. The second one (Val 2) followed a single-step design to assess RFP-LacI at steady-state with a randomly placed 1-hour IPTG switch. The third experiment (Val 3) followed the usual random input design, switching between media containing aTc and media without any inducer. I predicted the three validation experiments using the final parameter posteriors (Iter 3) inferred from randomly designed (Ran) and optimally designed experiments (OED). I evaluated the results of the two inference schemes used regarding prediction capabilities; employing a multi-experimental inference with uninformative priors (model predictions for a validation data-set seen in Fig. 7.11a) or a single-experiment inference with a previous prior update (model predictions for a validation data-set seen in Fig. 7.11c). I assessed the quality of the results by computing the normalised root mean square error (nRMSE, see Section 3.2.4, p. 83 of Chapter 3 for more details) between experimental data and model predictions for the observables of the system.

I considered predictions of the model using the two parameter posteriors (Ran and OED) obtained from the multi-experimental inference to compare the results. The model inferred from optimally designed experiments (M^a_{OED}) attained a lower nRMSE for the three validation experiments than the one inferred from randomly designed experiments (M^a_{Ran}). First, I computed the nRMSE for each simulation performed from 8000 parameter posterior samples for predictions of RFP-LacI (Fig. 7.11b, left) and GFP-TetR (Fig. 7.11b, middle) independently. The results show lower scores for most prediction samples (> 50 %) of model M^a_{OED} and a higher variability for model M^a_{Ran} when predicting the third validation experiment. Model M^a_{OED} achieved, on average (Fig. 7.11b, right),

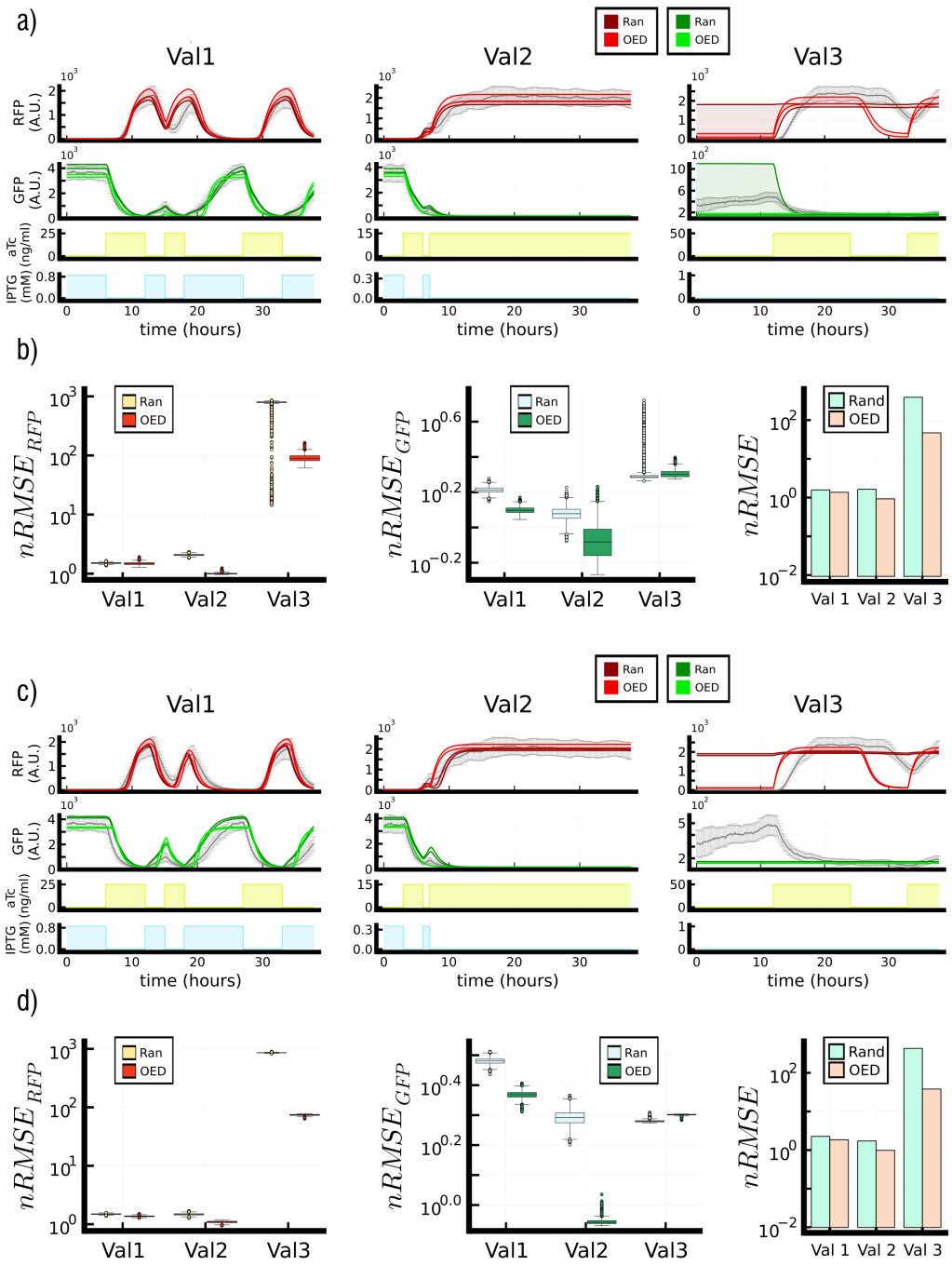


Figure 7.11: Validation of the inferred mathematical model for the genetic toggle switch. Validation experimental data sets and model predictions using the posteriors obtained from the inferences of random (Rand) or optimal (OED) experiments employing uninformative starting priors for each multi-experimental inference (a) or a combination of uninformative (Iter 1 and 2) and informative (Iter 3) ones (c). Normalised root mean square error for RFP (left) and GFP (middle) predictions and the average (right) between samples and observables using the posteriors obtained with uninformative priors (b) and a combination of uninformative and informative ones (d).

better scores when predicting the two observables of the system for the three validation experiments. Despite these results, the nRMSE for the predictions of the first two validation experiments (Val 1 and 2) are comparable when using parameter samples from the two available posteriors. I attribute the moderately better nRMSE scores of model M^a_{OED} to better predictions of the higher dynamic range for the two observables, while model M^a_{Ran} was better at predicting the lower ones.

Predictions for the third validation experiment hold the largest difference between models M^a_{OED} and M^a_{Ran} while being the worst predicted experiment of the set. At the beginning of the experiment (first 12 hours), model M^a_{Ran} predicted two distant steady-states for the two observables, explaining most of the variability enclosed in the nRMSE. Note that this is the type of prediction that the OED scheme focused on improving, showing unique steady-state predictions for model M^a_{OED}. After the first input switch, model M^a_{Ran} predicts a constant RFP-LacI steady-state concentration without dynamics present in the experimental data due to the introduction and removal of aTc. The model M^a_{OED} showed more accurate predictions for the third validation set, yet, with some deviations from the experimental data. The model presents dynamics to the introduction and removal of aTc and accurately predicts the RFP-LacI steady-state levels, however, with dynamic delays. I observed this delay in RFP-LacI dynamics in predictions of other experiments, suggesting the need for experiments containing a higher amount of aTc switches in the inference data set. M^a_{OED} also attains the correct steady-state GFP-TetR level for most of the experiment (the last 26 hours). However, fails to predict the slight increase of GFP-TetR during the first 12 hours. The biological system shows a low basal transcription of GFP-TetR ($\approx$ 4 % of the maximum expression) in the absence of the two inducers, which gets partially repressed in the presence of RFP-LacI, with still some leakiness for the *plac* promoter regulating GFP-TetR, correctly predicted by the model.

I obtained similar results and conclusions when analysing the model predictions using the two posteriors obtained from a sequential single-experiment inference as seen in the nRMSE results represented in Fig. 7.11d. As previously stated, these sequential single-experiment inference means that, first, I performed a multi-experimental inference using the initial pulse-like experiment and another from iteration 2 (random or optimal), and then used the posteriors obtained as my priors when inferring the correspondent experiments from iteration 3. The

model using the parameter posterior inferred from optimally designed experiments (M^b_{OED}) still better predicts the three validation experiments than when using the parameter posterior inferred from randomly designed experiments (M^b_{Ran}). The most notable difference between the predictions of M^b_{OED} and M^b_{Ran} compared to M^a_{OED} and M^a_{Ran} is their lower prediction uncertainty. While I can extract the same prediction conclusions for the first two validation experiments (Val 1 and 2), M^b_{Ran} predicts a single steady-state for the entire duration of the third validation experiment (Val 3). However, M^b_{Ran} points to the wrong dominant steady-state in the absence of inducers and is unable to predict dynamics for the introduction and removal of aTc. M^b_{OED} provides similar predictions for the experiment to the ones obtained by M^a_{OED}. Such results stress the importance of prior definitions during inferences and their sequential update after ensuring an absence of overfitting. Moreover, the results point to the advantage of Bayesian OED to generate highly informative experiments for parameter inference, showing further reduction of parameter uncertainties and better recapitulating the behaviour of biological systems *in vivo*.

The supplementary Fig. S32 shows the toggle switch model M_3 prior predictions for all the experiments used for parameter inference (a-e) and validation (f-h). The plots indicate the high uncertainties and the poor prediction capabilities obtained when using highly uninformative priors. The supplementary Fig. S33 shows box plots for each inferred model parameter and each performed inference when performing multi-experimental inferences using the same uninformative priors at each iteration. The results show a general high uncertainty when using only the first pulse-like experiment. Generally, random experiments drove the means of the posteriors to different regions of the parameter space compared to optimal experiments, with lower uncertainties for the latter. These results indicate a clear difference in the informative content of the two types of designs. The supplementary Fig. S34 shows box plots for each inferred model parameter and each performed inference when performing a combination of multi-experimental (iteration 2) and sequential single-experiment (iteration 3) inferences. The results show similar results as in the supplementary Fig. S33, but with lower uncertainties for the posteriors obtained from experiments from iteration 3. As expected, the means of the parameter posteriors from iterations 2 and 3 are also closer than in the previous scenario.

The supplementary Fig. S35 shows the model predictions using the posterior

obtained in iteration 1 for all the inference (a-e) and validation (f-h) experiments. Here, overfitting becomes evident, with a low prediction uncertainty combined with poor accuracy. The supplementary Fig. S36 shows the model predictions using the posteriors obtained in iteration 2 for all the inference (a-e) and validation (f-h) experiments. Generally, the optimal design shows a slightly higher prediction uncertainty but better accuracy. The supplementary Fig. S37 shows the model predictions using the posteriors obtained in iteration 3 following multi-experimental inferences for all the inference experiments. The plots show a slightly higher prediction uncertainty for the posterior obtained using optimal experiments. In this case, the prediction accuracy is adequate for both models, but the best one depends on the experiment evaluated. The supplementary Fig. S38 shows the model predictions using the posteriors obtained in iteration 3 following a combination of multi-experimental and single-experiment inferences for all the inference experiments. Both models had low prediction uncertainties for all the experiments evaluated. Yet, posteriors obtained using optimal experiments resulted in an overall higher accuracy.

7.2.2 *In vivo* Implementation of On-line Bayesian Optimal Experimental Design Using the Genetic Toggle Switch

I used the optimal design obtained in iteration 2 of the previous section (i.e. constant IPTG at 0.64 mM) and performed on-line OED as described in Section 6.2 (p. 170) of Chapter 6, using it as a starting point. I then assessed the potential advantages of automated parameter and input updates within an experiment instead of the model-based modification previously proposed. My preliminary computational studies (see Section 7.1.3, p. 198 of this chapter) showed that on-line OED managed to overcome the issue of generating sub-optimal input designs due to highly constrained experimental setups. For this reason, I implemented such model and input design updates into our microfluidics experimental platform (more details in Section 6.2, p. 170 of Chapter 6).

First, I executed the original off-line optimal experiment from iteration 2 and inferred (BI_1) the toggle switch model M_3 to assess its informative content. Fig. 7.12a shows the progress of the experiment (for comparison to the process with an on-line optimal experiment) and the model predictions considering that

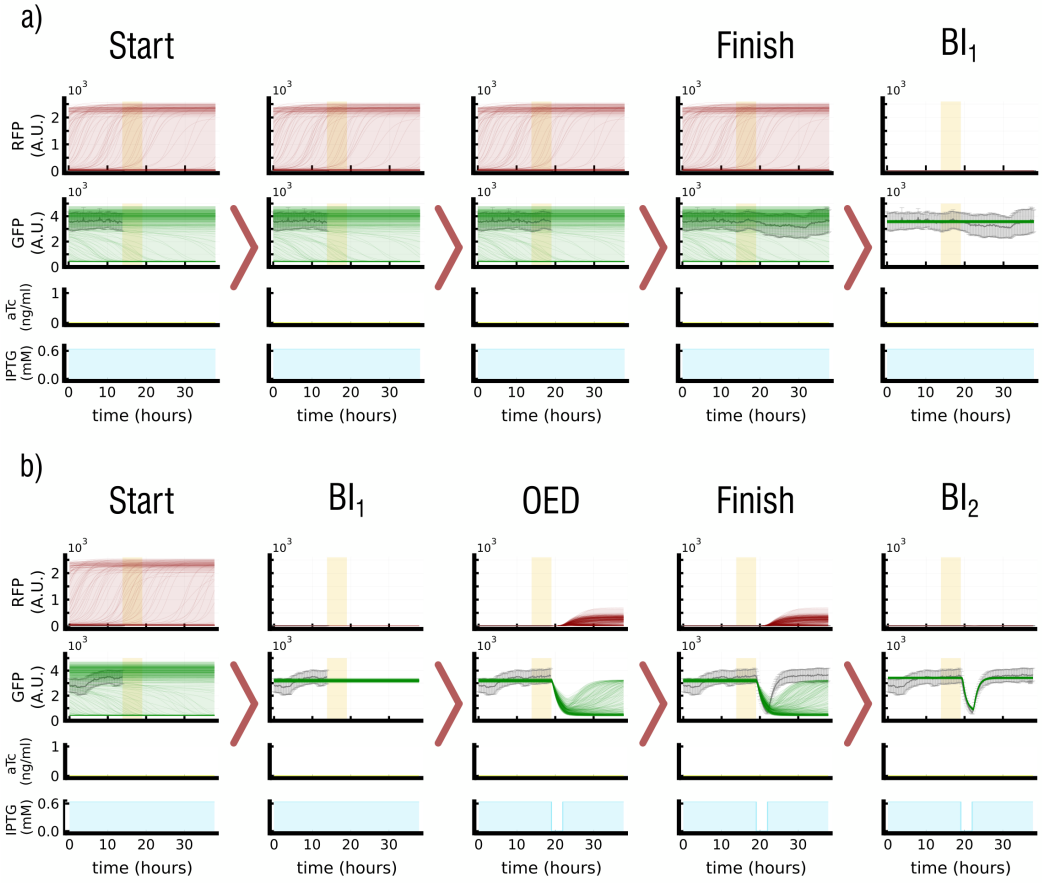


Figure 7.12: Bayesian optimal experimental design for parameter inference. a) Original off-line OED obtained for iteration 2 before the model-based experimental design adjustment. b) On-line OED starting from the constant input profile obtained for iteration 2. Experimental data are represented in grey, while model predictions and confidence intervals are represented in red (RFP-LacI) and green (GFP-TetR). The on-line OED was divided into two sub-experiments. The computational time allocated for Bayesian inference (BI) and Bayesian OED is represented in an orange area before the end of the first sub-experiment (represented but unused in the off-line OED case).

parameter beliefs only get updated after the experiment. For the optimal on-line experiment, I used the same starting conditions (IPTG = 0.64 mM, aTc = 0 ng/ml) and experimental constraints used for all the experiments (except the model-based one). Fig. 7.12b also shows the progress of the experiment and the model predictions, where in this case, the design of the former and the uncertainty of the latter vary within the same experiment due to automated Bayesian inference and OED schemes integrated within the experimental process. Due to time limitations arising from Bayesian computations, I divided the experiment into two sub-experiments of equal length (19 hours each). The first sub-experiment followed the same experimental design as the off-line counterpart. However, before this finished (at hour 14), our experimental platform temporary inferred the model parameters with the available data (BI_1) and redesigned the second sub-experiment following my Bayesian OED strategy. As priors for this inference, I used the posterior inferred from the initial pulse-like experiment of iteration 1. This selection was under sequential information update schemes, reducing the prohibitive computational time associated with highly uninformative priors and incorporating past experimental data for such a task.

The model predictions after inference for this first sub-experiment already highlighted that the model learned all it could from the data. Hence, maintaining the same constant input for the second half of the experiment resulted in a redundant task that on-line OED automatically overcame. Once the image processing platform acquired and processed all the data, I performed the last parameter inference (BI_2). I followed the same procedure as in the previous section to avoid overfitting when I used only the initial pulse-like experiment to identify the model parameters. I performed a multi-experimental inference, including the initial experiment and the off-line or on-line OED one. Consequently, these final inferences used the initially set uninformative prior $Prior_b$.

The introduction of an update on the model parameters and experimental design halfway through an experiment significantly increased its informative content. While the off-line OED highlighted a crucial characteristic of the system, the experiment provided a low reduction in parameter uncertainty when comparing priors to posteriors. Since the experiment only investigated a single system steady-state, this resulted in a minor uncertainty reduction, as my previous computational studies indicated (see Section 7.1.3, p. 198 of this chapter). The

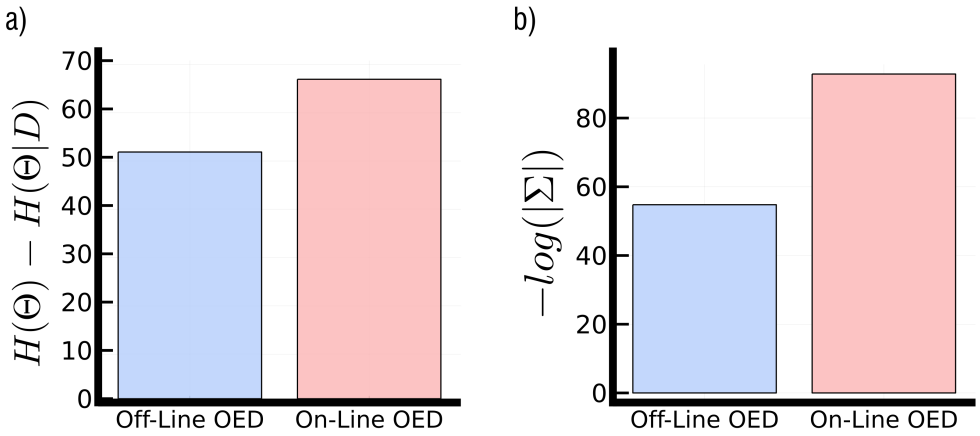


Figure 7.13: On-line OED overcomes heavily constrained experimental setups in *in vivo* scenarios. *A posteriori* gain in information measured as the difference between prior ($P(\theta)$) and posterior ($P(\theta|D)$) entropy (a) and the logarithm of the determinant of posterior covariances (b) for an off-line OED and an on-line OED that uses the off-line design as a starting point.

introduction of on-line OED starting with the same experimental conditions resulted in a posterior entropy 1.6 times smaller than the off-line counterpart. Consequently, the smaller posterior entropy resulted in a higher gain in information measured as the difference between prior and posterior entropies (Fig. 7.13a). I extracted the same conclusions when considering posterior normality and assessing the gain in information as the minus logarithm of the determinant of the covariance matrix (Fig. 7.13b).

Against my predictions, the gain in information obtained in both instances was comparable to the one obtained when using a random experiment instead of when using the model-based optimal one (Fig. 7.10). I analysed the gain in information of each marginal parameter posterior identifying that the issue focalised on the identification of three parameters; θ_{IPTG}, η_{IPTG} and η_{LacI}. If I excluded the three parameters, the marginal information gain of the rest of the parameters was comparable to the posteriors obtained when using the model-based optimal experiment for inference. The three mentioned parameters were also poorly identified in the off-line case compared to when using random experiments. Yet, the marginal gain in information for the three parameters in the online scenario was lower than a third of the ones obtained using random experiments. I identified that the most likely cause of this effect was the absence of aTc during the whole duration of one experiment in both scenarios. Hence, the

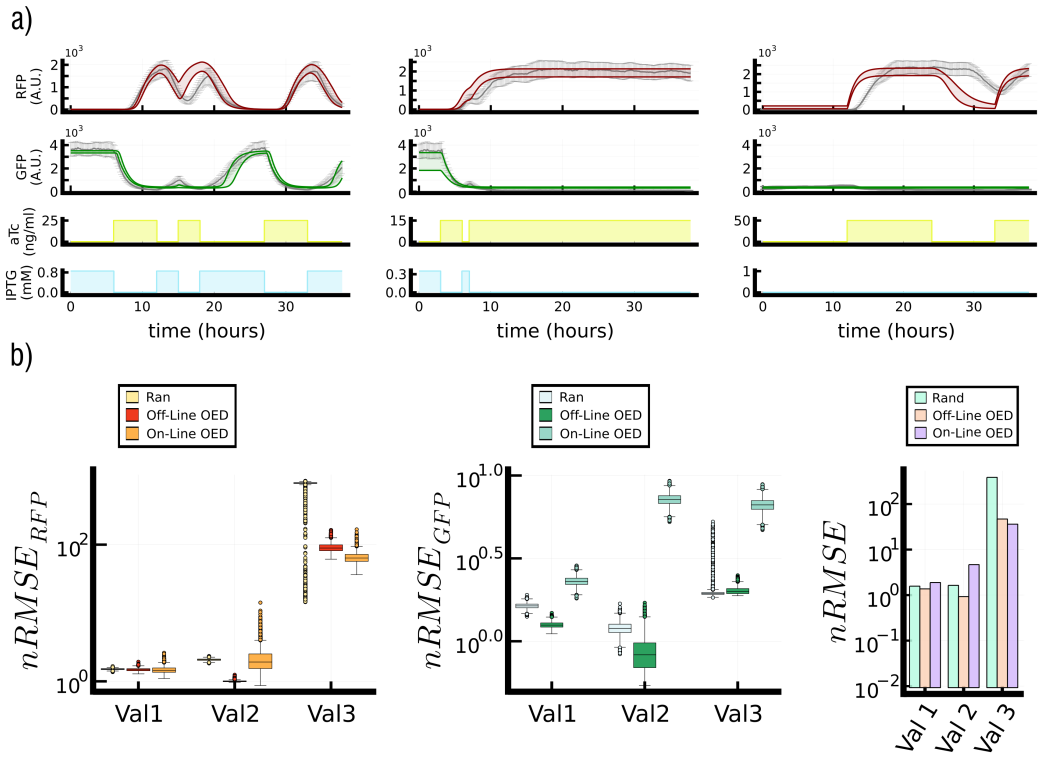


Figure 7.14: Prediction validation of a mathematical model for the genetic toggle switch inferred using data from an on-line OED. a) Validation experimental data and model predictions using parameter posteriors obtained from a multi-experimental inference of the initial (periodic input switch) and an on-line optimal experiment. b) Normalised root mean square error for RFP (left) and GFP (middle) predictions and the average (right) between samples and observables for the inferred model (on-line OED) compared with the ones inferred in Section 7.2.1 (rand and off-line OED from multi-experimental inferences using uninformative priors).

system presented a constant low basal level of RFP-LacI for the entire experiment, a term multiplied by the one containing the three parameters in the ODE describing GFP-TetR dynamics. The inducer diffusion parameters were highly identified, with a gain in information ≈ 2.4 and ≈ 1.5 times higher than the marginal posteriors obtained using random experiments for k_{IPTG} and k_{aTc}, respectively.

Despite a higher parameter uncertainty, the posterior inferred from on-line OED data (M_3^{On-OED}) showed a comparable prediction of unseen validation experiments to the other ones. First, I used the inferred model to predict the three microfluidic validation experiments used in the previous section as shown in

Fig. 7.14a. I then compared the model predictions to the ones produced by the model inferred on all random (M_3^{Ran}) or off-line optimal ($M_3^{Off-OED}$) experiments using uninformative priors. In this circumstance, the model M_3^{On-OED} had seen one experiment less than the other two. To compare the performance of the three inferred models, I computed the nRMSE between experimental data and model simulations using posterior samples for the two system observables; RFP and GFP (Fig. 7.14b, left and centre, respectively). Overall, model M_3^{On-OED} showed better RFP predictions than GFP ones. The three models performed similarly for the first two validation experiments when predicting the RFP observable. However, model M_3^{On-OED} showed a slight worst performance in predicting GFP. When observing the average nRMSE between samples and observables (Fig. 7.14b, right), I observed similar predictions by the three models. Yet, model M_3^{On-OED} scored last, which I attribute to slightly higher uncertainties in model predictions. Despite a slight worst GFP prediction, model M_3^{On-OED} scored the best nRMSE for the third validation experiment, with equivalent performance to model $M_3^{Off-OED}$. Note that model M_3^{Ran} was unable to recover the dynamics for the third validation experiment. Hence, despite a higher posterior uncertainty, model M_3^{On-OED} had a comparable prediction performance to model $M_3^{Off-OED}$. The results highlight the advantages of on-line OED and the automation of the scheme, which can suppress the need for human intervention to assess and overcome design pitfalls introduced by highly restrictive experimental constraints.

Next, I assessed the capabilities of the three inferred models (M_3^{Ran}, $M_3^{Off-OED}$, M_3^{On-OED}) to identify the fluorescence reporter dominance at steady-state in the presence of both external inducers. Due to our microfluidic setup experimental constraints, I generated the steady-state data in liquid LB cultures. I randomly selected 18 inducer concentration combinations within the allowed bounds. After I set all the experimental conditions to test, I prepared overnight cultures holding maximum IPTG or aTc to ensure that cells started at a GFP-TetR or RFP-LacI dominant state, respectively. To avoid aTc adaptation from cells, I minimised the exposure of the inducer as much as possible (6 to 8 hours) while ensuring that the system was at the desired steady-state by introducing an initial overnight growth with only LB. Then, I generated four cycles of centrifugation and media wash with fresh LB media without inducers and prepared new overnight cultures containing the different inducer combinations. Next, I imaged cells in fluorescence microscopy, keeping the same acquisition settings used in microfluidic

IPTG (mM)	aTc (ng/ml)	True	Rand	Off-Line OED	On-Line OED
1.0	100				
0.2	60				
1.0	90				
0.2	40				
0.2	10				
1.0	30				
0.9	40				
0.8	10				
0.2	90				
1.0	50				
0.1	70				
0.5	0				
0.0	30				
0.0	40				
0.8	90				
0.6	50				
0.6	30				
0.5	70				

Figure 7.15: Steady-state model validation analysis for the genetic toggle switch. Table containing 18 randomly selected IPTG and aTc concentration mixes between the allowed bounds (0-1 mM and 0-100 ng/ml, respectively). Experimental steady-state for the system (True) and the predicted steady-state of model M_3 when using the mean of the posterior inferred from random (Rand) and optimal (off-line and on-line OED) experiments. Colour red in the bacteria-like shapes indicates a monostable RFP-LacI dominant state, green a monostable GFP-TetR dominant state and yellow bistability.

experiments. I then determined whether the system had a bistable or monostable (and fluorescent reporter dominance) behaviour at each of the 18 conditions. Column True of Fig. 7.15 summarises this data by showing cell-like shapes in colour yellow to indicate bistability, colour red for RFP dominant monostability and colour green for GFP dominant monostability. I will further use this type of encoding for the data representation and prediction of steady-states throughout the remaining work of this chapter. I generated at least two biological replicates for each inducer combination and starting condition. If the system presented higher levels of RFP independently of the starting conditions, I determined monostability with RFP dominance. If I detected GFP at higher levels, I concluded monostability with GFP dominance. Consequently, if the dominant reporter depended on the initial condition for the system, then I determined bistability.

I simulated the models replicating the experimental conditions and evaluated the ratio between RFP and GFP at the last sampling time. Instead of using all the posterior samples for the simulations, I only considered their means. I designated the system as bistable when the ratio depended on the initial conditions. For monostable cases, if the RFP/GFP ratio was greater than one, I determined that the system was RFP dominant, while if the ratio was lower than one, the system was GFP dominant. A ratio of one would have indicated uncertainty of the dominant reporter, yet, none of the ratios turned out to be one. I then compared if the predicted ratios or presence of bistability matched my experimental results as shown in in Fig. 7.15.

The three models showed a low but equivalent accuracy in predicting the fluorescent reporter dominance for the 18 evaluated experimental conditions. Model M_3^{Ran} showed an accuracy of 33.3 % (Fig. 7.15, Rand), model $M_3^{Off-OED}$ of 27.8 % (Fig. 7.15, Off-Line OED) and model M_3^{On-OED} of 38.9 % (Fig. 7.15, On-Line OED). All models correctly predicted three conditions where one of the inducers was missing and a mix of inducers with low IPTG (0.1 mM) and high aTc (70 ng/ml). While model $M_3^{Off-OED}$ had the worst prediction accuracy, it pointed to bistability for all the wrongly predicted steady-states. Wrongly predicting a steady-state as bistable still allows for the possibility of a posterior to shift towards the correct region on the parameter space. On the other hand, incorrectly indicating monostability with the opposite dominant reporter connotes that the posterior distribution is likely to be in the wrong parameter

space region, suggesting the need for a new formulation for the current prior beliefs. This steady-state prediction results suggest that optimal experimental design presents an advantage in parameter identification and increased quality of model predictions. Yet, inferences would benefit from larger data set with experiments including combinations of the two inducers to obtain a more general and reliable model for the genetic toggle switch system.

The supplementary Fig. S39 shows the box plots for the marginal parameter posterior distributions of M_3^{Ran} (Ran_2), $M_3^{Off-OED}$ (OED_2) and M_3^{On-OED} ($On-LineOED$). The results manifest a general discrepancy between the first moments of the three posteriors for most of the parameters. $M_3^{Off-OED}$ had the lowest overall posterior uncertainty when assessing individual parameters. The supplementary Fig. S40 shows the model M_3^{On-OED} predictions for the initial pulse-like experiment (a) and the random (b,c) and optimal (d,e) experiments used for inferences. While the plots show a relatively small prediction uncertainty, accuracies were low, particularly for GFP. Panel f shows the predictions of M_3^{Ran} and $M_3^{Off-OED}$ for the on-line optimal experiment. Both models show good predictions for the experiment, yet $M_3^{Off-OED}$ had a higher accuracy. The supplementary Fig. S41 shows the steady-state predictions for the models inferred using only the initial pulse-like experiment and one random or optimal experiment (Iteration 2) as well as for model M_3^{On-OED}. As expected from the results in Fig. 7.15, all models resulted in poor steady-state predictions.

7.2.3 Optimal Selection of Steady-State Conditions for the Genetic Toggle Switch

In light of the steady-state prediction results from the previous section, I investigated the possibility of including such information in parameter identification schemes and assessing the best conditions to evaluate. I reduced the problem to identifying the parameter region that correctly predicted the stability properties of the system for a specific set of aTc and IPTG concentrations. Hence, in a parameter identification scheme, I can simulate a model for a particular overnight condition, extract the last sampling time and assess the ratio between the two reporters. Such an approach would shift our prior beliefs to a region of the parameter space that complies with the predefined steady-state conditions. However, this approach would be unable to distinguish

between different parameter sets inside the domain due to the qualitative nature of the data. For this reason, I conclude that the best approach is to include these steady-state evaluations as an additional constraint in parameter identification when using my microfluidics data.

After exploring different alternatives, I identified the best option for this particular case as introducing a penalty term for misidentified steady-states. As previously done, I simulate a model with a specific overnight starting condition (i.e. RFP or GFP dominance) and inducer concentrations, extract the last sampling time and compute the RFP/GFP ratio. I identified parameter vectors that generated ratios closer to one during preliminary tests. For this reason, I excluded the previously stated binary selection of reporter dominance. To do so, I assumed that a reporter needs to be three times smaller or higher than the other to be confident in assessing the dominant fluorescent reporter. Consequently, if the RFP/GFP ratio is higher than 3, the model predicts RFP dominance (with probability $p = 1$), and if it is smaller than $\frac{1}{3}$, it indicates GFP dominance (with probability $1 - p = 1$). Hence, for a ratio equal to 1, both reporters should present the same probability ($p = 0.5$). If the ratio falls between these two values, I assigned a probability p for RFP dominance and $1 - p$ for GFP dominance following a logarithmic function of the form:

$$p = 0.5 + (1/\log(9)) \cdot log(RFP/GFP) \tag{7.1}$$

This logarithmic function ensures symmetry of provability decrease/increase in both directions from ratios equal to 1. This means that a ratio of 2 ($p \approx 0.81$) will have the same amount of increase in p considering the ratio 1 ($p = 0.5$) as my reference point, than of decrease when the ratio is $\frac{1}{2}$ ($p \approx 0.19$), where in this case the difference is of ≈ 0.31 in both directions. The parameters are result of fitting the function considering the previously stated conditions for p. Due to the nature of the logarithmic function and to ensure that p falls within a probability range (0-1), when the function resulted in values higher than 1 (ratio higher than 3) these were set to 1, and when the function resulted in values lower than 0 (ratio smaller than $\frac{1}{3}$), these were set to 0.

At this point, I predefined a penalty term to add during MLE or subtract during

Bayesian inference into our cost function (to be minimised) or target posterior probability (to be maximised), respectively. Hence, if experimental data pointed to RFP dominance, I multiplied the penalty term by $1 - p$, while if it indicated GFP dominance, I multiplied it by p. Using this approach, RFP/GFP ratios that were too close to 1 still contributed to the penalty term, but not as much as a wrong one. In cases where I included multiple steady-state evaluations during parameter identification, I followed an additive scheme of the penalty terms.

I also evaluated my Bayesian strategy to assess *a priori* the information content of experiments to optimally select steady-state concentrations to add as constraints during parameter identification (see metric PU_e^C from Section 3.2.5, p. 84 of Chapter 3). In this scenario, I considered as priors the posterior obtained after inferring model M_3 using the initial pulse-like experiment since my original priors resulted in high prediction uncertainty in all cases. As a microfluidic experiment for parameter inference, I considered the on-line optimal one. Since I examined each of the 18 steady-state conditions evaluated in Section 7.2.2 (p. 217) of this chapter with cells starting in an RFP or GFP steady-state, I considered these as 36 different ones. Then I simulated the model with 8000 posterior samples for each condition and evaluated the prediction uncertainty at the last sampling time. I then picked the 14 ones with the highest prediction uncertainty as my optimally selected set. For comparability, I additionally randomly selected 14 conditions from the remaining 22 as a randomly selected set. I then inferred the model using the initial pulse-like experiment and the on-line optimal experiment using as priors $Prior_b$ (Table 3.7, p. 67 of Chapter 3) to avoid issues from overfitting and including one of the two sets of steady-state conditions as inference constraints. I performed MLE to obtain a good initial guess to reduce the computational time associated with the inference.

The optimally selected constraint conditions significantly increased the model prediction accuracy for steady-state ratios. To generate a good initial guess for the model parameters, I initially performed MLE. I introduced a high penalty for each wrong steady-state ratio predicted (addition of 10^6 to the likelihood term for each wrongly predicted condition) to ensure that this initial guess complied with all the considered constraints. The best parameter vector obtained using optimally selected steady-state ratios ($M_{3-MLE}^{OLOptSS}$) obtained a significantly higher accuracy than when I used randomly selected ones ($M_{3-MLE}^{OLRanSS}$) for the constraints as represented in Fig. 7.16. When I consider the 36 different

IPTG (mM)	aTc (ng/ml)	True	Rand SS	Optim SS
1.0	100			
0.2	60			
1.0	90			
0.2	40			
0.2	10			
1.0	30			
0.9	40			
0.8	10			
0.2	90			
1.0	50			
0.1	70			
0.5	0			
0.0	30			
0.0	40			
0.8	90			
0.6	50			
0.6	30			
0.5	70			

Figure 7.16: Steady-state model validation analysis for the genetic toggle switch after MLE using reporter ratios as constraints. Table containing 18 randomly selected IPTG and aTc concentration mixes between the allowed bounds (0-1 mM and 0-100 ng/ml, respectively). Experimental steady-state for the system (True) and the predicted steady-state of model M_3 when using the best parameter vector according to MLE when estimated using the initial pulse-like experiment and the on-line optimal one with 14 randomly (Rand SS) or optimally (Optim SS) selected steady-state ratios (splitting the 18 as 36 with two different initial conditions) as inference constraints. Colour red in the bacteria-like shapes indicates a monostable RFP-LacI dominant state, green a monostable GFP-TetR dominant state and yellow bistability.

conditions (two opposite initial system conditions for each one assessed), the model $M_{3-MLE}^{OLOptSS}$ achieved an accuracy of 97.2 %, while model $M_{3-MLE}^{OLRanSS}$ of 75 %. Since I am using a biological system capable of presenting bistability, if I consider only the 18 conditions with both steady-states, the accuracies were 94.4 % and 50 %, respectively.

I also quantified using the nRMSE the prediction capabilities of the two models when predicting all the microfluidic experiments presented in this work as represented in Fig. 7.17. I performed this analysis to assess if the introduction of constraints had a significant deleterious effect on the prediction of the model dynamics. The results show a slight decrease in the prediction capabilities of both models. However, both were still capable of recapitulating, to some extent, experimental dynamics. Model $M_{3-MLE}^{OLOptSS}$ resulted in better nRMSE scores than model $M_{3-MLE}^{OLRanSS}$ (represented with the lower nRMSE values in Fig. 7.17c), particularly for GFP-TetR predictions (Fig. 7.17b). For most experiments, the nRMSE is similar for both models when considering RFP-LacI (Fig. 7.17a). Yet, for four unseen experiments (Ran_1, OED_1, Val_1, Val_3), model $M_{3-MLE}^{OLRanSS}$ resulted in poor ones, primarily due to dynamic delays.

The supplementary Fig. S42 shows the predictions for all the microfluidic experiments included in this work of the toggle switch model M_3 using the best parameter estimates according to MLE using the initial pulse-like experiment and the on-line OED one and randomly or optimally selected constraints. The results indicate that the two groups of constraints had a distinct effect on the model's predictive capabilities. While optimal constraints resulted in an overall better prediction accuracy, this depended on the experiment considered.

The only difference between the two models was the steady-state observable ratio constraints. Nonetheless, optimally selected ones improved both types of predictions, steady-state and dynamic ones, resulting in a more generalisable model. I observed similar general results independently of the amount or type of microfluidic experiments included in the MLE (data shown in supplementary figures). However, the microfluidic experiments included strongly affected the prediction error of other microfluidic experiments, deeming it crucial to predict system dynamics. These results highlight the importance of both types of data to infer parameters of bistable systems and the advantage that my OED strategy poses to evaluate the informative content of experiments *a priori*.

The supplementary Fig. S43 shows the steady-state observable ratio predictions for the model using the best parameter estimate according to MLE. The approach included randomly selected observable ratios as constraints using the sequential inference scheme from Section 7.2.1 (p. 207) of this chapter, with the randomly designed microfluidic experiments. This combination of randomly designed experiments and selected constraints resulted in low prediction accuracy of the steady-state ratios at all process iterations, except for the conditions included as constraints. The supplementary Fig. S44 shows the steady-state observable ratio predictions where MLE includes randomly selected ratios as constraints using the optimally designed microfluidic experiments. Again, these randomly selected constraints resulted in poor steady-state ratio predictions, independently of the microfluidic experiments considered. The supplementary Fig. S45 shows the steady-state observable ratio predictions where MLE includes optimally selected ratios as constraints using the randomly designed microfluidic experiments. Using optimally selected constraints resulted in prediction accuracy of steady-state ratios of 100 %, indicating that these contain a crucial degree of information for the task. The supplementary Fig. S46 shows the steady-state observable ratio predictions where MLE includes optimally selected ratios as constraints using the optimally designed microfluidic experiments. Again, choosing optimal constraints results in a high steady-state ratio prediction accuracy (94 %) with only one wrong prediction.

The supplementary Fig. S47 shows the nRMSE for RFP-LacI (left), GFP-TetR (middle) and the mean of both observables (right) of the model predictions for all microfluidic experiments included in this work when using the best parameter estimates obtained with MLE using randomly or optimally designed microfluidic experiments and randomly or optimally selected steady-state observable ratios as constraints. As expected, using only the initial pulse-like experiment resulted in the worst model predictions, independently of the constraints used. Generally, introducing these constraints had a strong effect in predicting GFP but a significantly smaller one for RFP. Yet, it appears inconclusive to select the best type of constraint by observing the prediction of microfluidic experiments since the apparent quality of this selection seems to change case by case. The overall prediction quality seemed comparable for most conditions regardless of the type of selection done for the constraints.

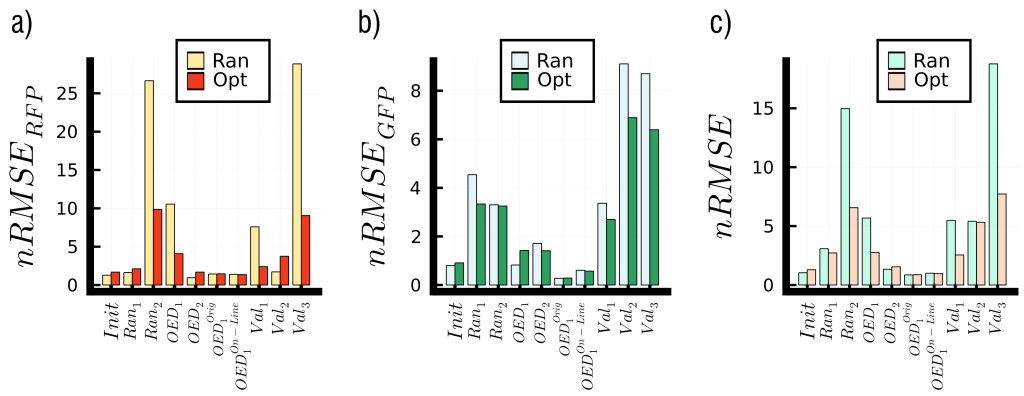


Figure 7.17: Microfluidics data model validation analysis for the genetic toggle switch after MLE using observable ratios as constraints. Normalised root mean squared error (nRMSE) of the toggle switch model for RFP-LacI (a), GFP-TetR (b) and the mean for both observables (c) using the best parameter estimate according to MLE. The parameter estimation strategy used the on-line OED microfluidic data and observable ratios as constraints predicting all the microfluidic experiments included in this work. The predictions that used the best parameter estimate from MLE that included randomly selected steady-state ratios as constraints are indicated as Ran, and the one using optimally selected ones as Opt. For the optimally designed microfluidic experiments, OED_1 is the model-based modified one, and OED_1^{Orig} is the one without modifications (output from the optimisation scheme).

The supplementary Fig. S48 shows the model predictions for the models using the best parameter estimates according to MLE using steady-state ratio constraints with the initial pulse-like microfluidic experiment. As indicated in the supplementary Fig. S47, the plots show poor model predictions independently of the type of constraint set used. The supplementary Fig. S49 shows the model predictions for the models using the best parameter estimates according to MLE using steady-state ratio constraints with the initial pulse-like and the first randomly designed microfluidic experiments. Predictions seem comparable between both cases for all the experiments, with small dynamic differences for the prediction of some experiments. The supplementary Fig. S50 shows the model predictions for the models using the best parameter estimates according to MLE using steady-state ratio constraints with the initial pulse-like and the two randomly designed microfluidic experiments. The predictive difference for the models was slightly higher than in the previous supplementary figure, especially for the validation experiments. The supplementary Fig. S51 shows the model

predictions for the models using the best parameter estimates according to MLE using steady-state ratio constraints with the initial pulse-like and the first off-line optimally designed microfluidic experiments. In this case, the predictive difference between the two models was more apparent, benefiting the optimal selection of constraints for most experiments evaluated. The supplementary Fig. S52 shows the model predictions for the models using the best parameter estimates according to MLE using steady-state ratio constraints with the initial pulse-like and the two off-line optimally designed microfluidic experiments. Once again, the predictions of both models were comparable for most of experiments evaluated, with the biggest differences in the three validation experiments.

I then introduced into the Bayesian inference scheme the different randomly or optimally selected steady-state observable ratios as penalisation terms, obtaining models $M_{3-Bay}^{OLRanSS}$ and $M_{3-Bay}^{OLOptSS}$, respectively. Such penalisation influenced the sampling efficiency of the MCMC algorithm used, suggesting a complicated posterior shape to explore. To resolve the issues, I re-scaled the different marginal prior distributions expanding the range of difficult-to-sample parameters to allow a larger and more efficient step size. Such transformations involved mapping the normally distributed priors to standard normal ones and adjusting their standard deviation. I then mapped the posterior samples obtained to the original prior range, getting the correct parameter posteriors for the toggle switch model. I also had to introduce a significantly lower penalisation term (between 1 and 10 added to the target log probability) for each wrongly predicted ratio compared to when performing MLE. Higher penalisation terms made the sampling even more complicated, making it unfeasible for some parameters (e.g. k_{aTc}, θ_{aTc}, k_T^{pm}, θ_{LacI} or η_{TetR}). Additionally, I initialised all the MCMC chains from the best parameter vector obtained from MLE, significantly reducing the number of warm-up samples required for inference.

The introduction of steady-state observable ratios as inference constraints had a negligible effect on parameter uncertainties but a significant one on model predictive capabilities. Independently of the group of ratios used as constraints (randomly or optimally selected) for inference penalisation, the posteriors obtained had a comparable uncertainty to the inference that excluded them (model M_3^{On-OED}). I quantified the gain in information using entropy (Fig. 7.18a) or a metric summarising variances/covariances (Fig. 7.18b). Nonetheless, I noticed that the different constraints highly affected the position of the different

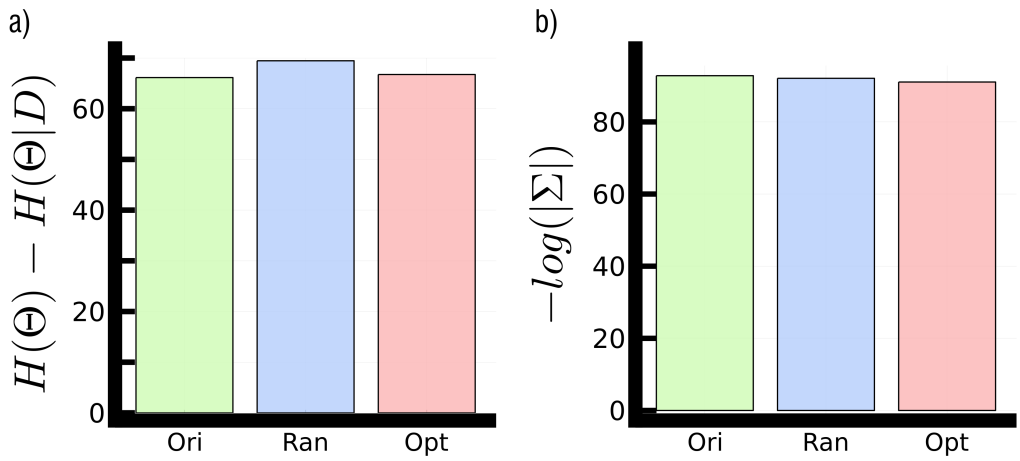


Figure 7.18: Steady-state observable ratio constraints effect on parameter posterior uncertainty. Parameter uncertainty reduction measured as the difference between prior and posterior entropy (a) and the logarithm of the determinant of the posterior covariance matrix (b). All inferences included the initial pulse-like and on-line OED microfluidics data (Ori) and additionally randomly (Ran) or optimally (Opt) selected steady-state observable ratios as constraints.

posteriors in the parameter space. These results mean that introducing such penalisations in inferences excluded regions of the parameter space that resulted in wrong observable ratio predictions. Hence, since posterior uncertainty resulted ineffective to assess the quality of the inferred posteriors, I relied on validation data (both, microfluidics and steady-state observable ratios) for the task.

The supplementary Fig. S53 shows box plots for each inferred parameter of models M_3^{On-OED} (None), $M_{3-Bay}^{OLRanSS}$ (Ran) and $M_{3-Bay}^{OLOptSS}$ (Opt). The plots show how the most significant effect of the constraints is the localisation of the first moment of the posterior for almost all the parameters. Hence, the constraints aid in finding the region of the parameter space where the model predicts the selected steady-state ratios while having a null effect on the shape of the posterior within this region.

The type of steady-state observable ratio constraints introduced as penalty terms during inferences significantly affected the predictions of the dominant fluorescent reporter. Parallel to the results obtained when performing MLE, using model prediction uncertainty to select optimal steady-state observable ratios resulted in higher accuracy when predicting this type of data as shown in Fig. 7.19. When

IPTG (mM)	aTc (ng/ml)	True	Rand SS	Optim SS
1.0	100			
0.2	60			
1.0	90			
0.2	40			
0.2	10			
1.0	30			
0.9	40			
0.8	10			
0.2	90			
1.0	50			
0.1	70			
0.5	0			
0.0	30			
0.0	40			
0.8	90			
0.6	50			
0.6	30			
0.5	70			

Figure 7.19: Steady-state model validation analysis for the genetic toggle switch after Bayesian inference using reporter ratios as constraints. Table containing 18 randomly selected IPTG and aTc concentration mixes between the allowed bounds (0-1 mM and 0-100 ng/ml, respectively). Experimental steady-state for the system (True) and the predicted steady-state of model M_3 when using the mean of the posterior inferred using the initial pulse-like experiment and the on-line optimal one with 14 randomly (Rand SS) or optimally (Optim SS) selected steady-state ratios (splitting the 18 as 36 with two different initial conditions) as inference constraints. Colour red indicates a monostable RFP-LacI dominant state, green a monostable GFP-TetR dominant state and yellow bistability.

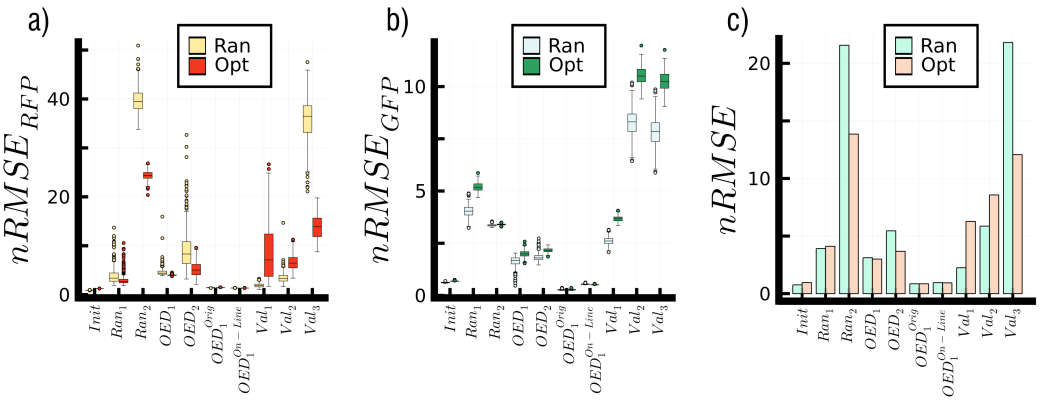


Figure 7.20: Model validation analysis for a model describing the genetic toggle switch employing microfluidics data for Bayesian inference using observable ratios as constraints. Box plots containing the nRMSE for model M_3 using posterior samples to predict RFP-LacI (a) and GFP-TetR (b). c) Average nRMSE between posterior prediction samples and observables. Both inferences used the initial pulse-like experiment and the on-line OED microfluidics data, including randomly (Ran) or optimally (Opt) selected observable ratio constraints. For the optimally designed microfluidic experiments, OED_1 is the model-based modified one, and OED_1^{Orig} is the one without modifications (output from the optimisation scheme).

considering the 36 different conditions (starting from two different initial conditions), model $M_{3-Bay}^{OLOptSS}$ achieved an accuracy of 94.4 %, while model $M_{3-Bay}^{OLRanSS}$ of 58.3 %. If I consider the correct selection of bistability or monostability (including reporter dominance), for the 16 conditions assessed, model $M_{3-Bay}^{OLOptSS}$ achieved an accuracy of 88.8 %, while model $M_{3-Bay}^{OLRanSS}$ of 44.4 %. I noted that the results obtained are slightly different from the ones obtained using the MLE results. I attribute this outcome to the lower penalisation (hence, a lesser effect of the constraints) used during Bayesian inferences to ease the posterior sampling and the use of the mean of the posterior distribution obtained to simulate the model.

Model prediction quality for microfluidic experiments was comparable for models $M_{3-Bay}^{OLRanSS}$ and $M_{3-Bay}^{OLOptSS}$. I assessed the predictive capabilities of both models by computing the nRMSE between simulations and data for all the microfluidics experiments generated in this work. For RFP predictions (Fig. 7.20a), model $M_{3-Bay}^{OLOptSS}$ better predicted experiments Ran_2, OED_2 and Val_3. The model showed lower means and standard deviations for the nRMSE computed using 400 model simulations employing randomly selected posterior samples. Model

$M_{3-Bay}^{OLRanSS}$ better predicted experiments Val_1 and Val_2. For the rest of the experiments, both models had comparable results. When considering GFP predictions (Fig. 7.20b), the two models achieved similar results in terms of the mean and standard deviation for the nRMSE. The computation of the nRMSE average between simulation samples and observables (Fig. 7.20c) better highlights the comparability between the prediction of both models. The most notable prediction quality differences were for experiments $Rand_2$ and Val_3 favouring model $M_{3-Bay}^{OLOptSS}$ and experiments Val_1 and Val_2 favouring model $M_{3-Bay}^{OLRanSS}$. I also observed a general decrease in the overall prediction error compared to the result of inferences excluding the observable ratio constraints, as observed when analysing the MLE results. Hence, by including steady-state observable RFP/GFP as constraints, I obtained more generalisable models for the genetic toggle switch, especially when optimally selecting these (i.e. $M_{3-Bay}^{OLOptSS}$).

The supplementary Fig. S54 shows the posterior predictions for all the microfluidic experiments generated in this work using models $M_{3-Bay}^{OLRanSS}$ and $M_{3-Bay}^{OLOptSS}$. Overall, model $M_{3-Bay}^{OLRanSS}$ had higher prediction uncertainties. Accuracies between the two models were comparable for most experiments, with the largest differences in the validation set. The supplementary Fig. S55 shows the nRMSE for all the microfluidic experiments generated in this work using model M_3^{On-OED} for the simulations. The prediction uncertainty error was significantly higher for RFP (panel a) compared to GFP (panel b). Because of this larger RFP error, the model had poor mean predictions (panel c) for the second random experiment, the off-line optimal experiments from iterations 2 and 3 and the third validation experiment (the most troublesome for all models to predict).

7.2.4 Conclusions

Model prediction uncertainty is an efficient and advantageous tool to evaluate the informative content of experiments in a Bayesian context. Such a sampling-based method allows for designing or selecting highly informative experiments directed at parameter inference. In this section, I tested the strategy in an *in vivo* scenario using the genetic toggle switch developed by Lugagne *et al.* 2017 [85] as a case study. OED methods, particularly in Bayesian schemes, are still mostly limited to computational studies in biological sciences. I showed the methodology advantages in an *in vivo* scenario. The results underlined its potential to reduce

the tedious experimental work when inferring model parameters or, at least, obtaining better results with a predefined budget.

First, I compared the Bayesian OED strategy to random input design for microfluidic experiments. Such experiments are characterised by a high amount of data, access to observe system dynamics and the ability to modulate external inducers over time. Yet, these experiments come at a high cost in terms of time and resources. Hence, ensuring maximum informative content is crucial. I showed how OED generates highly informative data, evaluated with parameter uncertainty reduction and the capability of the inferred model to predict unseen experiments. However, restrictive experimental constraints might result in sub-optimal input designs, as I observed previously in computational studies. I showed how we could significantly increase the informative content of these sub-optimal designs by analysing the optimisation results and employing a model-based informed modification. Yet, while combining both methods yields the best results, it involves human intervention, limiting the automation of the process.

To overcome the sub-optimal designs generated due to strict experimental constraints, I implemented an on-line Bayesian OED platform, automating the process. Previous computational results suggested the potential of on-line OED to avoid these constraint-caused sub-optimal designs. For this reason, I modified our microfluidics setup to perform Bayesian inference with a sub-set of the experimental data and update the input design during an experiment to increase its informative content. While my platform was able to generate on-line Bayesian OEDs, the computational complexity of the task limited the number of updates done within an experiment. I only tested the case of one inference and input update halfway through the experiment using the sub-optimal design as a starting point. Yet, the resultant data was significantly more informative than the one using the original design in terms of parameter uncertainty reduction and prediction capabilities. When comparing the results with the ones obtained using other experiments, the on-line optimal experiment achieved similar parameter uncertainty reduction results to when I used random experiments. However, the prediction capabilities of the resultant model were comparable to the ones obtained when using the off-line experimental design. The results indicate the advantage of on-line OED when process automation is desired. Yet, human intervention in the early steps of the process might be highly beneficial in some

scenarios.

Finally, I tested the prediction capabilities of all the inferred models using experimental conditions that our microfluidics setup was incapable of performing: fluorescent reporter dominance in the presence of both external inducers at varying concentrations. The model inferred with microfluidics data had low accuracy for predicting the dominant reporter, independently of the type of experimental design used. For this reason, I tested introducing such reporter ratios (RFP/GFP) as constraints in MLE and Bayesian inference schemes. I also used the model prediction uncertainty to assess its potential in selecting the best inducer conditions from a predefined set to use as constraints. The results showed that, while one should take care in introducing a penalisation term for the constraints, these improve the predictive capabilities of the inferred models. Moreover, optimally selected constraint conditions resulted in significantly higher steady-state ratio accuracies and better dynamic predictions of microfluidics data. These results validated, once again, the benefits of assessing the informative content of experiments *a priori*. The results also suggest the potential need for different kinds of experimental data to infer model parameters *in vivo* due to the complexity of the task. This complexity can come from sparse and noisy data, restrictive laboratory setup constraints, inappropriate assessment of prior distributions or slight imprecise representation of a biological system by a mathematical model.

7.3 Chapter Conclusions

OED is a powerful tool to generate highly informative data for a specific task, aiding the arduous process of model development, as shown in the case of model selection (Chapter 5). In this chapter, I proposed and validated a Bayesian approach to maximise the informative content of experiments aimed at parameter inference. The strategy leverages parameter uncertainty to identify the experimental designs where this translates to poor model predictions and hence, the ones we can learn more from. First, I used a simple case study to prove the potential validity of the strategy. I aimed to infer four parameters of a model describing a hypothetical genetic system. Here I showed how, while an intuition-driven approach to experimental design can be sufficient to identify most model parameters, the OED strategy poses a better strategy. Since we are using a Bayesian approach, sequential updates of parameter priors allowed me to update the experimental designs sequentially using the parameter knowledge at each step. This approach proved even more efficient at identifying all four parameters of the model considered. To further test the validity of this Bayesian OED approach, I scaled up this computational study to a more complex scenario. I used a model describing a genetic toggle switch (with 14 parameters to infer) to compare random and optimal dynamic input designs for its two external inducers (i.e. aTc and IPTG). Here, I used information theory, distance metrics and stability analysis to prove that, even in more complex scenarios, the Bayesian OED strategy still provides a significant advantage for the task of parameter inference. Yet, restrictive experimental constraints (although necessary in some lab settings) can have a detrimental effect on the optimisation process, resulting in sub-optimal solutions at early stages. In such situations, I showed how an automated approach for sequential parameter knowledge and input design updates within an experiment (i.e. on-line OED) could, in practice, surpass some of these issues, generating the most informative experiments.

After the positive results from the computational study for the Bayesian OED strategy proposed, next, I applied this in an *in vivo* scenario using the genetic toggle switch system developed in *E. coli*. Since the end goal of these statistical methodologies is to implement them experimentally to make model development easier, I tested them using fluorescence microscopy data from microfluidic experiments. I used this data to infer the parameters of a mathematical model describing the system, comparing random and optimal dynamic input designs as I

did in the previous computational analysis. As expected by these previous analyses, our experimental platform constraints of our platform resulted in sub-optimal designs at the beginning of the process. To solve this, I introduced a model-based design modification for the first iteration of optimisations, successfully avoiding the issue down the line. In this study, OED (combined with a model-based design) manifested its advantages, reducing the time and resources I needed to invest in experimentation to get equivalent or better inference results compared to a random design of input profiles. To assess this, I used an information theoretic approach. Yet, in these situations, we do not know the exact parameter values since this would redeem this task as something unnecessary. Consequently, I also included a validation data set to assess the model's prediction quality using the different inferred posteriors. Again, the model using parameters inferred with optimally designed data achieved the best results. Next, to see if process automation could aid the detrimental effect of our experimental platform constraints, I used the on-line OED platform described in Section 6.2.3 (p. 177) of Chapter 6. In this case, while the approach presented a significant advantage compared to the originally sub-optimally designed experiment, the reduction of uncertainty analysis indicated better results for the random input design. However, validation studies showed that the model using the posteriors inferred from the on-line OED presented better predictions of unseen data. Since most of the issues encountered in this study were related to the prediction of steady-states (a crucial property for this system), I included a stability analysis. All the inferred models had poor predictive capabilities when predicting steady-states for inducer conditions that I was unable to introduce in my microfluidic experiments due to our platform constraints. Hence, I studied how to incorporate this information as a constraint for our Bayesian inference scheme to improve model prediction capabilities. I also used the prior prediction uncertainty to select the optimal steady-states to include as constraints from a predefined set. The gain in information achieved in this case was similar to when excluding such constraints. However, when I introduced these constraints in the inference, the model's predictive capabilities drastically improved. Furthermore, using the optimally selected constraints resulted in significantly better model predictions after inference than when selecting these randomly, proving the advantages of OED once more. Thanks to the promising results of this study, I included a generalised version of the Bayesian OED methodology as a part of the Julia package described in the following Chapter 8, BOMBs.jl.

Chapter 8

BOMBs.jl: A Julia Package to Ease Simulations, Parameter Inference and Bayesian Optimal Experimental Design for Model Selection and Calibration

The package described in this chapter is included in the general Julia registry and can be installed following the Julia package manager commands (Pkg.add("BOMBs")). The package was presented at the ***JuliaCon 2022 conference*** *as a lightning talk titled* ***An introduction to BOMBs.jl****.*

Mathematical models are a crucial tool to predict [354], understand [202], control [109] and ultimately design [28] biological systems. However, working with models stands an arduous labour, particularly for inexperienced users, hindering the widespread use of these amongst the biological community. Some of the most common, yet critical, steps in the development of models focus on how to select the best candidate model, how to infer kinetic rates or how to identify the most informative experiments for a task. In this chapter, I will describe BOMBs.jl, an open-source low-code Julia package I developed to ease work with models, automating the most labour-intensive aspects of the process (Fig. 8.1). First, I will describe how the package automatically generates all the scripts to simulate models with time-varying parameters (i.e. external inducers) with the minimum information required by the user. Then, I will show how BOMBs.jl eases the simulations of these models and automates the generation of pseudo-data for

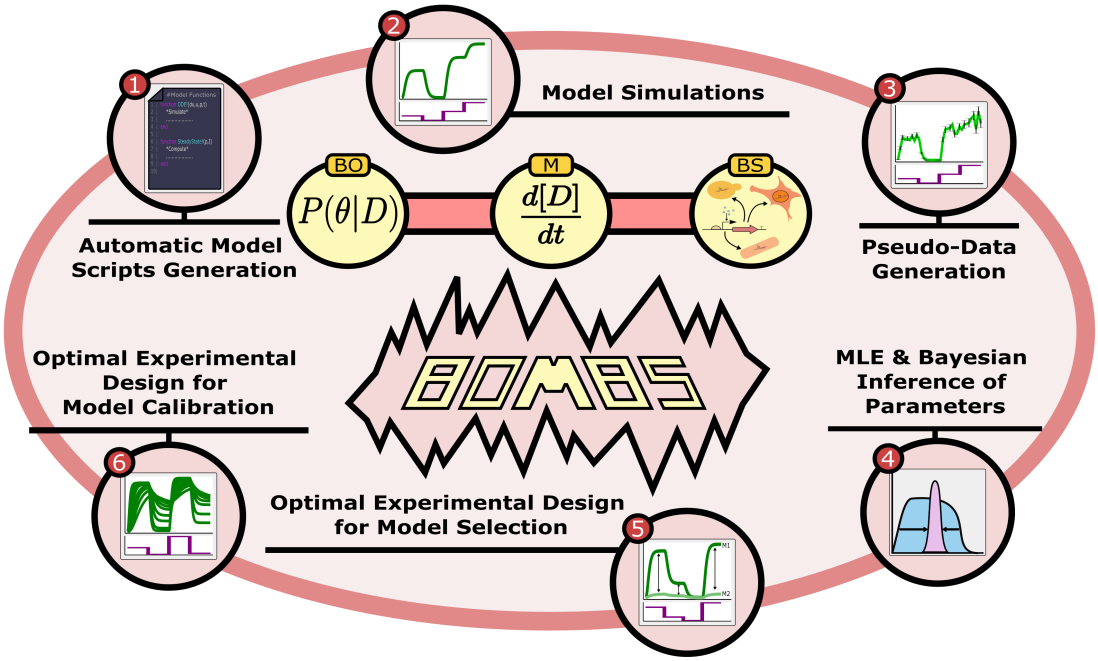


Figure 8.1: Bayesian optimisation of models for biosystems (BOMBs). Logo and schematic of all the functionalities included in the Julia package BOMBs.jl. The package works with ordinary differential equations defined by the user, automating Julia scripts generation, model simulations and pseudo-data traces, maximum likelihood estimation, Bayesian inference schemes and optimal experimental designs for model selection and parameter inference.

computational studies. Next, I will show how the package automates the estimation of parameters for inexperienced users using maximum likelihood estimation or Bayesian inference (via Stan or Turing.jl) to relax noise assumptions associated to parameters (as described in Chapter 4). Finally, I will show how BOMBs.jl automates the optimisation of experiments focused on model selection or inference by integrating generalisations of the methodologies described in sections 5.1 (p. 141) of Chapter 5 and 7.1 (p. 186) of Chapter 7, respectively. I have designed BOMBs.jl to facilitate the widespread use of models across life sciences, making it user-friendly and easy to use. Beginner users will take advantage of numerous suggestions within the package (error messages will point users to the most likely mistake), extensive documentation (document included in the GitHub repository) and detailed examples showing the use of all the package functionalities (in the form of Jupyter notebooks located in the GitHub repository).

```
model_def = defModStruct( );
model_def["NameF"] = ["TSM3Example"];
model_def["nStat"] = [4];
model_def["nPar"] = [14];
model_def["nInp"] = [2];

model_def["stName"] = ["IPTG","aTc","RFP_LacI","GFP_TetR"];
model_def["inpName"] = ["u_IPTG", "u_aTc"];
model_def["parName"] = ["k_IPTG" ,"k_aTc" ,"k_L_pm0" ,"k_L_pm" ,"theta_T" , "theta_aTc" ,"n_aTc" ,"n_T" , "k_T_pm0"
,"k_T_pm" , "theta_L" , "theta_IPTG" ,"n_IPTG" ,"n_L"];

model_def["eqns"] = [
"dIPTG=k_IPTG*(u_IPTG-IPTG)-0.0165*IPTG;",
"daTc=k_aTc*(u_aTc-aTc)-0.0165*aTc;",
"dLacI=((1/0.1386)*(k_L_pm0+(k_L_pm/(1+(TetR/theta_T*1/(1+(aTc/theta_aTc)^_aTc))^n_T))))-0.0165*LacI;",
"dTetR = ((1/0.1386)*(k_T_pm0+(k_T_pm/(1+(LacI/theta_L*1/(1+(IPTG/theta_IPTG)^n_IPTG))^n_L))))-0.0165*TetR;"];

model_def["Y0eqs"] = [
"IPTG = (k_IPTG*u_IPTG)/(k_IPTG+0.0165);",
"aTc = (k_aTc*u_aTc)/(k_aTc+0.0165);",
"LacI = ((1/0.1386)*(k_L_pm0+(k_L_pm/(1+((expTetR/theta_T)*1/(1+(aTc/theta_aTc)^n_aTc))^n_T))))/0.0165;",
"TetR= ((1/0.1386)*(k_T_pm0+(k_T_pm/(1+((expLacI/theta_L)*1/(1+(IPTG/theta_IPTG)^n_IPTG))^n_L))))/0.0165;"];

model_def["Y0Sim"] = [true];
model_def["tols"] = [1e-14, 1e-14];
model_def["solver"] = ["CVODE_BDF"];

model_def = GenerateModel(model_def);
```

Figure 8.2: BOMBs.jl automatically generates scripts for model simulations. Code necessary to automatically generate all the Julia model simulation functions with BOMBs.jl using a mathematical model for the genetic toggle switch (M_3) as an example.

8.1 Functionality and Architecture of BOMBs.jl With an Illustrative Case Study Using a Model for the Genetic Toggle Switch

8.1.1 Model Definition

BOMBs.jl automatically generates any necessary scripts to use ordinary differential equations (ODEs) in Julia, easing work with mathematical models by decreasing the coding expertise needed from users. The package allows models with time-varying inputs thanks to the event input implementation described in Section 3.2.1.4 (p. 73) of Chapter 3. I designed the package to mainly work with dictionaries that users need to fill with the relevant information about their models. Users can retrieve the main dictionary structure to generate model scripts by calling the function defModStruct(). Then, users need to fill the respective dictionary fields with a string that will be attached to all files for

distinction ("NameF"), the number and names of states ("nStat" and "stName", respectively), parameters ("nPar" and "parName", respectively) and external inputs ("nInp" and "inpName", respectively) for the model, and the set of ODEs ("eqns") introduced as a vector of strings for each separate line in the model. The package allows users to type lines of Julia code (e.g. for loops or variable definitions) aside from ODEs if required. Additionally, users can define the desired initial value problem (IVP) solver ("solver"), the absolute and relative tolerances ("tols") and the analytical steady-state equations for the model ("Y0eqs"), if desired. BOMBs.jl will use the analytical steady-state equations to recompute the initial state variables (y_0) for any simulation dependent on the parameter vector considered. For cases where the analytical steady-state depends on values from experimental data, users can perform overnight simulations to generate y_0 by setting the field "Y0Sim" to true.

Once users fill all dictionary fields, the function GenerateModel(model_def) (where model_def is the Julia dictionary containing the model information) introduces the information into a code template, generating the script containing all the necessary functions to simulate models in Julia (Fig. 8.2). The generated Julia script contains four main functions: 1) the function containing the set of ODEs, 2) the function used to compute y_0 (empty function in absence of equations), 3) the function implementing the event-based method (Section 3.2.1.4, p. 73 of Chapter 3) to simulate systems with external inducers and 4) the function used to iteratively simulate a model if the user provides multiple parameter vectors. BOMBs.jl will use all these functions to simulate the system, yet, users can call them outside the package.

8.1.2 Simulations and Pseudo-Data Generation

BOMBs.jl eases the simulation of models under different experimental conditions (e.g. physico-chemical stimuli or maximum experimental time) and automates the generation of pseudo-data for *in silico* studies. For more details of the procedure, see Section 3.2.1.4 (p. 73) of Chapter 3. Users can retrieve the main dictionary structure to simulate models by calling the function defSimulStruct(). Next, users need to fill the dictionary with information about the experiment to simulate (Fig. 8.3a). The dictionary entries refer to the number of experiments ("Nexp"), experiment length ("finalTime"), desired sampling times ("tsamps"), inducer values ("uInd") and time switches for these ("switchT"), a guess for the

a)

```
simul_def = defSimulStruct( );
simul_def["Nexp"] = [1];
simul_def["finalTime"] = [1439];
simul_def["switchT"] = [[0, 500, 1000, 1439]];
simul_def["y0"] = [[1,0,30,4000]];
simul_def["preInd"] = [[1, 1e-7]];
simul_def["uInd"] = [[0.5 0.1 0.75; 100 50 1e-7]];
simul_def["theta"] = ["fit_ALL_Model3.stan.csv"];
simul_def["tsamps"] = [collect(0:10:1439)];
simul_def["plot"] = [true];
simul_def["flag"] = ["Example1"];

simuls, model_def, simul_def =
            simulateODEs(model_def, simul_def);
```

b)

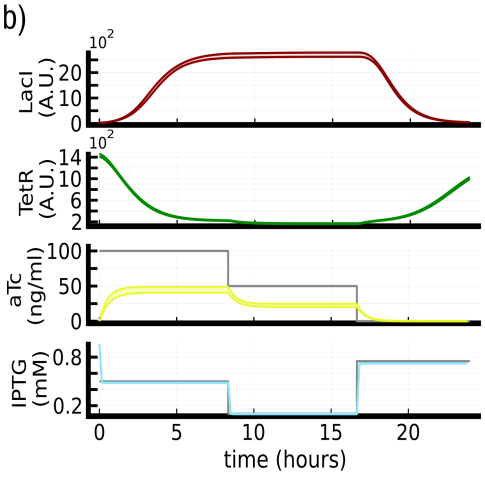


Figure 8.3: BOMBs.jl streamlines reiterative model simulations with minimum user input. a) Code necessary to simulate model M_3 for the genetic toggle switch for a three input step, 24 hour experiment. b) Confidence intervals (99.5th and 0.5th percentiles) for all the state variables of the toggle switch model M_3 for 8000 model simulations using the posterior parameter samples from Section 4.1.3 (multi-experimental inference). Gray lines in the inducer states (aTc and IPTG) represent the administered inputs over time.

a)

```
pseudo_def = defPseudoDatStruct( );
pseudo_def["Nexp"] = [1];
pseudo_def["finalTime"] = [1439];
pseudo_def["switchT"] = [[0, 500, 1000, 1439]];
pseudo_def["y0"] = [[1,0,30,4000]];
pseudo_def["preInd"] = [[1, 1e-7]];
pseudo_def["uInd"] = [[0.5 0.1 0.75; 100 50 1e-7]];
pseudo_def["theta"] =
      mean(Matrix(CSV.read("fit_ALL_Model3.stan.csv")), dims=1);
pseudo_def["tsamps"] = [collect(0:10:1439)];
pseudo_def["plot"] = [true];
pseudo_def["flag"] = ["Example1"];

pseudo_def["Obs"] = ["RFP_LacI", "GFP_TetR"];
pseudo_def["NoiseType"] = ["hetero"];
pseudo_def["Noise"] = [0.1, 0.2];

pseudo_res, model_def, pseudo_def =
            GenPseudoDat(model_def, pseudo_def);
```

b)

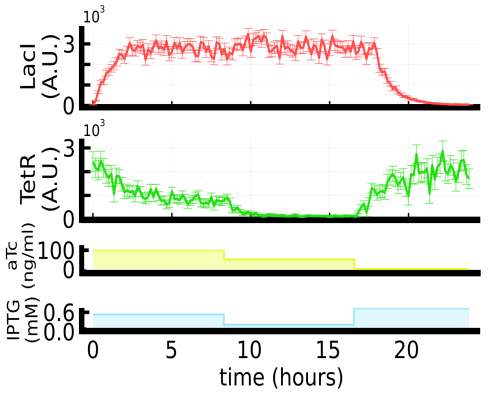


Figure 8.4: BOMBs.jl expands on model simulation automation to generate pseudo-data traces for *in silico* studies. a) Code necessary to generate pseudo-data using the toggle switch model M_3 for a three input step, 24 hour experiment. b) Pseudo-data traces for the observables of the system (RFP-LacI, GFP-TetR) using heteroscedastic Gaussian additive noise. The parameter vector used for the simulation was the mean of the posterior parameter samples from Section 4.1.3 (multi-experimental inference). Bottom panels show the concentration of the inducers over time.

system's initial conditions ("y0") and inducer values for overnight growth ("preInd"). Users can define a vector of parameters, a matrix of parameters (multiple parameter vector instances) or the path to a CSV file containing any of the two ("theta"). Additionally, users can introduce a string that will be attached to all generated files ("flag") and define if they want plots for the simulation results ("plots").

Once users fill all dictionary fields, the function simulateODEs(model_def, simul_def) (where simul_def is the Julia dictionary containing the experimental scheme information) simulates the system of ODEs for each parameter vector introduced and for each experiment defined using the automatically generated functions previously mentioned (Fig. 8.3b). The package saves all the results in .jld files and the simulation plots in .png files (if the dictionary entry "plots" is set to true).

For computational studies, users can follow a similar procedure to generate pseudo-data with additive Gaussian noise. Users can retrieve the main dictionary structure to generate pseudo-data by calling the function defPseudoDatStruct(). The dictionary has the same structure as the one used to simulate models with three additional fields to define 1) the state observables ("Obs"), 2) the use of heteroscedastic or homoscedastic noise ("NoiseType") and 3) the noise proportion (heteroscedastic) or noise error (homoscedastic) to use for the additive Gaussian noise ("Noise") (Fig. 8.4a).

Once users fill all dictionary fields, the function GenPseudoDat(model_def, pseudo_def) (where pseudo_def is the Julia dictionary containing the experimental scheme information to generate the pseudo-data) simulates the system of ODEs and uses the results to generate pseudo-data traces (Fig. 8.4b). Note that if the user introduces multiple parameter vectors in the form of a matrix, the package will generate a pseudo-data trace for each parameter vector. BOMBs.jl automatically saves all the results in .jld files, the pseudo-data plots in .png files (if the dictionary entry "plots" is set to true) and generates .csv files containing the pseudo-data results (observables) and the experimental scheme information (event inputs).

a)

```
mle_def = defMLEStruct( );
mle_def["Nexp"] = [1];
mle_def["finalTime"] = [1439];
mle_def["switchT"] = [[0, 500, 1000, 1439]];
mle_def["y0"] = [[1,0,30,4000]];
mle_def["preInd"] = [[1, 1e-7]];
mle_def["uInd"] = [[0.5 0.1 0.75; 100 50 1e-7]];
mle_def["tsamps"] = [collect(0:10:1439)];
mle_def["plot"] = [true];
mle_def["flag"] = ["Example1"];
mle_def["thetaMAX"] = [0.3366,112.2018,336.6055,
            112.2018,1.7783,11.2202,336.6055,1.1220];
mle_def["thetaMIN"] = [0.0027,0.8913,2.6738,0.8913,
            0.0056,0.0891,2.6738,0.0089];
mle_def["runs"] = [1];
mle_def["parallel"] = [false];
mle_def["DataMean"] = ["TSM3_EXP1_Example1_Observables.csv"];
mle_def["DataError"] = mle_def["DataMean"];
mle_def["Obs"] = ["RFP_LacI", "GFP_TetR"];
mle_def["OPTsolver"] = [ ];
mle_def["MaxTime"] = [120];
mle_def["MaxFuncEvals"] = [ ];

mle_res, model_defSmall, mle_def =
            MLEtheta(model_defSmall, mle_def);
```

b)

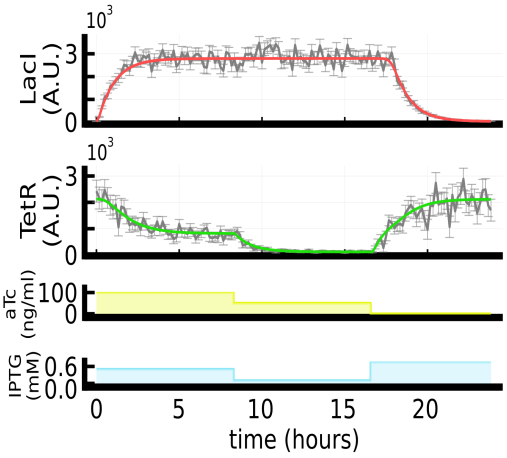


Figure 8.5: BOMBs.jl eases and automates fast parameter estimation for biomodels using maximum likelihood estimation (MLE). a) Code necessary to perform MLE for the toggle switch model M_3 using pseudo-data for a three input step, 24 hour experiment. b) Pseudo-data (gray, top panels), simulations using the best parameter estimate (red and green, top panels) and inducer concentration over time (yellow and blue, bottom panels).

8.1.3 Maximum Likelihood Estimation

BOMBs.jl allows easy and fast parameter estimation of mathematical models using maximum likelihood estimation (MLE). The package uses the univariate or multivariate Gaussian formulation for the log-likelihood to perform MLE, solving the non-linear optimisation problem by using the implemented meta-heuristic algorithms from the Julia package BlackBoxOptim.jl (see Section 3.2.2.1 p. 76 of Chapter 3 for more details). Users can retrieve the main dictionary structure to perform MLE by calling the function defMLEStruct(). Then, users need to fill the dictionary entries with the experimental design details, as done when simulating models (same dictionary keys) (Fig. 8.5a). However, users need to specify the parameter bounds ("thetaMIN" and "thetaMAX") (instead of a parameter vector) and the system observables ("Obs") when performing MLE. Users can introduce experimental data as arrays or as the path to .csv files from which the package will automatically extract the data ("DataMean" and "DataError"). Moreover, users can define which global optimiser to use from the package BlackBoxOptim.jl ("OPTsolver"), a maximum number of function

evaluations ("MaxFuncEvals"), a maximum optimisation time in seconds ("MaxTime") and the number of MLE runs to perform ("runs"). If users select multiple MLE runs, the package can generate the scripts to run them in parallel ("parallel" set to true), indicating how to run the scripts from the command terminal.

Once users fill all dictionary fields, the function MLEtheta(model_def, mle_def) (where mle_def is the Julia dictionary containing the MLE information) runs the optimisation. The package automatically saves all the results in .jld files and the simulation plots in .png files (if the dictionary entry "plots" is set to true). Fig. 8.5 shows the user input (a) and package output (b) for a case example, using model M_3 for the genetic toggle switch. While we can use the method estimating all the model parameters (Section 4.3.1, p. 131) of Chapter 4, for the sake of example simplicity and time reduction, I fixed the parameters k_{IPTG}, k_{aTc}, η_{aTc}, η_{TetR}, η_{IPTG}, η_{LacI} (randomly selected) to the values used to generate the pseudo-data described above. The best parameter vector resultant from the optimisation achieved an average normalised distance to the parameter used to generate the pseudo-data (d_{PSDM}) of 0.102.

Additionally, BOMBs.jl supports the log-likelihood computation for additional experiments to perform cross-validation. Users need to provide a similar dictionary structure (defCrossValMLEStruct()) without the optimisation details and the estimated parameter vectors ("theta") to the function CrossValMLE(model_def, cvmle_def) (where cvmle_def is the Julia dictionary containing the cross-validation information).

8.1.4 Bayesian Inference

BOMBs.jl eases Bayesian inference of biomodels for inexperienced users using the state-of-the-art statistical environment Stan [83]. Section 3.2.2.2 (p. 77) of Chapter 3 contains more details on Bayesian inference. Users can retrieve the main dictionary structure to perform Bayesian inference with Stan by calling the function defBayInfStruct(). To introduce the experimental data for inference, users can provide a dictionary structure in the key "Data" with the same structure as the previous sections. Alternatively, users can provide the path to .csv files containing the experimental data (Fig. 8.6a). The users need to specify the basic Stan necessary settings ("StanSettings"): path to the Stan installation

a)

```
bayinf_def = defBayInfStruct( );
bayinf_def["Data"] = defBayInfDataFromFilesStruct( );
bayinf_def["StanSettings"] = defBasicStanSettingsStruct( );
bayinf_def["flag"] = ["test"];
bayinf_def["plot"] = [true];
bayinf_def["runInf"] = [true];
bayinf_def["MultiNormFit"] = [true];
bayinf_def["Data"]["Observables"] =
        ["TSM3_EXP1_Example1_Observables.csv"];
bayinf_def["Data"]["Inputs"] =
        ["TSM3_EXP1_Example1_Events_Inputs.csv"];
bayinf_def["Data"]["Obs"] = ["RFP_LacI", "GFP_TetR"];
bayinf_def["Data"]["y0"] = [[1,1e-7,30,4000]];
bayinf_def["Priors"] = [11.2202 11.2202 11.2202;
0.8913 0.0891 0.0891];
bayinf_def["StanSettings"]["cmdstan_home"] =
        ["C:/Users/David/.cmdstanpy/cmdstan-2.20.0"];
bayinf_def["StanSettings"]["nchains"] = [4];
bayinf_def["StanSettings"]["nsamples"] = [1000];
bayinf_def["StanSettings"]["nwarmup"] = [500];
bayinf_def["StanSettings"]["printsummary"] = [true];
bayinf_def["StanSettings"]["maxdepth"] = [13];
bayinf_def["StanSettings"]["adaptdelta"] = [0.95];
bayinf_def["StanSettings"]["jitter"] = [0.5];
bayinf_def["StanSettings"]["init"] = [ ];

stan_res, model_defSmall2, bayinf_def =
        StanInfer(model_defSmall2, bayinf_def);
```

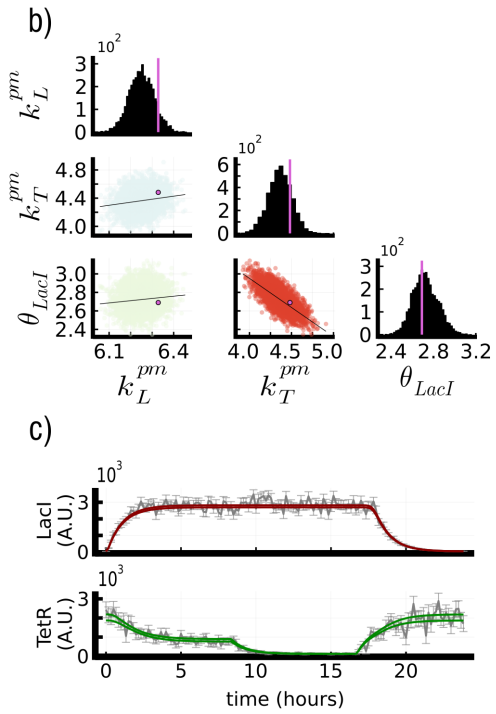


Figure 8.6: BOMBs.jl eases and automates Bayesian inference of models using the statistical programming language Stan. a) Code necessary to perform Bayesian inference using Stan for the toggle switch model M_3 using pseudo-data for a three input step, 24 hour experiment. b) Histograms (diagonal) and bi-variate scatter plots for the inferred parameter posterior using Stan. Purple lines (histograms) and dots (scatter plots) represent the parameter values used to generate the pseudo-data. c) Pseudo-data (gray) and confidence intervals (99.5th and 0.5th percentiles) of 4000 simulations using the inferred parameter posterior (red and green).

directory ("cmdstan_home"), number of chains for the MCMC chains ("nchains") and number of warm-up ("nwarmup") and post warm-up samples ("nsamples"). Users can define additional Stan settings such as "printsummary" (print summary inference file), "maxdepth" (ramification depth of the balanced binary tree), "adaptdelta" (target average proposal acceptance probability), "jitter" (step size jittering) and "init" (initial guess for the sampler).

Users can provide a string with priors defined in Stan language, parameter samples, parameter bounds or nothing to define prior distributions ("Priors"). If the users provide priors defined in Stan language, these will get indented into the Stan file generated. If the user provides parameter samples, BOMBs.jl will fit these using the best distributions (measured using log-likelihood) supported in

Stan (or a multivariate Gaussian distribution if the dictionary key "MultiNormFit" is set to true) and translate the results into the Stan programming language. If the user provides parameter bounds, BOMBs.jl will generate uninformative normal distributions truncated at $\pm 2\sigma$, with $\mu = (max + min)/2$ and $\sigma = (max - min)/4$. If the user provides an empty entry in the dictionary, BOMBs.jl will still generate the main Stan script and data structure lacking the priors section and without running inference. Consequently, if users want to generate all the inference scripts and data structures without running the inference, the dictionary key "runInf" should be set to false.

Once users fill all dictionary fields, the function StanInfer(model_def, bayinf_def) (where bayinf_def is the Julia dictionary containing the Stan Bayesian inference information) runs the inference. The package automatically saves all the results in .jld files, the posterior distribution samples as a .csv file and inference statistics and simulation plots in .png files (if the dictionary entry "plots" is set to true). Fig. 8.6 shows the user input (a) and package output (b,c) for a case example, using model M_3 for the genetic toggle switch. While we can use the method estimating all the model parameters (Section 4.1.3, 117 of Chapter 4), for the sake of example simplicity and time reduction, I fixed all parameters except k_L^{pm}, k_T^{pm} and θ_{LacI} (randomly selected) to the values used to generate the pseudo-data described above. Fig. 8.6b shows the obtained 4000 posterior samples (and correlations) for the three inferred parameters compared to the value used to generate the pseudo-data. Fig. 8.6c shows the model predictions using the 4000 posterior samples when predicting the experiment used for inference. Additionally, BOMBs.jl can fit posterior samples with Gaussian mixture models and approximate posterior entropies as described in Section 3.2.3 (p. 81) of Chapter 3 [334] by calling the function computeH(samples, model_def).

BOMBs.jl also eases Bayesian inference of biomodels in a Julia environment by generating all the necessary scripts for the state-of-the-art statistical environment Turing.jl. Users can retrieve the main dictionary structure to perform Bayesian inference with Turing.jl by calling the function defTurInfStruct(). The dictionary structure is the same as the one used for Stan inferences, but without the settings section since BOMBs.jl only generates the code to perform inference, guiding users on how to execute the scripts from the command terminal (Fig. 8.7a). To produce the Turing.jl model and adjacent working functions, users need to run the function TuringInfer(model_def, turinf_def) (where turinf_def is the Julia

a)

```
turinf_def = defTurInfStruct( );
turinf_def["Data"] = defturinfDataFromFilesStruct( );
turinf_def["flag"] = ["test"];
turinf_def["plot"] = [true];
turinf_def["MultiNormFit"] = [true];
turinf_def["Data"]["Observables"] =
            ["TSM3_EXP1_Example1_Observables.csv"];
turinf_def["Data"]["Inputs"] =
            ["TSM3_EXP1_Example1_Events_Inputs.csv"];
turinf_def["Data"]["Obs"] = ["RFP_LacI", "GFP_TetR"];
turinf_def["Data"]["y0"] = [[1,1e-7,30,4000]];
turinf_def["Priors"] = [11.2202 11.2202 11.2202;
0.8913 0.0891 0.0891];

turing_res, model_defSmall2, turinf_def =
            TuringInfer(model_defSmall2, turinf_def);

include("ModelsFunctions\\TSM4_TuringModel.jl");

model = TSM4_TuringModel(turing_res["turinf_def"]["Data"],
            turing_res["probExamp"])

chainss = sample(model, NUTS(500, .65),
            MCMCThreads( ), 1500, 4, progress=true)
```

b)

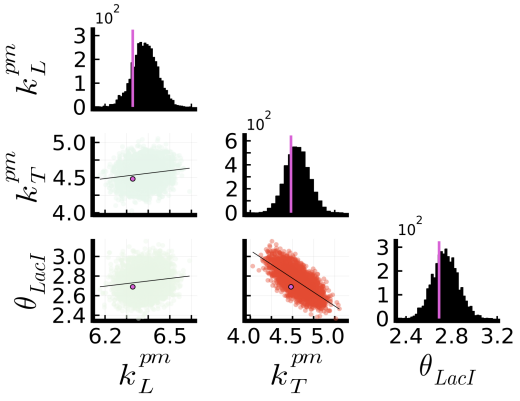


Figure 8.7: BOMBs.jl eases Bayesian inference of models using the Julia package Turing.jl. a) Code necessary to generate the Turing.jl model script and to perform Bayesian inference of models. b) Histograms (diagonal) and bi-variate scatter plots for the inferred parameter posterior using Turing.jl. Purple lines (histograms) and dots (scatter plots) represent the parameter values used to generate the pseudo-data.

dictionary containing the Stan Bayesian inference information). BOMBs.jl will then print in the command terminal how to proceed to perform Bayesian inference. First, users will need to include the generated Turing.jl model into the working path and spawn the model with the data structure. Then, users can run the sampling using any of the inference methods available in Turing.jl. Fig. 8.7 shows the user input (a) and inference output (b) for the same case example considered with Stan.

8.1.5 Optimal Experimental Design

BOMBs.jl automates the generation of optimal experiments for model selection and identification. The package uses the methods described in Sections 5.1 (p. 141) of Chapter 5 and 7.1 (p. 186) of Chapter 7 and described in more detail in Section 3.2.7 (p. 87) of Chapter 3, to quantify the informative content of experiments concerning model selection or inference, respectively. BOMBs.jl leverages the Bayesian optimisation algorithm implemented in the Julia package BayesianOptimisation.jl to account for the potentially high computational cost of

a)

```
oedms_def = defODEModelSelectStruct( );
oedms_def["Model_1"] = [model_defPre];
oedms_def["Model_2"] = [model_def];
oedms_def["Obs"] = ["RFP_LacI", "GFP_TetR"];
oedms_def["Theta_M1"] = "fit_ALL_Model1.stan.csv";
oedms_def["Theta_M2"] = "fit_ALL_Model3.stan.csv";
oedms_def["y0_M1"] = [1,1e-7,30,4000];
oedms_def["y0_M2"] = [1,1e-7,30,4000];
oedms_def["preInd_M1"] = [1,1e-7];
oedms_def["preInd_M2"] = [1,1e-7];
oedms_def["finalTime"] = [1439];
oedms_def["switchT"] = [[0, 1439]];
oedms_def["tsamps"] = [collect(0:10:1439)];
oedms_def["fixedInp"] = [ ];
oedms_def["fixedStep"] = [ ];
oedms_def["equalStep"] = [ ];
oedms_def["plot"] = [true];
oedms_def["flag"] = ["testOEDMS1"];
oedms_def["uUpper"] = [1, 100];
oedms_def["uLower"] = [1e-7, 1e-7];
oedms_def["maxiter"] = [100];

oedms_def = genOptimMSFuncts(oedms_def);
oedms_res, oedms_def = mainOEDMS(oedms_def);
```

b)

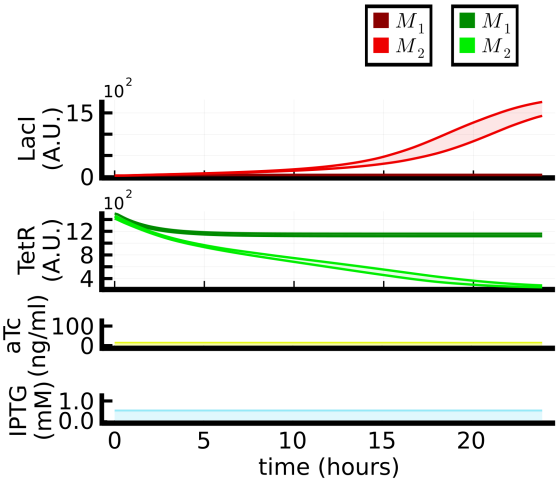


Figure 8.8: BOMBs.jl integrates automated optimal experimental design for model selection. a) Code necessary to design experimental schemes to optimally discriminate between pairs of competing models. b) Posterior predictions (99.5th and 0.5th percentiles) for the toggle switch models M_1 and M_3 simulating a one step optimal experiment for model selection designed by the package. To simulate the models, I used the posterior parameter samples from Section 4.1.3 (multi-experimental inference).

these strategies.

Users can retrieve the main dictionary structure to perform OED for model selection by calling the function defOEDModelSelectStruc(). Users need to fill the dictionary structure (Fig. 8.8a) with the information about models ("Model_1" and "Model_2", equivalent to the model_def previously defined), the system observables ("Obs"), the parameter samples ("Theta_1", "Theta_2") and the system's initial state guess ("y0_M1", "y0_M2"). Then users need to specify the experiment scheme details by providing information about the concentration of external inducers in overnights if applied ("preInd_M1", "preInd_M2"), experiment duration ("finalTime"), inducer change time points ("switch_T") and sampling time points ("tsamps"). For the optimisations, users must define the upper ("uUpper") and lower ("uLower") bounds for the external inducers, and the maximum number of iterations ("maxiter") or computational time allowed ("maxtime") for the optimiser.

a)

```
oedmc_def = defODEModelCalibrStruct( );
oedmc_def["Model"] = [model_def];
oedmc_def["Obs"] = ["RFP_LacI", "GFP_TetR"];
oedmc_def["Theta"] = ["TS_Ran1_Posterior2.csv"];
oedmc_def["y0"] = [1,1e-7,30,4000];
oedmc_def["preInd"] = [1,1e-7];
oedmc_def["finalTime"] = [1439];
oedmc_def["switchT"] = [[0, 1439]];
oedmc_def["tsamps"] = [collect(0:10:1439)];
oedmc_def["fixedInp"] = [ ];
oedmc_def["fixedStep"] = [ ];
oedmc_def["equalStep"] = [ ];
oedmc_def["plot"] = [true];
oedmc_def["flag"] = ["testOEDMC"];
oedmc_def["uUpper"] = [1, 100];
oedmc_def["uLower"] = [1e-7, 1e-7];
oedmc_def["maxiter"] = [150];
oedmc_def["util"] = [ ];

oedmc_def = genOptimMCFuncts(oedmc_def);
oedmc_res, oedmc_def = mainOEDMC(oedmc_def);
```

b)

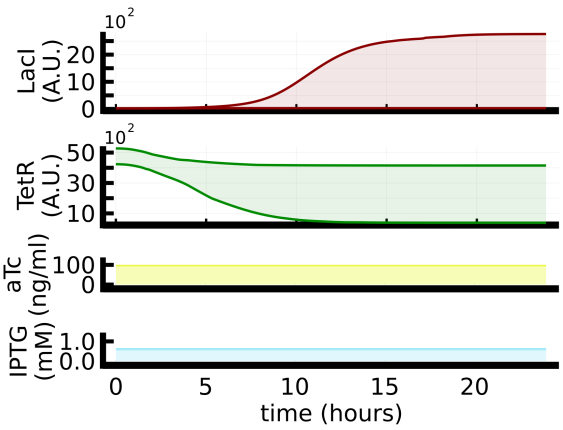


Figure 8.9: BOMBs.jl integrates automated optimal experimental design for parameter inference. a) Code necessary to design experimental schemes to optimally infer model parameters. Posterior predictions (99.5th and 0.5th percentiles) for the toggle switch model M_3 simulating a one step optimal experiment for parameter inference designed by the package. To simulate the models, I used the posterior parameter samples inferred from the initial random experiment in Section 7.2.

Additionally, the current BOMBs.jl implementation allows three types of optimisation constraints. First, in the case where a system has multiple external inducers, n-1 inducers (with n being the total number of inducers) can be optimised for a constant value whilst the rest of the inducers will be optimised following the inducer switching times ("fixedInp"). Second, the user can impose a specific value for a given experiment step for any of the inducers ("fixedStep"). Third, the user can constraint some experiment steps to have the same optimised inducer value ("equalStep").

Users can retrieve the main dictionary structure to perform OED for parameter inference by calling the function defOEDModelCalibrStruc(). The dictionary has the same structure as the one defined for model selection but refers to just one model structure (Fig. 8.9a). The only addition to the dictionary is the option to use percentiles or entropy when quantifying the prediction uncertainty of models ("util").

Before the package starts optimisation, users need to generate all the necessary

optimisation files by running the function genOptimMSFuncts(oedms_def) for model selection or genOptimMCFuncts(oedmc_def) for parameter inference (where oedms and oedmc are the Julia dictionaries containing the optimisation information for model selection and calibration, respectively). Then users can start the OEDs by running the function mainOEDMS(oedms_def) for model selection or mainOEDMC(oedmc_def) for parameter inference. The package automatically saves all the results in .jld files and simulation plots in .png files (if the dictionary entry "plots" is set to true). Fig. 8.8b shows the predictions of the genetic toggle switch models M_1 and M_3 for an OED for model selection. I optimised aTc and IPTG in a single step experiment to drive competing models to opposite steady-states. Fig. 8.9b shows the prediction of the toggle switch model M_3 for an OED for parameter inference. I optimised the inducers in a single step experiment to maximise the prediction uncertainty of the model.

8.2 Chapter Conclusions

There is an immediate need for easy-to-use computational tools allowing inexperienced users in life sciences to integrate models into their experimental routines. Such tools contribute to the understanding and predictive control and (re)design of biological systems. In this chapter, I described BOMBs.jl, the first low-code Julia package, to ease model simulations, Bayesian inference of parameters and optimal experimental design for model selection and calibration. I showed that, by using predefined code templates, the package automates the generation of scripts needed to simulate mathematical models with time-varying parameters (e.g. physico-chemical stimuli). While the current implementation of BOMBs.jl only allows models defined as ODEs, the code is readily extendable to other types of differential equations (e.g. stochastic or delayed differential equations) since it is based on the Julia package DifferentialEquations.jl. I also showed how to use BOMBs.jl to simulate systems and generate pseudo-data from these, an easy process that only requires basic information about the experimental scheme(s). The current package implementation only supports additive Gaussian heteroscedastic or homoscedastic noise. However, the code is readily extendable to add other types of noise (e.g. exponential).

I then showed how BOMBs.jl highly simplifies for users one of the most critical steps in using mathematical models, the inference of model parameters. Users

can quickly estimate model parameters using the Gaussian formulation for the log-likelihood (i.e. maximum likelihood estimation), where the package leverages state-of-the-art global optimisers (BlackBoxOptim.jl) to solve the non-linear optimisation problem. For more accurate parameter inferences, using Bayesian statistics to relax noise assumptions, I integrated the automated generation of Stan and Turing.jl models. This package section removes the need for computational proficiency in the statistical coding languages from the user to perform Bayesian inference of biomodels. While users can still modify some settings from the MCMC algorithms, the adaptive No-U-Turn Sampler (NUTS), implemented in both Stan and Turing.jl, proves an outstanding option for new users due to its flexibility and capability of tuning its hyper-parameters (e.g. step size or number of steps) [331].

Finally, I showed how BOMBs.jl removes all coding steps to perform optimal experimental design for model selection and inference, leaving the user with the only task of specifying the characteristics of the experiment to design (e.g. inducer bounds and time switches). I framed the non-linear optimisation problems in a Bayesian context. Consequently, I decided to integrate Bayesian Optimisation (BayesianOptimisation.jl) to efficiently explore the design spaces while minimising the number of evaluations. However, the code is readily extendable to integrate different global optimisers in the future. Overall, I designed the package to facilitate the widespread of mathematical models across our community, making it general, easy-to-use and highly automated while incorporating extensive documentation and examples to ease its use for new users.

Chapter 9

Discussion and Conclusions: Limitations and Future Perspectives

Recently, biological sciences are increasingly incorporating quantitative tools and methodologies to understand and (re-)design biological systems. Additionally, the recent technological advancements provide us with better interrogation platforms to study these systems. Mathematical modelling is a powerful tool to extract as much information as possible from this new data, better understand the different biological processes or predict the behaviour of engineered systems. We can already see the positive impact of technology and mathematical modelling in emerging disciplines such as systems and synthetic biology or bioinformatics. Yet, despite its multiple advantages and potential, there is still a limited spread of mathematical modelling across biology. As compelling as they might be, models still present a series of difficulties that limit their use amongst inexperienced users. While there is extensive literature on the formulation of such models (e.g. following mass action kinetics), lack of knowledge about a specific biological system might result in the formulation of different candidates. At this point, researchers are left with the arduous problem of model selection. An even more recurrent issue is the identification of model parameters. It is usually impractical (if even possible) to experimentally measure kinetic rates and other parameters. So researchers need to indirectly infer them from experimental data, a challenging task with multiple approaches. While there is a consensus that probabilistic methods (i.e. Bayesian inference) yield better results, these are complex to implement and extremely time-consuming (e.g. some inferences in this work took up to two weeks to finish). Such limitations result in the selection of easier or

faster methodologies that might incorporate unreasonable assumptions and noise overestimations. Furthermore, solving such core issues to start using mathematical models requires experimental data. This data is usually expensive, time-consuming to generate and inherently noisy. The problem is that different experiments contain different amounts of information. Researchers need to find a way to design highly informative experiments to minimise time and resources and still get the desired results; again a complex task. While all these problems have been tackled for years in other disciplines, biological sciences still have limited expertise, with limited widespread tools to solve them. Consequently, all this hampers the use of mathematical modelling amongst less experienced users, ultimately been counterproductive for the progress of the field.

In this work, I addressed the previously stated limitations to help overcome some of the main steps that hinder the spread of mathematical modelling in biological sciences. First, I evaluated the use of Bayesian statistics to infer parameters of mathematical models describing biological systems. Due to their increasing interest, especially in synthetic biology, I also investigated the inference of parameters from models describing biological systems with time-varying external inducers. I report this results in Section 4.1, (p. 112) of Chapter 4. I then investigated methodologies to identify the informative content of experiments *a posteriori* using metrics to assess the distance of a posterior concerning a known parameter value in computational studies (i.e. d_{PSD} and d_{PSDM}) and information theoretic metrics (i.e. $H(\theta) - H(\theta|D)$ and $-log(|\Sigma|)$). I show this results in Section 4.2, (p. 124) of Chapter 4. Since it is highly advantageous to estimate the informative content of experiments *a priori*, I investigated the use of model prediction uncertainty for the task in Bayesian scenarios. I included this study in Section 4.3, (p. 131) of Chapter 4. Having established these core methodologies, I investigated the traditional approach to model selection and proposed an OED strategy to design highly informative experiments for the task. This results are explained in Section 5.1, (p. 141) of Chapter 5. I then further investigated the results obtained by observing how the methodology leveraged intrinsic properties of the system under study (i.e. bistability of a genetic toggle switch developed in *E. coli*) to generate experiments that drive two competing models to show opposite behaviours. I show this study in Section 5.2, (p. 151) of Chapter 5. I then showed my implementations for an efficient image processing platform to process microscopy images in real time and a platform to execute on-line optimal experiments using our microfluidics setup. I report this results in Sections 6.1 (p.

158) and 6.2 (p. 170) of Chapter 6, respectively. Then, I investigated the potential use of using model prediction uncertainty to develop a Bayesian OED strategy for model inference due to the promising results obtained when I used it to assess the information content of experiments *a priori*. I successfully tested the methodology, both *in silico* and *in vivo* in Sections 7.1 (p. 186) and 7.2 (p. 206) of Chapter 7, respectively. Finally, I gathered most of the mentioned computational methodologies into an open-source, easy-to-use Julia package called BOMBs.jl. I describe the package in depth in Chapter 8.

I successfully showed that we could use Bayesian statistics to efficiently infer model parameters using the open-source probabilistic language Stan in Section 4.1, (p. 112) of Chapter 4. While been generally regarded as the best approach to infer model parameters, its traditional complexity hindered its use for decades. I showed that we could use the strategy to infer parameters of models describing biological systems as previously done in the literature. Consequently, I expanded the work to also include systems with time-varying external inputs, a type of Bayesian inference setting unexploited in systems and synthetic biology. I then highlighted the importance of experimental selection when estimating model parameters and how parameter posteriors tend to deviate from normality. I also underlined the importance of the predictive capabilities of the inferred models and the uncertainty of these predictions. Yet, I acknowledge that Bayesian approaches still present limitations in more complex scenarios with bigger models with a higher number of degrees of freedom. These limitations come from the intractability of analytical solutions for the Bayes theorem in this scenarios where linearity and Gaussianity can rarely be applied and the current need for sampling algorithms to circumvent the problem. However, thanks to technological advancements and the development of efficient sampling algorithms (e.g. MCMC), we are increasing the level of complexity of the models we can infer. More focus should be directed toward designing better sampling algorithms (e.g. NUTS) and open-source easy-to-use platforms that implement these (e.g. Stan). Furthermore, I focused my work on just one type of mathematical model used to describe biological systems, ordinary differential equations. Further work should focus on assessing these methodologies and tools on other types of models commonly used in biology, such as partial or stochastic differential equations.

I investigated the *a posteriori* assessment of informative content of experiments directed at parameter inference in Section 4.2, (p. 124) of Chapter 4. First, I

showed that different experimental conditions provide varying degrees of information using models describing four biological systems. Using computational examples, I compared a commonly used metric, the logarithm of the determinant of the covariance matrix (related to the FIM), to a normalised posterior sample distance to known parameter values used to generate the pseudo-data. I showed a high correlation between the two metrics accentuating the usefulness of the $log(|\Sigma|)$ to assess the gain in information after inference. However, since, in real cases, the parameter values are unknown (otherwise, this would delete the need for estimating them), the sample distance metric is only practical in *in silico* studies. Nonetheless, the metric poses a useful tool for preliminary analyses. The metric also highlights that reduction of posterior uncertainty is only part of the goal and that a good validation set of experiments is highly convenient to ensure that these posteriors are in the correct region of the parameter space. Since inferred parameter posteriors rarely follow normality, I introduced an information theoretic approach by approximating posteriors using Gaussian mixture models and estimating their entropy. I then defined the gain in information as the difference between prior and posterior entropy. While I found a high correlation between this metric and the $log(|\Sigma|)$ (assumption of posterior normality), the differences between the two highlight the potential errors introduced when assuming posterior normality. Note that this is a cumulative error, and hence, will have a stronger effect as the number of parameters to infer increases. On top of these, I used an example (the genetic toggle switch) to show the positive correlation between a higher number of inducer changes during an experiment and the informative content of these for parameter estimation. These larger amount of inducer changes better explored variations in the system dynamics and provided more information about the parameters. I acknowledge that the posterior entropy computation I implement is an approximation based on Gaussian mixture models. Further work should focus on better and faster approximations of such entropy for multi-dimensional non-normal posterior distributions.

Model prediction uncertainty is an important element of the inference processes, usually overlooked in frequentist approaches to the problem. I inferred three mathematical models for the genetic toggle switch using both, frequentist and Bayesian strategies in Section 4.3.1, (p. 131) of Chapter 4. While in Bayesian inferences one recovers the posterior uncertainty (or an approximation) for the model parameters, in the frequentist case, I followed a common approach: using

the FIM to approximate the covariance matrix. This procedure allowed me to draw a parallelism between both strategies to compare the advantages and disadvantages of the two. I observed that while both methodologies used the same experimental data, a FIM-based approach highly overestimated parameter uncertainty, highlighting the advantages of a Bayesian approach to better estimate parameter variances and covariances. This overestimation also means that if we perform Bayesian inference using uninformative priors (i.e. uniform priors), this would still be, in principle, more accurate at determining parameter uncertainties. Next, I observed that, in all cases, parameter uncertainty propagated into model predictions. I evaluated the possibility of using this relationship to efficiently assess *a priori* the informative content of experiments in a Bayesian framework in Section 4.3.2, (p. 134) of Chapter 4. Other Bayesian approaches for *a priori* identification of the informative content of experiments for model inference are usually exceedingly computationally expensive if even tractable (e.g. using mutual information). Here I present a computationally efficient methodology for the task, validating it using previously published results and finding a high correlation between prediction uncertainty and *a posteriori* experiment informative content evaluations. Hence, I used prior information to identify the experimental designs that would, in principle, result in the highest reduction of its uncertainty. Despite its potential, the method is a sampling-based one. Hence, the computational complexity of the task becomes directly correlated to model complexity and the number of parameter samples used for approximations. A previous evaluation of the number of parameter samples to use should be done case by case in the presence of computational or time constraints. I also would deem it advantageous to thoroughly compare it other with well-established methods (e.g. FIM-based) to evaluate the pros and cons of using this approach. Quantifying model prediction uncertainty using percentiles is a well-established method. However, the entropy-based strategy I additionally proposed could still be improved. This methodology would benefit from including more distributions for the fit than just the ones selected in this work. Yet, I note that the distribution functions selected (and other potential ones) have different numbers of parameters, which can result in overfitting since I use the log-likelihood to choose the best one. While I was interested in the shape of the distribution, further improvement of this strategy could be achieved by introducing a complexity penalisation term as in the AIC or BIC metrics (see Section 3.2.4, p. 83 of Chapter 3). Alternatively, a Gaussian mixture model could be used to fit the distributions. Nonetheless, such an approach could slow down

the process if an optimisation scheme should be needed to select the optimal number of components.

Model selection is an important, yet, a usually shunned task due to its complexity. I first investigated traditional approaches to the problem using three models for the genetic toggle switch as an example in Section 5.1.1, (p. 141) of Chapter 5). These approaches include metrics such as model evidence (Bayesian frameworks), the AIC (frequentist frameworks) and others (BIC, nRMSE). As previously observed in the literature, model selection depends on the experimental design and the metric used for the assessment. Furthermore, in the Bayesian scenario, the model evidence used to compare models was sensitive to prior distributions. Yet, in the model selection problem, we want to evaluate models using posteriors, so the metric would be better suited to compare and test prior hypotheses. The metric selection was based on its widespread use for the task in literature. While, in this work, I used highly similar models with the same priors for the shared parameters, this prior dependence can become problematic in other studies focused on the model selection problem with highly dissimilar models or prior distributions [355]. Regarding the results, on average, all the metrics pointed to the same best model, but the experimental data that supported this selection depended on the methodology and metric used. Yet, experimental data acquired *a posteriori* showed that the model was unable to recapitulate essential properties of the system. While I attribute most of this effect to the limited data set used to infer parameters, it is worth noting that the "best" model selected using this approach was the one with a higher degree of freedom when describing the inducer transport. Hence, there is a need for better strategies to aid researchers in discriminating between competing models. I proposed an OED strategy to generate experiments that maximise the difference between model predictions, pointing to the optimal data that would deem a model wrong in Section 5.1.2, (p. 145) of Chapter 5. Our strategy can incorporate model prediction uncertainty (Bayesian case) or exclude it (frequentist case), yet both approaches exploit inherent characteristics of the system to design the best experiments. For my example (the genetic toggle switch), the optimisation drove two competing models to opposite steady-states. I then evaluated a simple design experimentally, highlighting that the model selected by the previous metrics was the one achieving the wrong steady-state. Nonetheless, one would better benefit from this methodology by including the new optimal experiments into the inference scheme in a cycle until making one of the competing models unable to recapitulate all the

available data. During my tests, I observed an effect in the dynamic range when I included multiple observables in the computations. I deem it convenient to assess the effect case by case and focus preliminary work on determining a normalisation factor or a procedure to determine observable weights to match the contribution of these in the final cost function. Also, the Bayesian methodology is a sampling-based one making it more time-consuming to compute. One should preliminarily assess the need to include prediction uncertainty in the process or follow the frequentist approach. We hypothesise that this selection would depend on a case by case base. Finally, while my results highlight the potential of our strategy to generate optimal data for model selection, future work should focus on assessing the validity and capability of the method with different biological systems. Also, the methodology would benefit from an *in silico* study showing how using these optimal experiments in a cycle of design and inference results in the convergence to the "true" predefined model while invalidating the wrong one.

Since both optimal design strategies (i.e. frequentist and Bayesian) resulted in the same type of results, driving the system to opposite steady-states, I further investigated the reason behind it in Section 5.2, (p. 151) of Chapter 5. I investigated the appearance of bistable or monostable behaviour (and protein dominance) when applying different concentrations of the inducers to the system. I observed that the stability profiles for competing models presented differences and that our OED strategy leveraged these discrepancies. Such stability analysis is a powerful tool for analysing the properties of inducible systems with potential bistable behaviour or monostable with multiple proteins involved. Nonetheless, the approach only provides a static view of the system without exploring dynamics. Hence, while it could be used to design experiments for model selection, these will be reduced at exploring steady-states. To better assess the goodness of fit of models for the genetic toggle switch or add additional constraints in inference schemes, we would benefit from experimentally measuring all the inducer concentration conditions in a similar way as proposed by Barbier *et al.* 2020 [356].

Due to the potential of using model prediction uncertainty to assess the informative content of experiments for model inference, I coupled this to an optimisation scheme to propose an efficient Bayesian OED strategy in Section 7.1 (p. 186) of Chapter 7. First, I performed an *in silico* study to show the methodology potential, illustrating one of the first efficient Bayesian strategies for

the task. I tested the strategy on a simple study case (see Section 7.1.1, p. 186 of Chapter 7) and increased the complexity of the tests by using the genetic toggle switch while considering different levels of experimental constraints (see Section 7.1.2, p. 191 of Chapter 7). All tests indicated the potential of my strategy, generating experiments that significantly decreased posterior uncertainty. The optimal experiments also made posteriors get closer to the parameter values used to simulate the pseudo-data and better recovered the stability properties of the genetic toggle switch. I also showed that the presence of high experimental constraints can limit the optimisation scheme to select sub-optimal designs. An on-line OED approach can overcome these issues, generating maximally informative experiments in an automated fashion (see Section 7.1.3, p. 198 of Chapter 7). Yet, despite its advantages, the strategy is a sampling-based one. Hence, the complexity of the problem will scale up in more complex scenarios with bigger models. A computational study comparing this methodology with other established ones (e.g. FIM-based) would highlight the advantages and disadvantages of my strategy. It will also point out the cases where using this sampling-based method becomes highly beneficial. I would also deem it convenient to test the methodology using more complex models with a higher number of equations, parameters and observables. At this point, future work should also focus on determining an optimal number of parameter samples to use for the prediction uncertainty approximation. This work would also benefit from optimising the hyper-parameters of the Bayesian optimisation algorithm used or exploring other highly efficient optimisers for the task. As with our OED strategy for model selection, the different dynamic range of observables affects the results. Hence, better normalisation or weight assignment strategies should be investigated. Finally, I tested the methodology following microfluidic setups for the type of data generated. Yet, some work should focus on different common types of data besides time-series (e.g. flow cytometry data) and optimisation of other experimental variables (e.g. sampling time distribution).

After the *in silico* analysis of the Bayesian OED strategy, I evaluated this in an *in vivo* scenario using the genetic toggle switch developed by Lugagne *et. al.* 2017 [85] as a case example in Section 7.2 (p. 206) of Chapter 7. In computational studies, we know the model and parameters used to generate the pseudo-data and we can freely adjust experimental variables. However, in real scenarios, experimental data is hard to obtain, tends to enclose higher noise levels, and laboratory setups are highly constrained. Additionally, we must

assume that the model used is a good representation of the system under study (or perform model selection with multiple model candidates). Assessment of the goodness of fit and prior definition is also more complicated due to the lack of knowledge about model parameters, making the whole process highly intricate. Yet, despite the potential of OED to minimise resources, effort and time allocated to experimentation while obtaining equivalent or better inference results, the vast majority of the work performed in biological sciences is computational. Here, I study OED in an *in vivo* scenario, analysing all the advantages and limitations of my methodology (see Section 7.2.1, p. 207 of Chapter 7). On top of this, I implemented an on-line OED strategy in our microfluidics platform (see Section 7.2.2, p. 217 of Chapter 7). My results show how OED reduces the amount of time and resources allocated to costly and tedious experiments while maintaining or even improving the quality of the results obtained. I also evaluated the stability properties of the inferred models and proposed a process to introduce this information in the form of constraints for an inference scheme (see Section 7.2.3, p. 225 of Chapter 7). My approach using prediction uncertainty to estimate the informative content of experiments also worked in the selection of the constraints to select, further improving the predictive capabilities of models. Nonetheless, future work should assess the potential adverse effect of introducing these constraints in the parameter identification schemes. These could be performed by inferring the mathematical model including other microfluidic experiments, since I only used one in my Bayesian analysis (on-line optimal experiment). All results highlight the advantage of OED while reiterating the importance of adequate validation data to assess the inferred models. Reduction of posterior uncertainty only provides limited information about the goodness of fit. Yet, I deem it crucial to keep testing the methodology in the future using different biological systems *in vivo*. I also acknowledge that other models should be tested to tackle the assumption that the one used was a good representation of the system, potentially combining it with our OED for model selection strategy to select the best model. Additionally, my work would have benefited from an experimental estimation of scaling factors for the model to allow easy transferability of the inferred parameters into different experimental setups. However, the introduction of such scaling factors made the system to be unidentifiable. I also acknowledge that the research would have benefited from a larger validation data set and a larger number of on-line experiments that I was unable to acquire due to experimental constraints. Finally, I consider it important to introduce in future works new microfluidic device designs that

reduce the constraints in the type of experimental designs allowed, especially for systems with multiple external inputs.

To process the data from our microfluidics platform, I developed an image processing platform capable of processing fluorescence microscopy images right after these are acquired in Section 6.1 (p. 158) of Chapter 6. This real-time image processing is a crucial step for process automation (e.g. on-line OED). I generalised the platform (see Section 6.1.3, p. 164 of Chapter 6) to be able to process images from the two main groups of cells we work with, yeast (i.e. *S. cerevisiae*) and bacteria (i.e *E. coli*). Future work should focus on expanding the platform to include other kinds of yeast and bacteria cells or even mammalian ones. Also, the platform works in close connection with the image acquisition one. Upcoming work should centre on generalising it and deleting the dependency on my image processing platform to allow other research groups to benefit from it with minimal modifications. A critical step in image processing is cell segmentation from microscopy images. For this, I used U-Net (see Section 6.1.2, p. 161 of Chapter 6), a convolutional neural network widely adopted in recent years by research groups due to its easy training and use and high segmentation accuracy. The only limiting step of the process was the generation of grown truth annotations for the training, a tedious and time-consuming task. I also investigated an efficient and accurate strategy to calibrate Dial-A-Wave (DAW) junctions (see Section 6.1.1, p. 158 of Chapter 6). I wrote most of the platform components using Matlab as a core language. Future work should focus on translating this into an open-source language (e.g. Python or Julia) for a better spread amongst other potential users.

To perform on-line OED, I built a new platform that worked in parallel with the image processing one in Section 6.2 (p. 170) of Chapter 6. Since we were able to process microscopy images as we were acquiring them, I designed a second platform that checked the processed data. Once the platform had enough data to estimate the model parameters and redesign the next section of the experiment, this started all necessary computations. Hence, I built one of the first on-line OED platforms for microfluidic experiments in frequentist (see Section 6.2.2, p. 173 of Chapter 6) and Bayesian contexts (see Section 6.2.3, p. 177 of Chapter 6). Since the platform works together with the image processing one, it would be easily expandable to use data from other types of setups (e.g. optogenetic ones). Future work should be directed to merge this platform and the image processing

one into the same one for easier use by other research groups. I acknowledge that since the frequentist platform is written in Matlab, it has limited spread for users without a licence. Hence, work should focus on translating this into other open-source languages. Additionally, I should dedicate more work to optimising the Bayesian platform to reduce the time allocated to computations, the limiting step of the process.

Finally, I compiled all the methodologies presented to infer mathematical models for biological systems and perform Bayesian OED for model selection and calibration into an open-source easy-to-use Julia package, BOMBs.jl, in Chapter 8. Additionally, the package automatically generates necessary scripts to work with ODEs in Julia and simplifies model simulation and pseudo-data generation. The package leverages state-of-the-art optimisers and the Just-In-Time compilation from Julia to work with mathematical models effortlessly and efficiently. The package allows for broader adoption of mathematical models across inexperienced users in biological science. On top of this, it provides a build platform for OED, aiding the spread of these highly useful methodologies. Nonetheless, I acknowledge that the package is still far from its final version. BOMBs.jl would benefit from improvements in code efficiency or expansions to include different kinds of pseudo-data noise (e.g. multiplicative noise) or types of mathematical modelling supported (e.g. SBML or stochastic models). Even though I filled the package with user input checks, I could also expand these to ease its use amongst inexperienced users. I also intend to add additional parameter estimation strategies besides MLE (e.g. residual sum of squares) and optimisers for MLE and OED. Furthermore, I could add a frequentist FIM-based OED for model calibration scheme for users that want to use them within the same platform. Finally, future work could be directed toward designing a graphical user interface to delete the need for any coding knowledge from users.

Hence, in this thesis, I addressed and proposed solutions to some of the pitfalls of using mathematical models in biological sciences. I used Bayesian statistics to infer model parameters from experimental data, analysing the quality of the results. Then I contributed to the field of optimal experimental design by proposing and validating strategies directed at model selection and calibration in a Bayesian framework. Then, I build experimental platforms to aid process automation for microfluidic experiments. Finally, I aimed to be a step closer to the broader adoption of mathematical modelling in biological sciences with the

design of BOMBs.jl. I provided straightforward solutions to the stated problems, validations of new methodologies, *in vivo* examples and an easy-to-use platform to perform most of the computations shown in this work in an automated fashion.

Appendix A

Supplementary Data

A.1 Supplementary Figures for Chapter 4

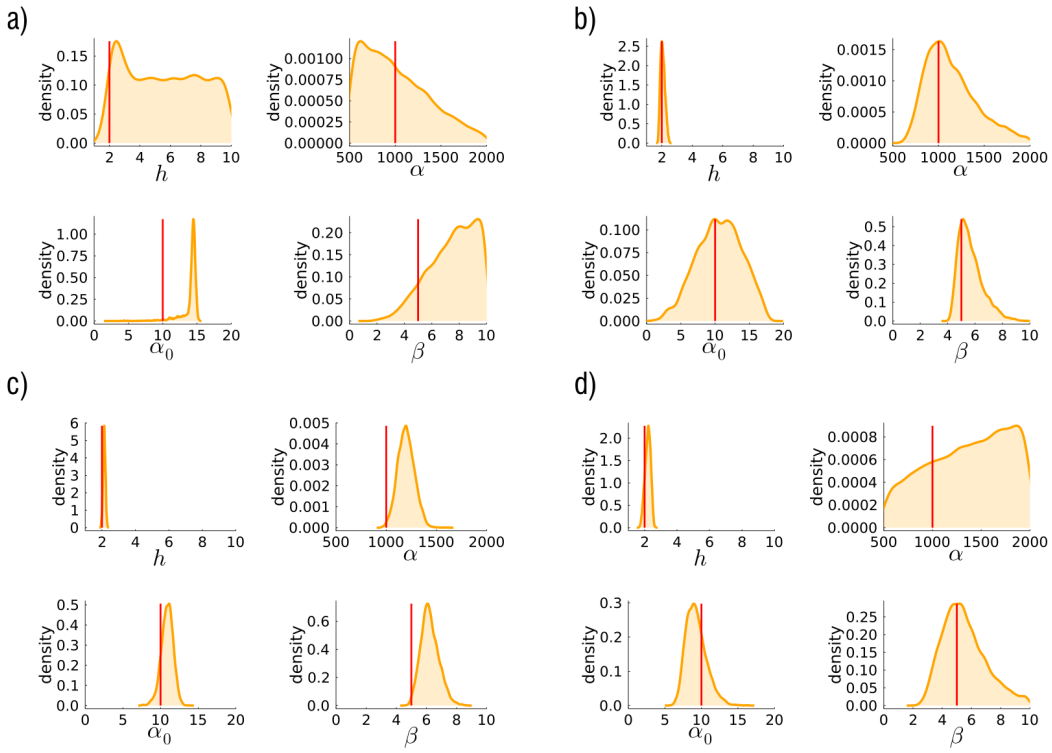


Figure S1: Repressilator posterior kernel densities (orange) against the parameter values used to generate the pseudo-data (red) for experimental sets 1 (a), 2 (b), 3 (c) and 4 (d).

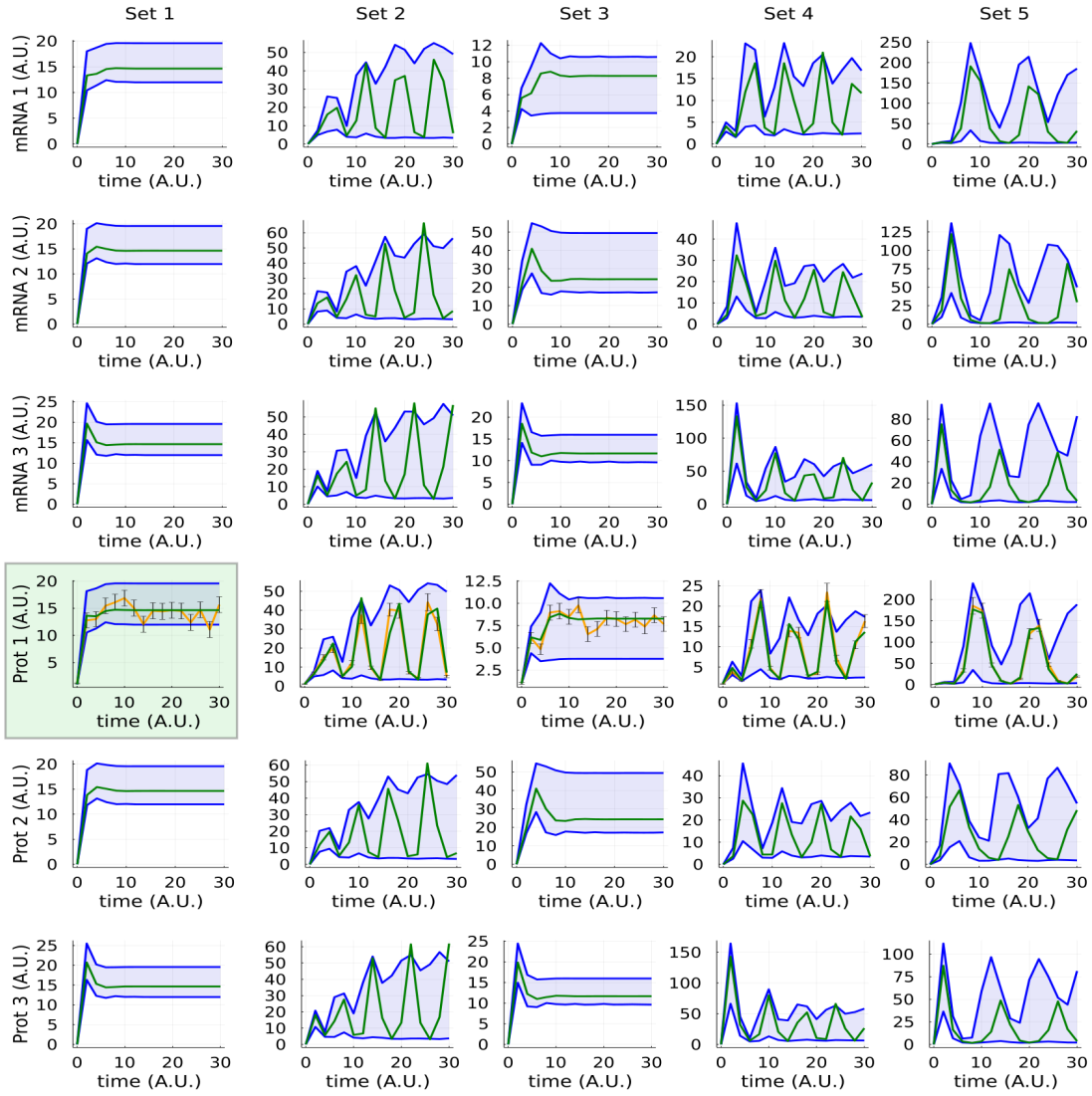


Figure S2: Repressilator posterior predictions (blue), against simulations using the parameter values used to generate pseudo-data (green) and pseudo-data (orange), for all states (y axis) and all data sets (x axis). The posterior used for simulations is the one inferred using the pseudo-data from set 1 (green box).

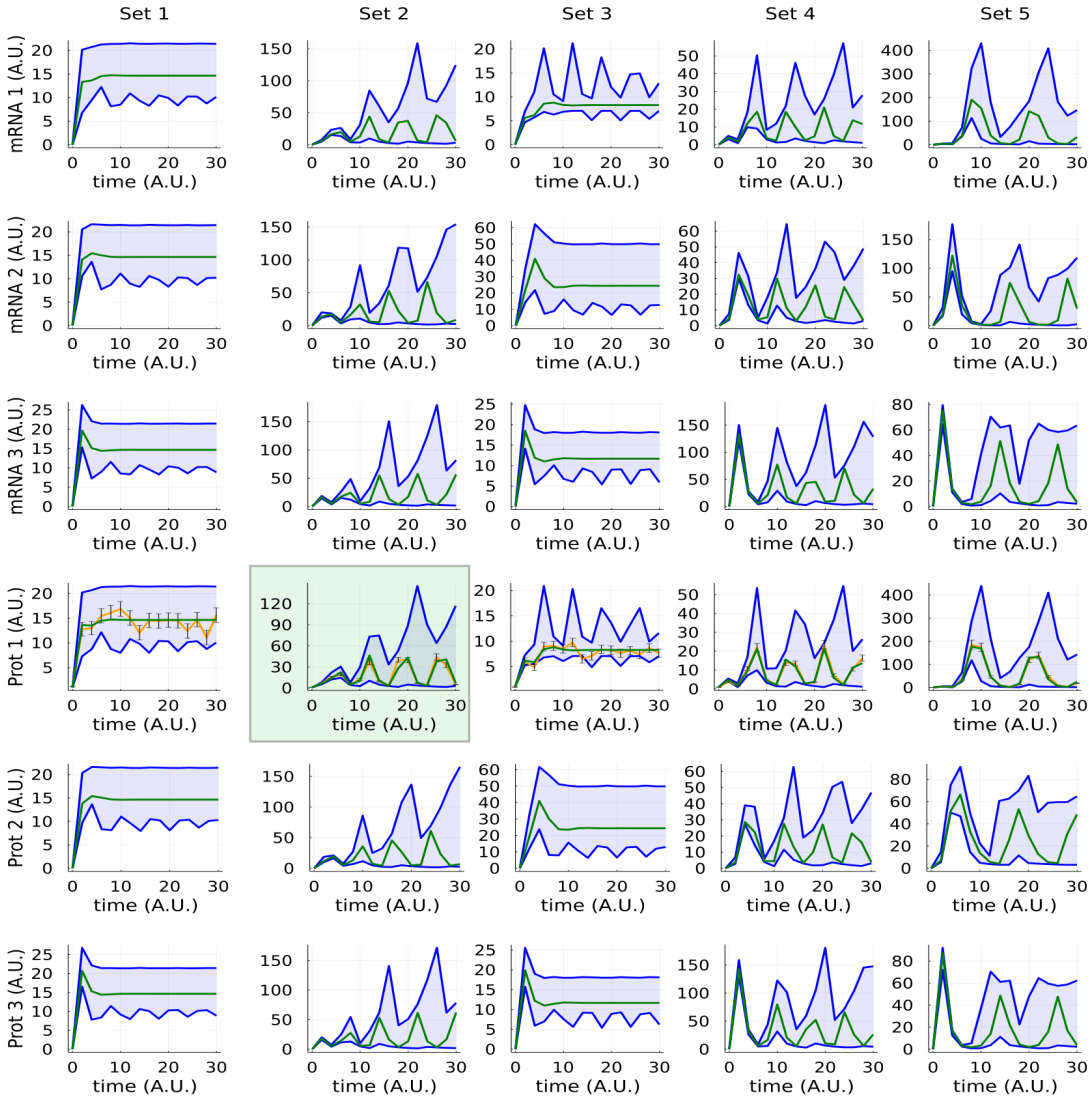


Figure S3: Repressilator posterior predictions (blue), against simulations using the parameter values used to generate pseudo-data (green) and pseudo-data (orange), for all states (y axis) and all data sets (x axis). The posterior used for simulations is the one inferred using the pseudo-data from set 2 (green box).

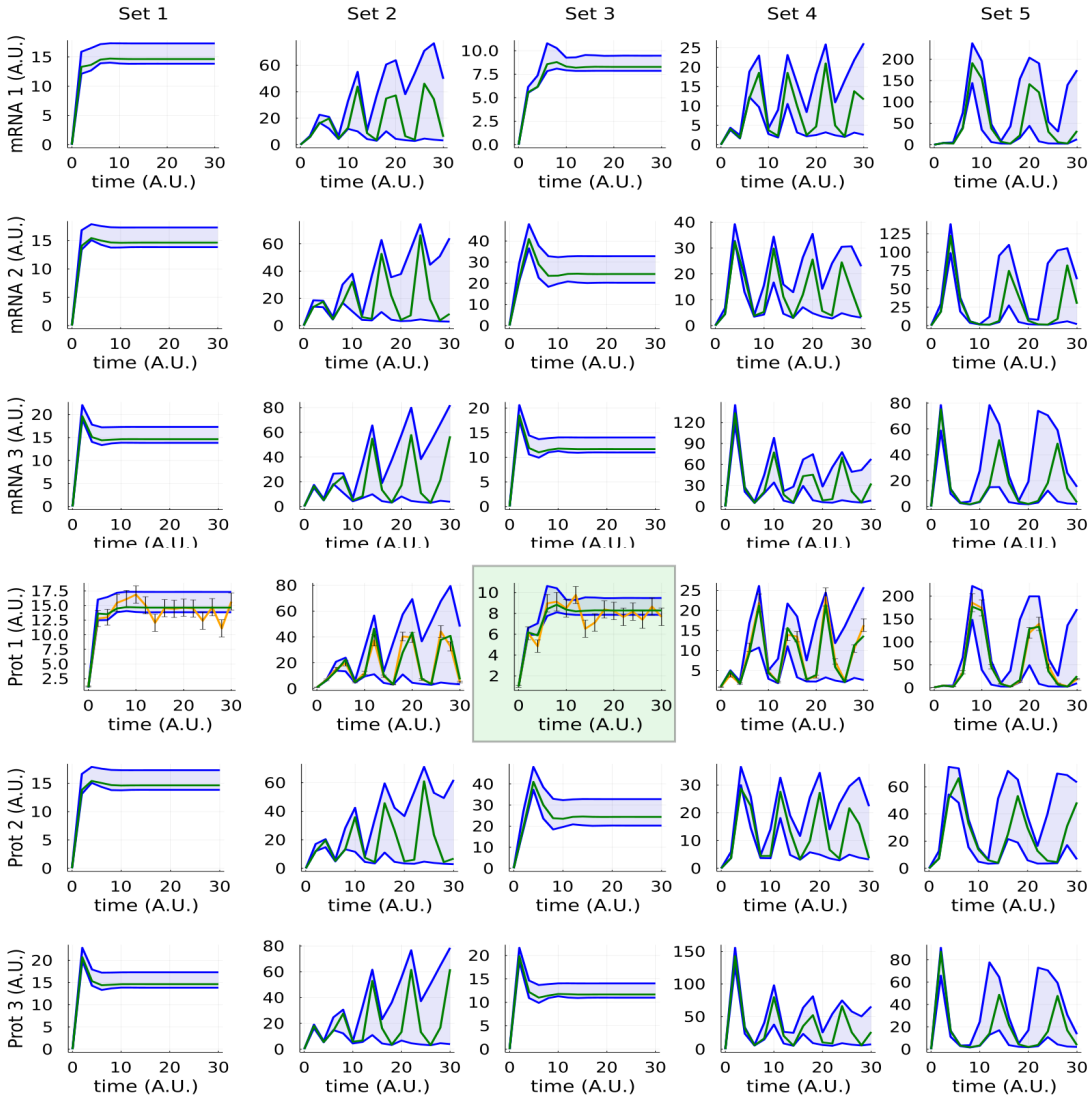


Figure S4: Repressilator posterior predictions (blue), against simulations using the parameter values used to generate pseudo-data (green) and pseudo-data (orange), for all states (y axis) and all data sets (x axis). The posterior used for simulations is the one inferred using the pseudo-data from set 3 (green box).

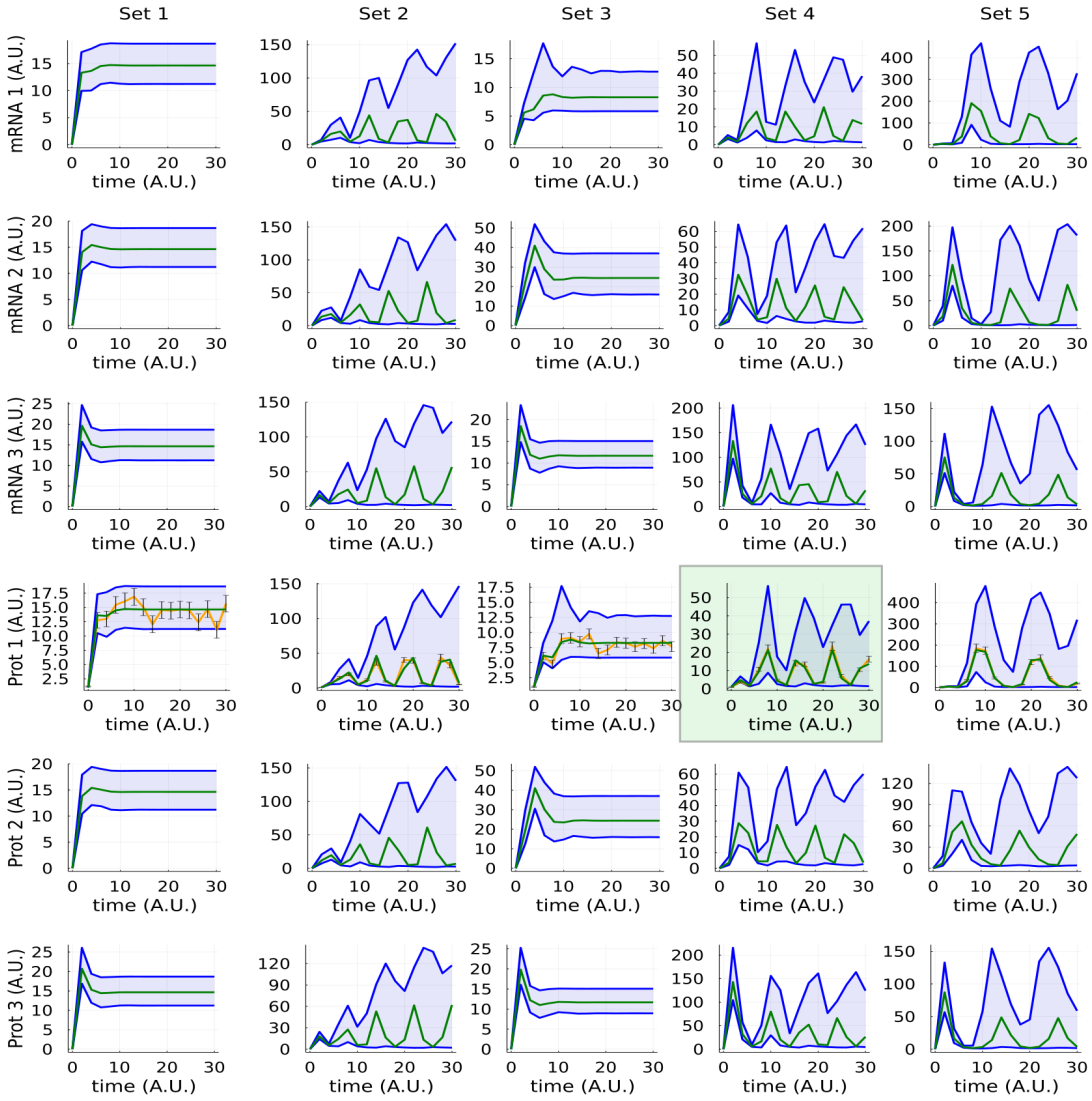


Figure S5: Repressilator posterior predictions (blue), against simulations using the parameter values used to generate pseudo-data (green) and pseudo-data (orange), for all states (y axis) and all data sets (x axis). The posterior used for simulations is the one inferred using the pseudo-data from set 4 (green box).

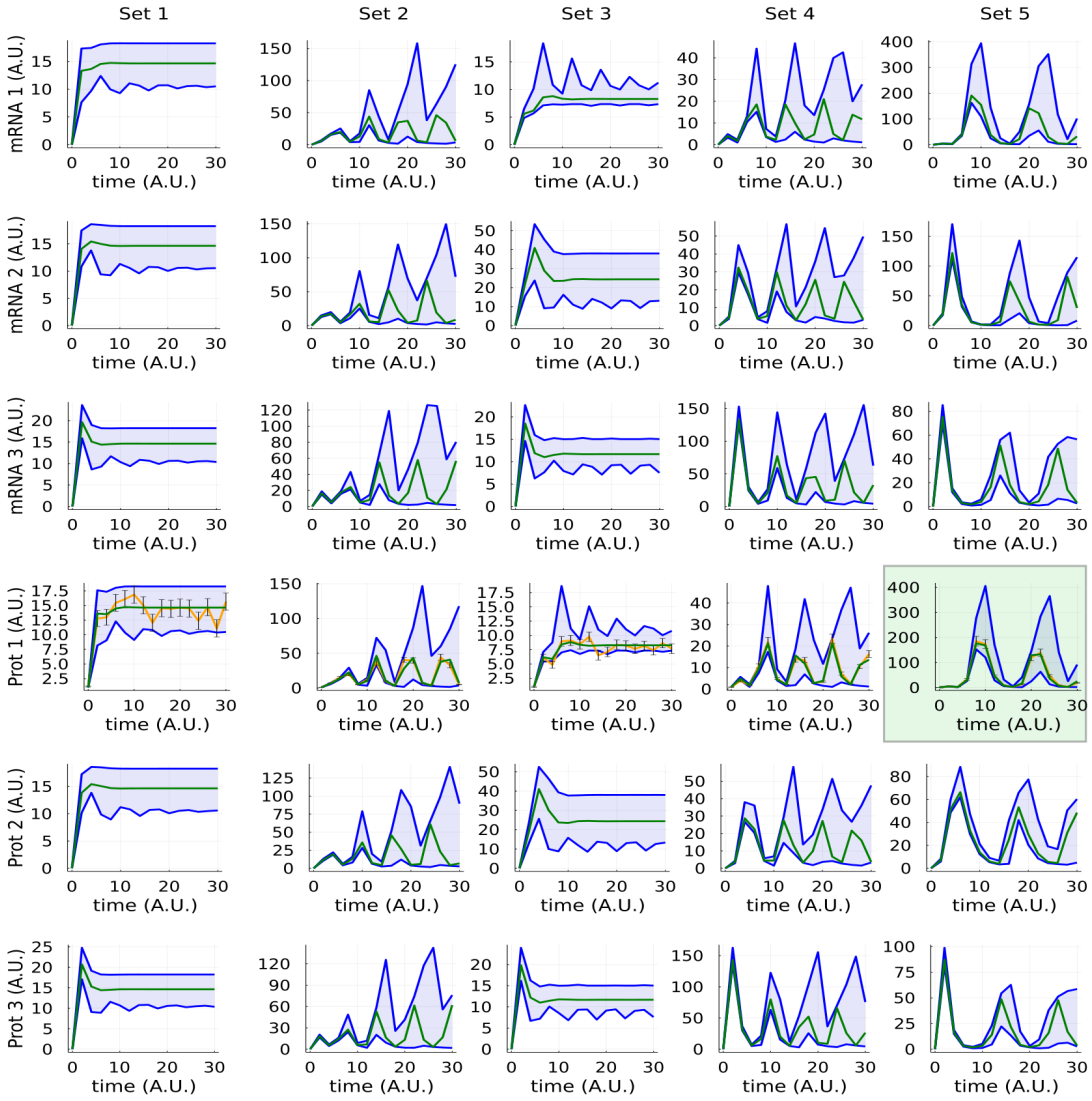


Figure S6: Repressilator posterior predictions (blue), against simulations using the parameter values used to generate pseudo-data (green) and pseudo-data (orange), for all states (y axis) and all data sets (x axis). The posterior used for simulations is the one inferred using the pseudo-data from set 5 (green box).

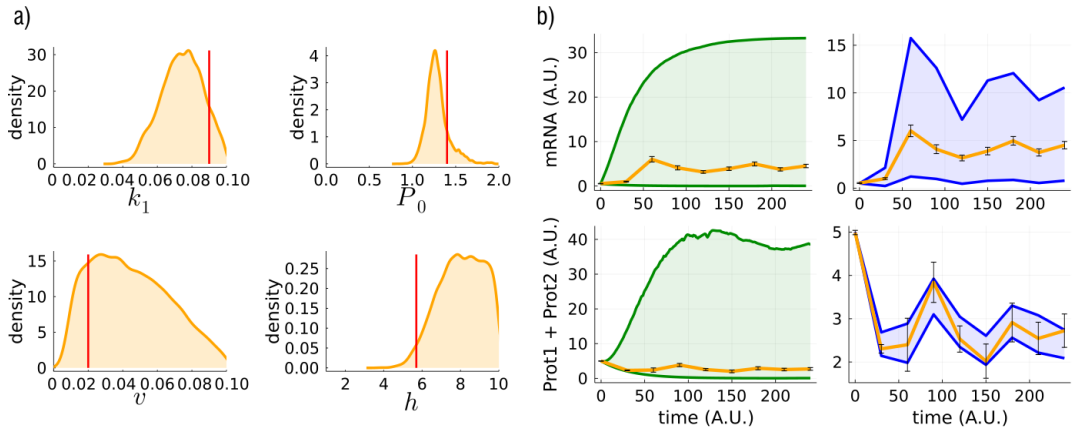


Figure S7: Bayesian inference of the Hes1 system considering the total Hes1 cell protein as observable. a) Kernel densities of the posteriors (orange) compared to the parameter values (red) used to generate the pseudo-data. b) Prior (green) and posterior (blue) model predictions (99.5th and 0.5th percentiles) agains mean (orange) and standard deviation (grey) of the pseudo-data (10 % heteroscedastic noise) used for the inference (bottom panel) and the pseudo-data representing the mRNA concentration (top panel).

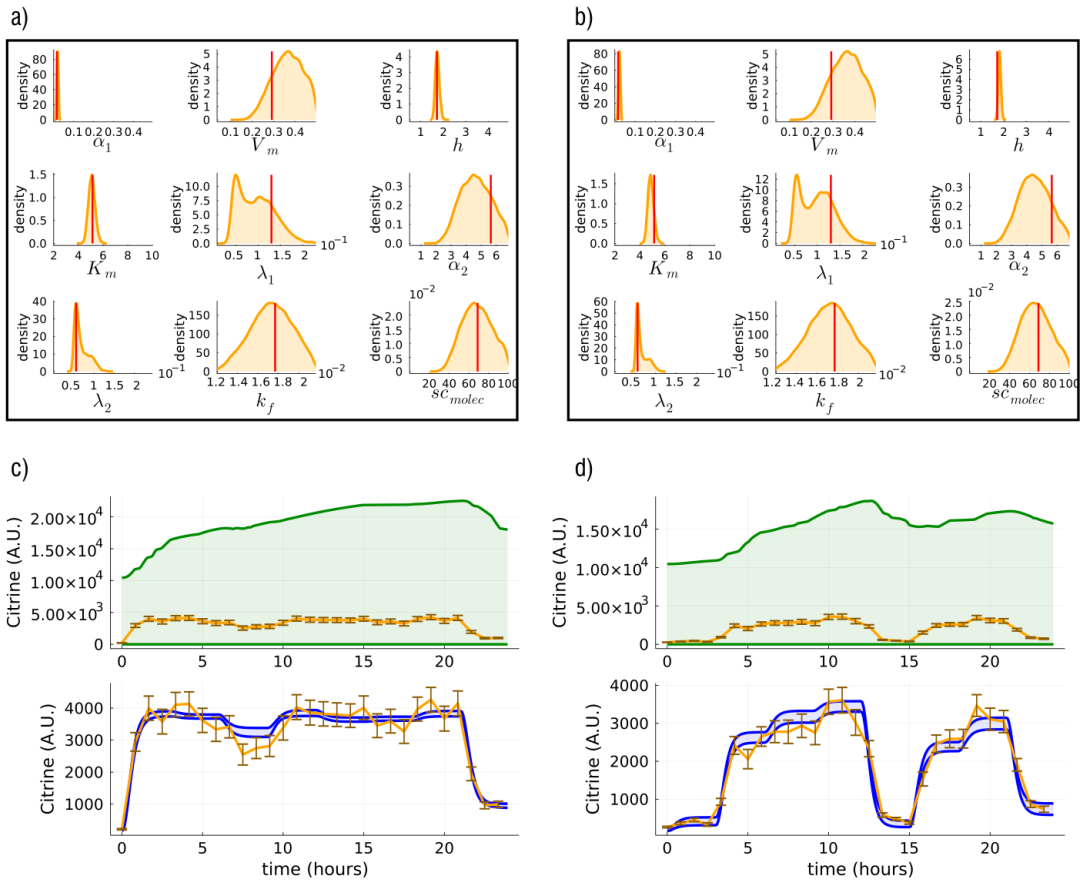


Figure S8: Bayesian inference of the orthogaonal inducible promoter developed in *S. cerevisiae*. a) Kernel densities of the posterior distributions (orange) inferred, compared to the parameter values used to generate the pseudo-data (red). In the experiment simulated, the concentration of IPTG changes every 3 hours at a random value selected between the bounds 0 and 50 μM (38, 27, 12, 44, 21, 23, 39, 2). The pseudo-data used for the inference assumes a 10 % heteroscedastic additive Gaussian noise. b) Kernel densities obtained after inference of an experiment where the concentration of IPTG changes every 3 hours at a random value selected between the bounds 0 and 15 μM (1, 7, 10, 14, 1, 6, 9, 2). c) Comparison between the prior (green) and posterior (blue) predictions (99.5th and 0.5th percentiles) and mean (orange) and standard deviation (grey) of the pseudo-data used for the inference. The posterior used is the one represented in panel a). d) Comparison between the prior (green) and posterior (blue) predictions (99.5th and 0.5th percentiles) and mean (orange) and standard deviation (grey) of the pseudo-data used for the inference. The posterior used is the one represented in panel b).

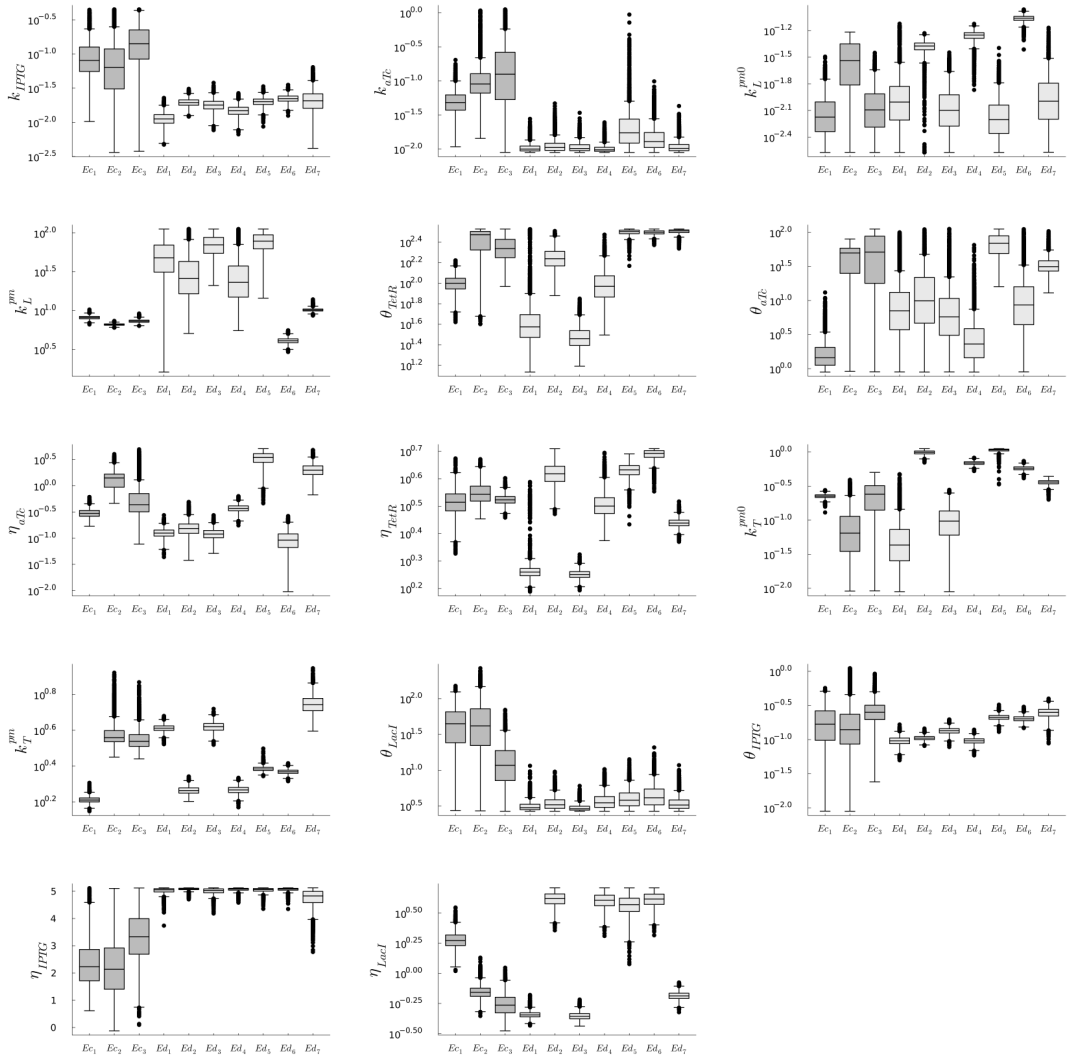


Figure S9: Marginal distribution box plots for each parameter for the toggle switch model M_3 inferred in each experiment considered from Lugagne *et al.* 2017 [85].

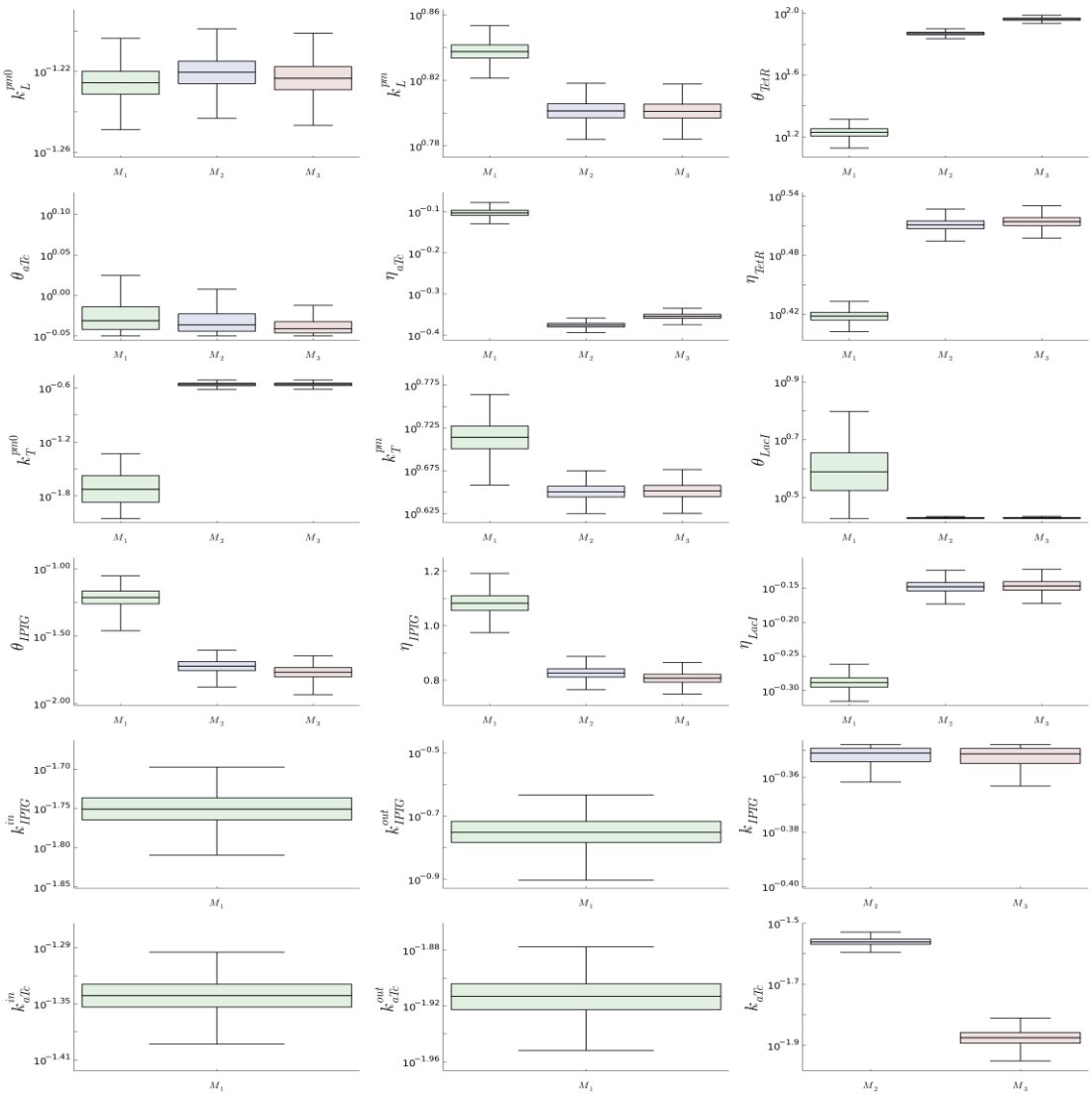


Figure S10: Marginal distribution box plots for each parameter for the toggle switch models M_1, M_2 and M_3 inferred in all experiments considered from Lugagne *et al.* 2017 [85].

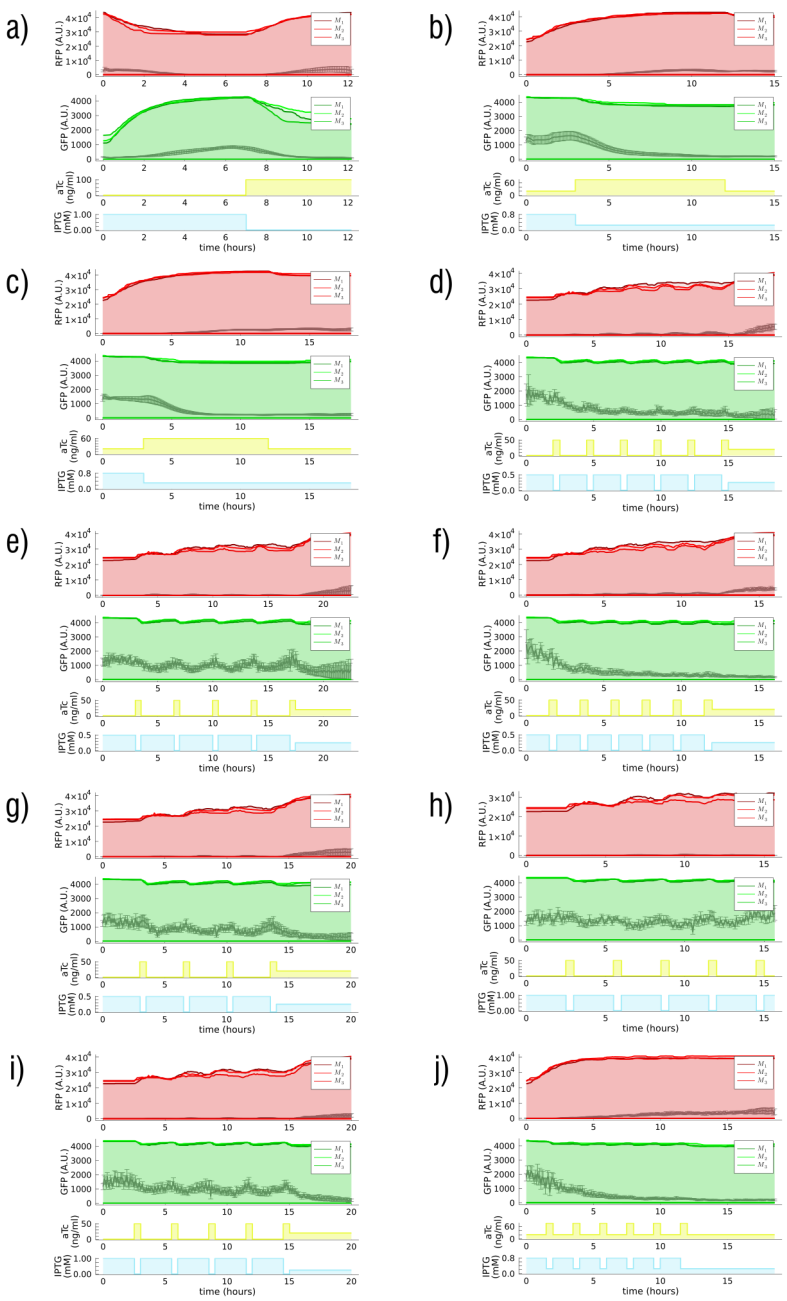


Figure S11: Prior predictions (99.5th and 0.5th percentiles) for the three toggle switch models considered (M_1, M_2 and M_3) for the 10 experiments used from Lugagne *et al.* 2017 [85] in the order described in Table 3.5.

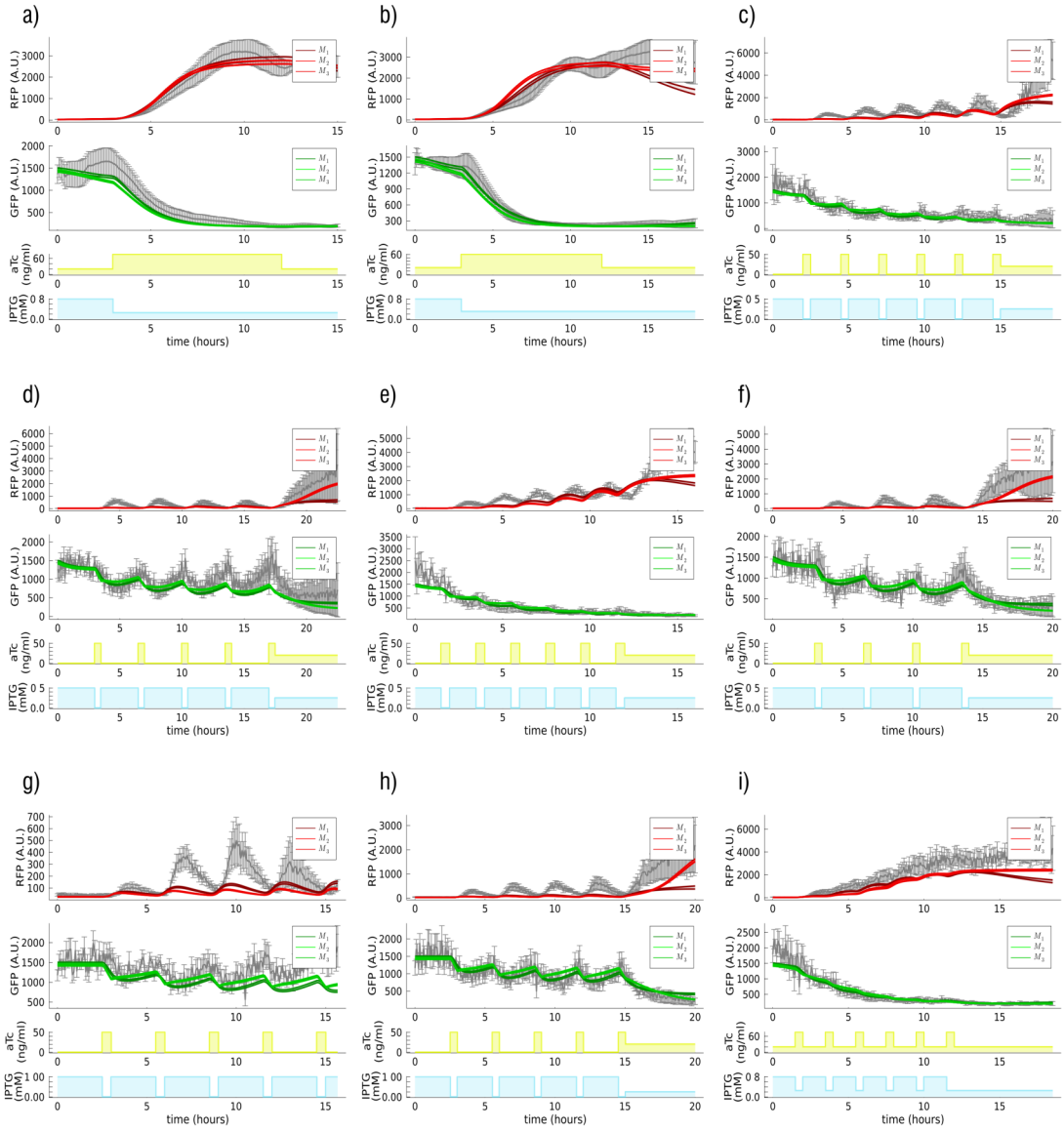


Figure S12: Posterior predictions (99.5th and 0.5th percentiles) for the three toggle switch models considered (M_1, M_2 and M_3) for the experiments Ec_2 (a), Ec_3 (b), Ed_1 (c), Ed_2 (d), Ed_3 (e), Ed_4 (f), Ed_5 (g), Ed_6 (h) and Ed_7 (i) from Lugagne *et al.* 2017 [85].

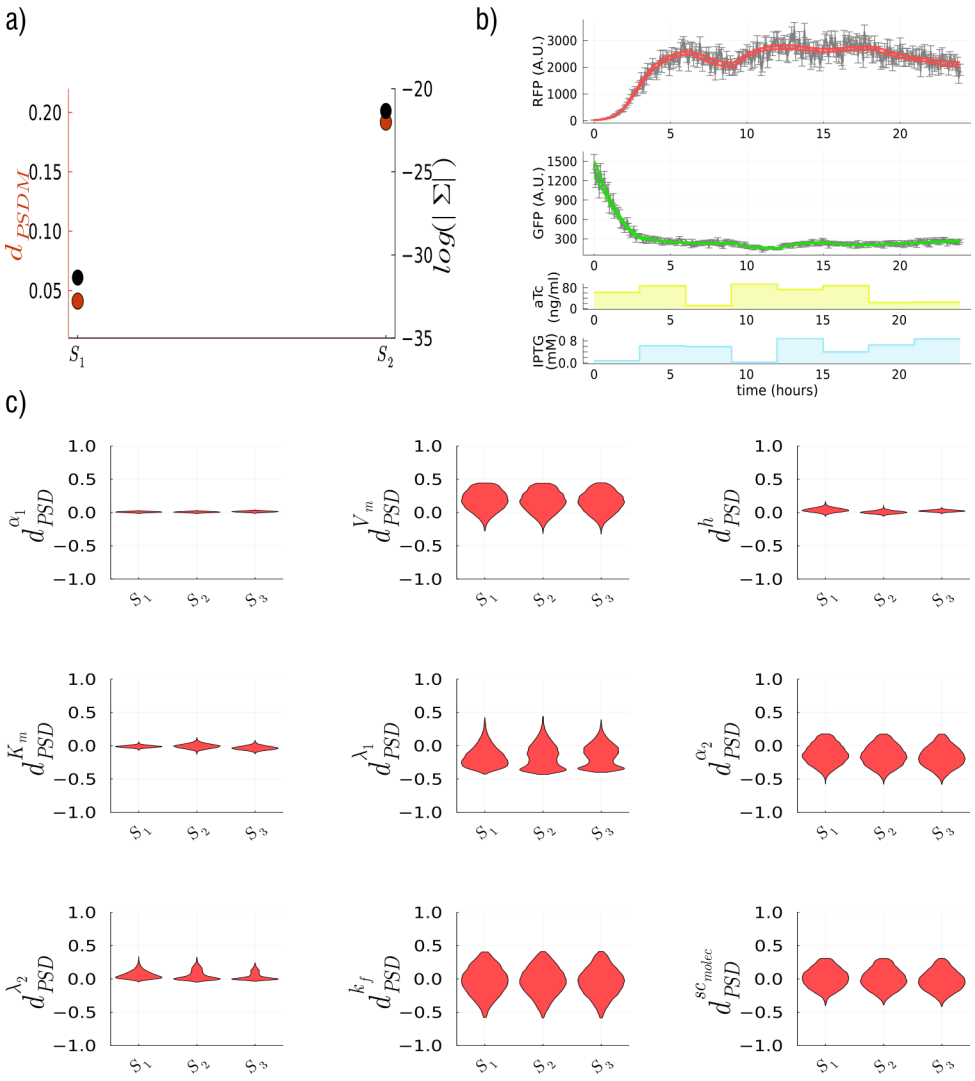


Figure S13: *A posteriori* informative content analysis. a) Comparison between the average posterior sample distance to true value d_{PSDM} and a metric for the posterior covariance for the Hes1 system where S_1 indicates the posterior inferred considering mRNA as the system's observable and S_2 when considering the total cell Hes1 as the observable. b) Pseudo-data used to infer the toggle switch model M_3 in Section 4.2.1 and posterior prediction (99.5th and 0.5th percentiles). c) d_{PSD} for each parameter of the orthogonal inducible promoter model inferred for each of the three pseudo-data sets considered.

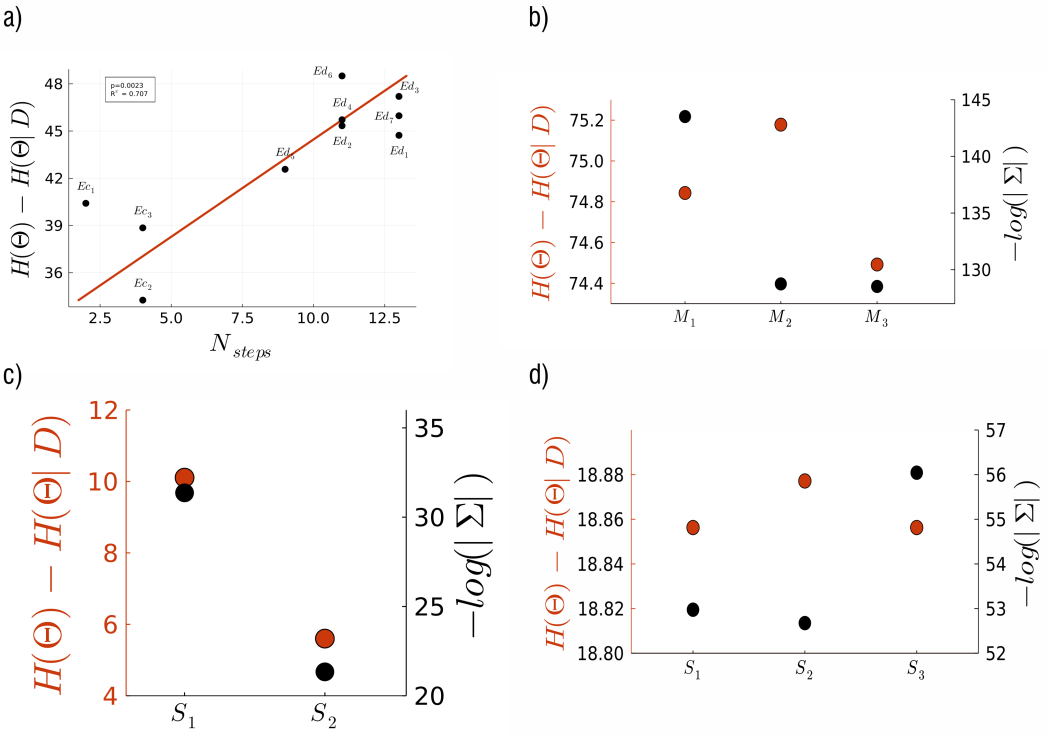


Figure S14: Quantitative informative content analysis of experiments. a) Correlation plot between the information gain and amount of concentration changes for the inducers for the ten data sets considered from Lugagne *et al.* 2017 [85] for the toggle switch model M_3. b) Comparison between the information gain and posterior covariance for the three toggle switch models M_1, M_2 and M_3 after being inferred using the 10 data sets considered from Lugagne *et al.* 2017 [85]. c) Comparison between the Hes1 information gain and posterior covariance when considering as an observable mRNA (S_1) or the total Hes1 cell concentration (S_2). d) Comparison between the orthogonal inducible promoter information gain and posterior covariance for the three pseudo-data sets considered.

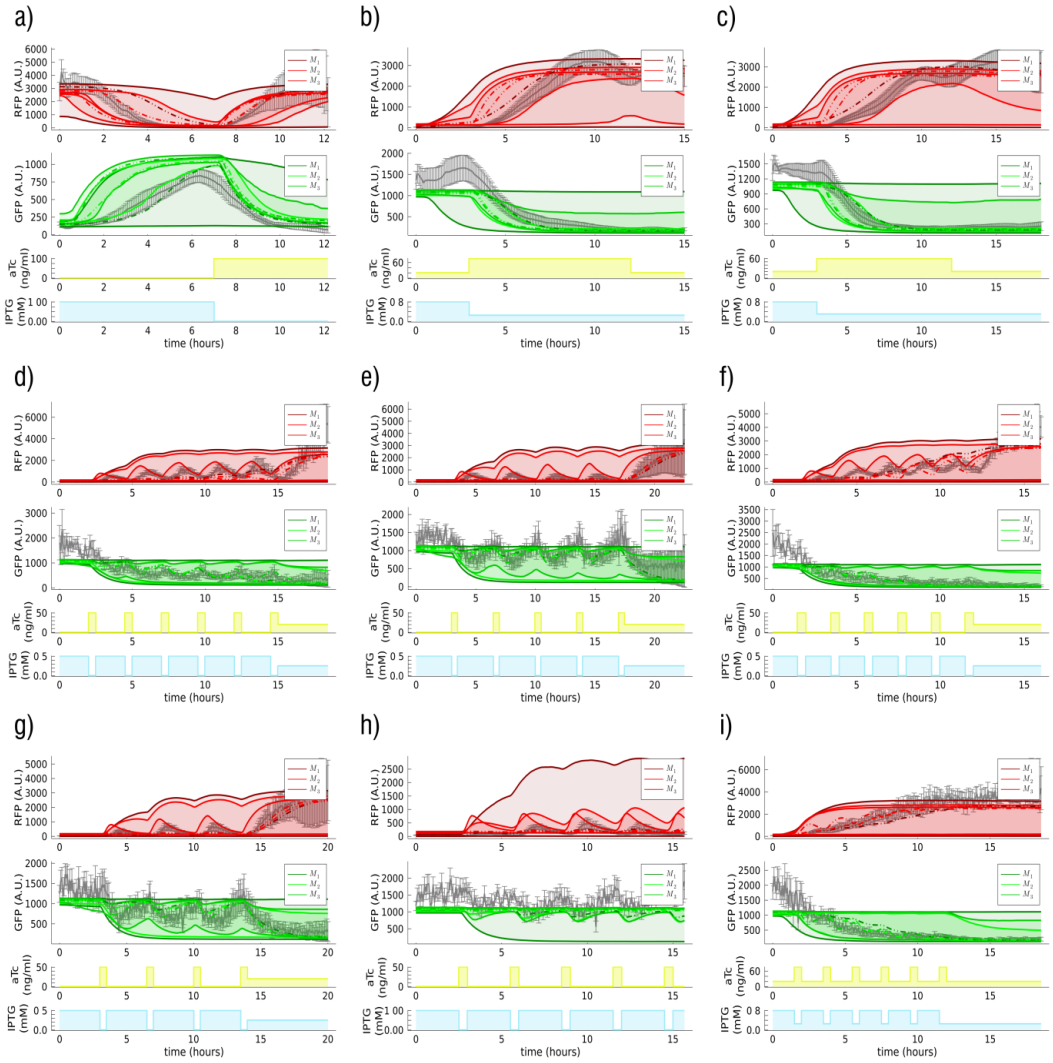


Figure S15: Predictions (99.5th and 0.5th percentiles) for the three toggle switch models considered (M_1, M_2 and M_3) using 8000 samples from the approximated multivariate normal distribution resultant from the frequentist parameter estimation for the experiments Ec_1 (a), Ec_2 (b), Ec_3 (c), Ed_1 (d), Ed_2 (e), Ed_3 (f), Ed_4 (g), Ed_5 (h) and Ed_7 (i) from Lugagne *et al.* 2017 [85]. Dashed lines represent the models simulation using the best parameter estimate.

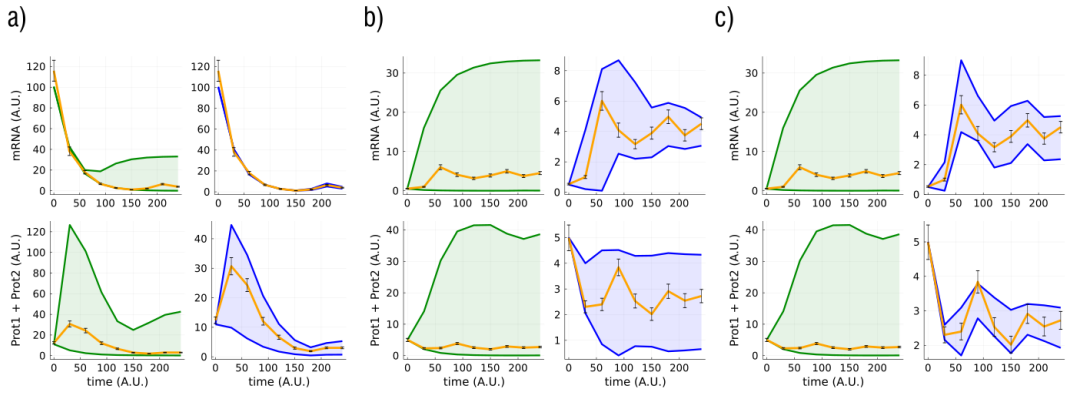


Figure S16: Hes1 model predictions when inferred using the data set that considers the system's initial condition as $y_0 = [100.5, 1.0, 10.0]$ for mRNA, cytosolic Hes1 and nuclear Hes1. a) Comparison between the prior (green) and posterior (blue) predictions (99.5th and 0.5th percentiles) and mean (orange) and standard deviation (grey) of the pseudo-data used for the inference (top panel) and pseudo-data representing the total Hes1 concentration in cells (bottom panel). b) Comparison between the prior (green) and posterior (blue) predictions (99.5th and 0.5th percentiles) and mean (orange) and standard deviation (grey) of the original experimental set ($y_0 = [0.5, 3.0, 2.0]$) when the system was inferred using mRNA as an observable. c) Comparison between the prior (green) and posterior (blue) predictions (99.5th and 0.5th percentiles) and mean (orange) and standard deviation (grey) of the original experimental set ($y_0 = [0.5, 3.0, 2.0]$) when the system was inferred using the total Hes1 cell concentration as an observable.

A.2 Supplementary Figures for Chapter 5

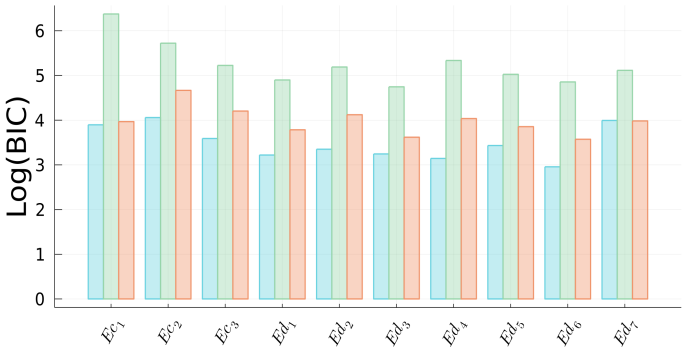


Figure S17: Bar plot for the minus base-10 logarithm of the Bayesian information criteria (BIC) for the three toggle switch models (M_1, M_2 and M_3) using the best parameter estimate (frequentist parameter estimation) to predict the ten data sets considered from Lugagne *et al.* 2017 [85].

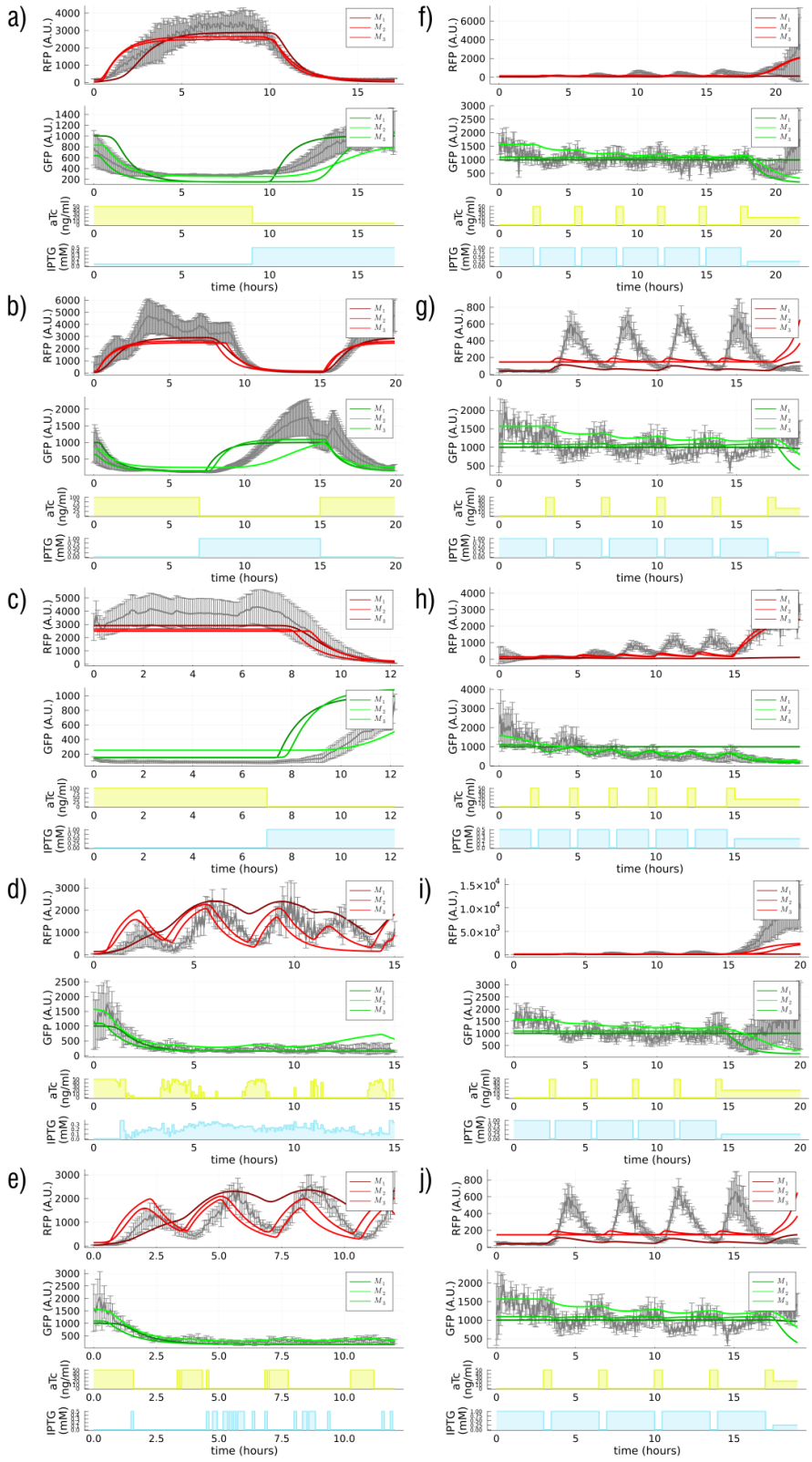


Figure S18: Frequentist predictions for the three toggle switch models considered (M_1, M_2 and M_3) for the validation experiments Ec_4 (a), Ec_5 (b), Ec_6 (c), Ed_8 (d), Ed_9 (e), Ed_{10} (f), Ed_{11} (g), Ed_{12} (h), Ed_{13} (i) and Ed_{14} (j) from Lugagne *et al.* 2017 [85].

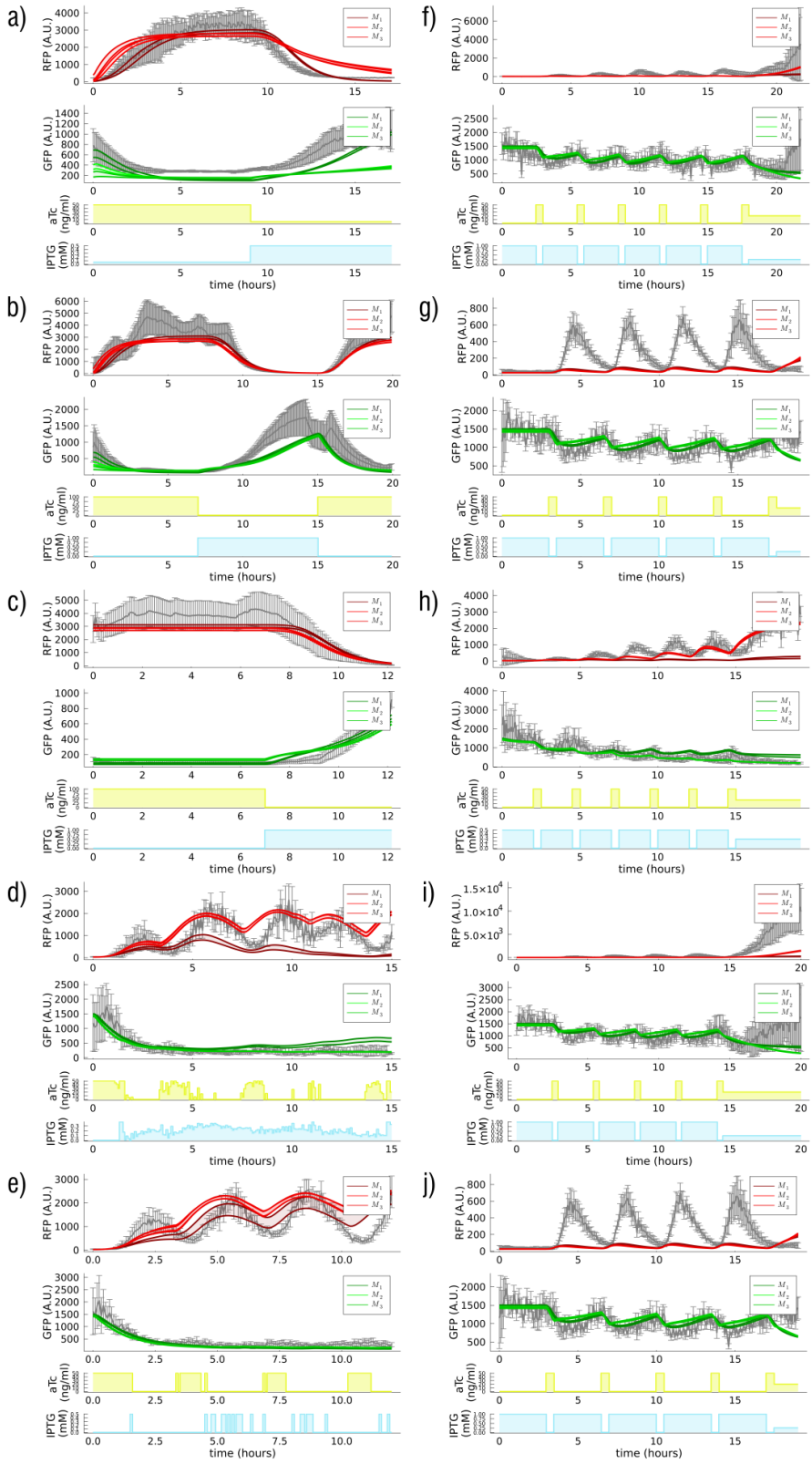


Figure S19: Bayesian predictions (99.5th and 0.5th percentiles) for the three toggle switch models considered (M_1, M_2 and M_3) for the validation experiments Ec_4 (a), Ec_5 (b), Ec_6 (c), Ed_8 (d), Ed_9 (e), Ed_{10} (f), Ed_{11} (g), Ed_{12} (h), Ed_{13} (i) and Ed_{14} (j) from Lugagne *et al.* 2017 [85].

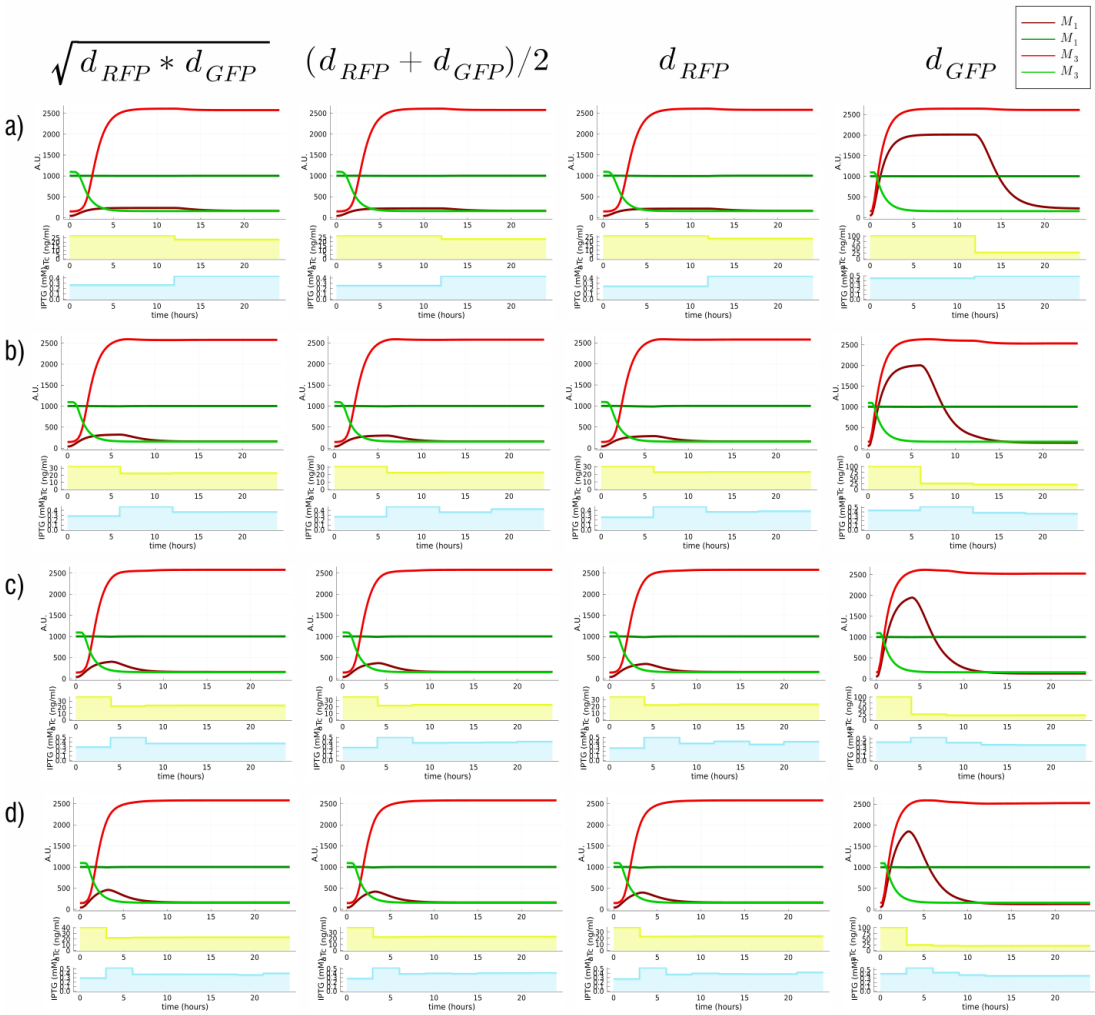


Figure S20: Frequentist optimal experimental for model selection design results (bottom panels) and competing model simulations (top panels) for optimisations considering 2 (a), 4 (b), 6 (c) and 8 (d) input perturbation steps and for each combination of the observable(s) considered.

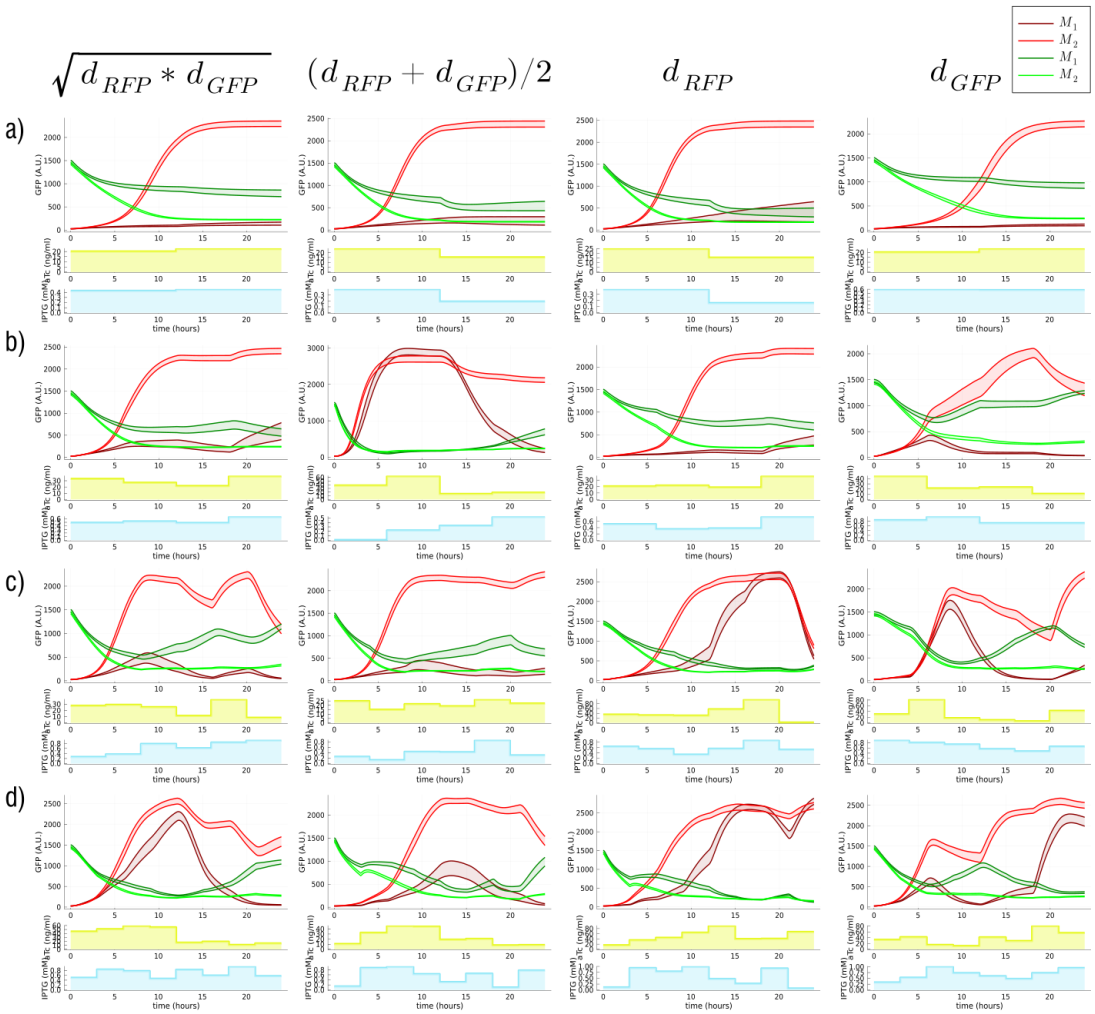


Figure S21: Bayesian optimal experimental design for model selection results (bottom panels) and competing model posterior predictions (top panels) for optimisations considering 2 (a), 4 (b), 6 (c) and 8 (d) input perturbation steps and for each combination of the observable(s) considered.

A.3 Supplementary Figures for Chapter 6

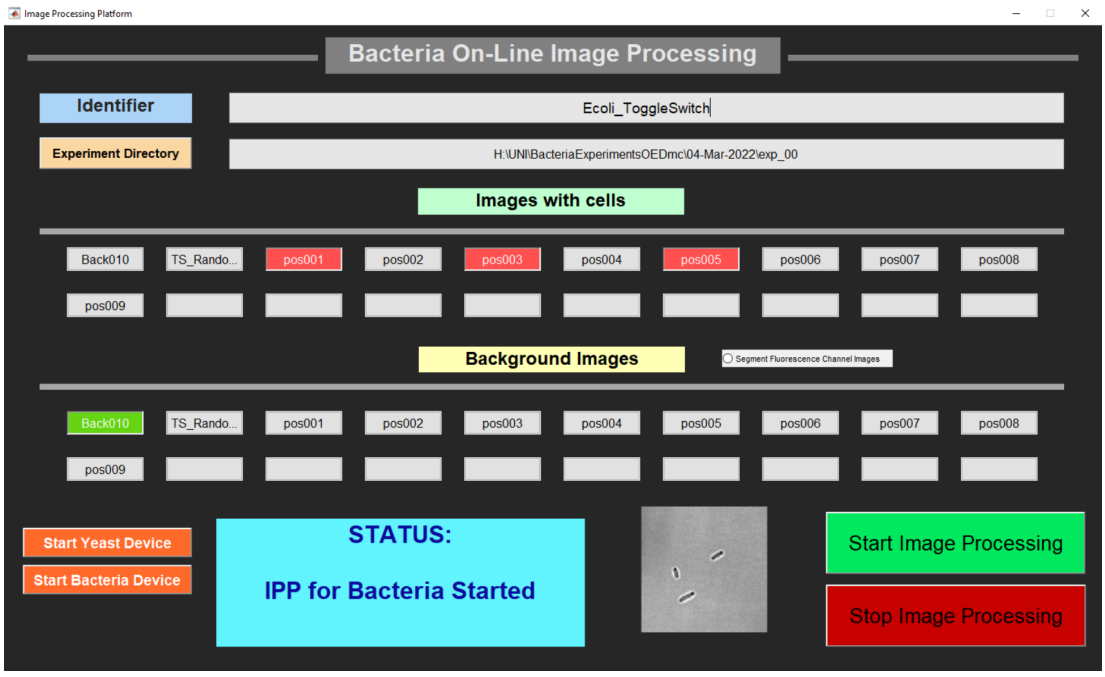


Figure S22: Graphical user interface (GUI) for the image processing platform. Users only need to specify the file identifier for the results, the directory containing the experimental images acquired during an experiment and the different positions to use for cell fluorescence extraction and background fluorescence computation. The platform works for yeast and bacteria microscopy images acquired from microfluidic devices.

A.4 Supplementary Figures for Chapter 7

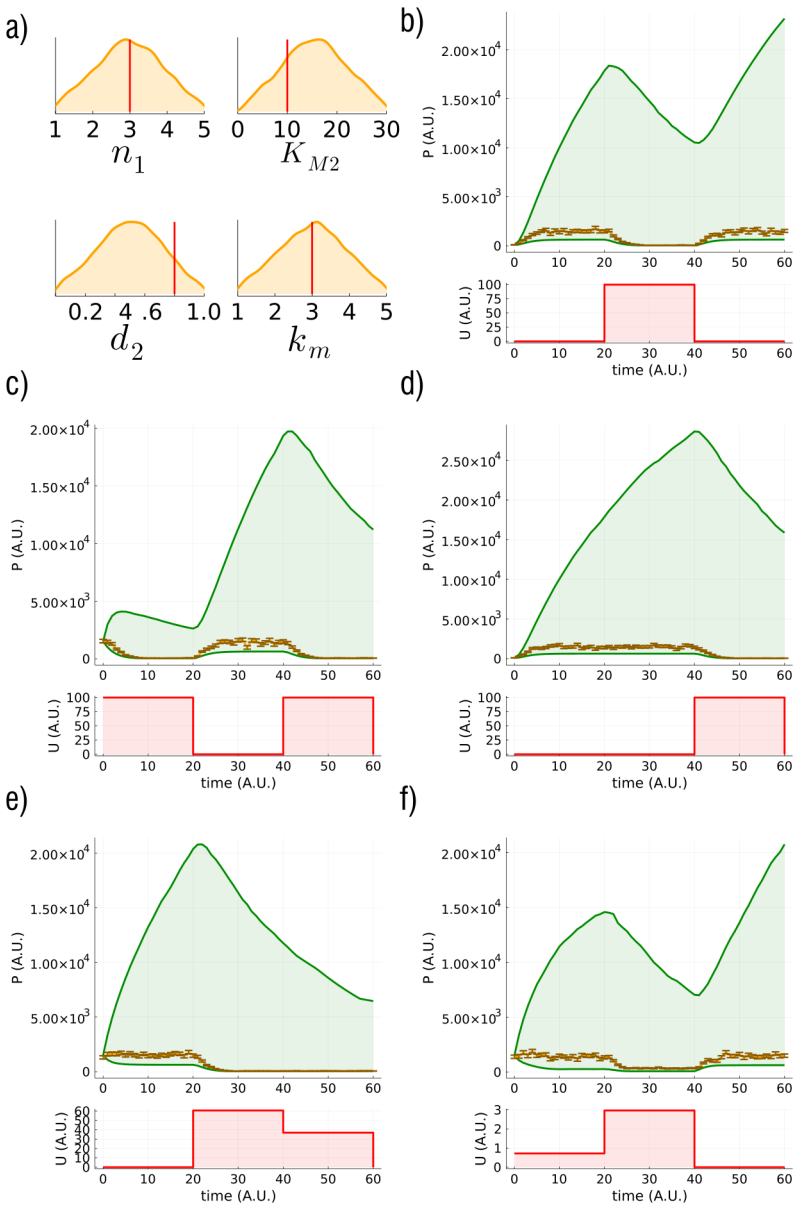


Figure S23: Parameter priors and pseudo-data used for the Bayesian OED proof of concept. a) Marginal prior distributions set for the analysis. Pseudo-data and prior model predictions from the intuition-driven experiment 1 (b) and 2 (c), the optimally designed experiment 1 (d) and 2 (e) and the sequential optimal experiment (f). The bottom panels indicate the input U profiles.

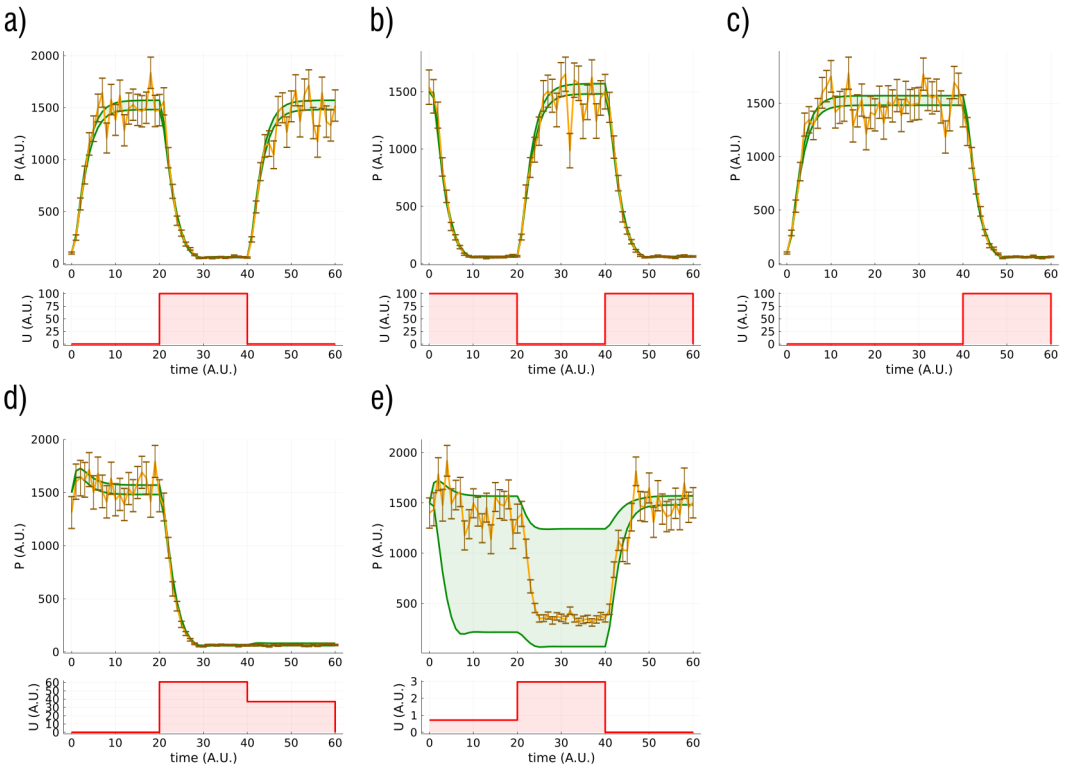


Figure S24: Pseudo-data and intuition-driven posterior predictions used for the Bayesian OED proof of concept. Pseudo-data and intuition-driven posterior model predictions from the intuition-driven experiment 1 (a) and 2 (b), the optimally designed experiment 1 (c) and 2 (d) and the sequential optimal experiment (e). The bottom panels indicate the input U profiles.

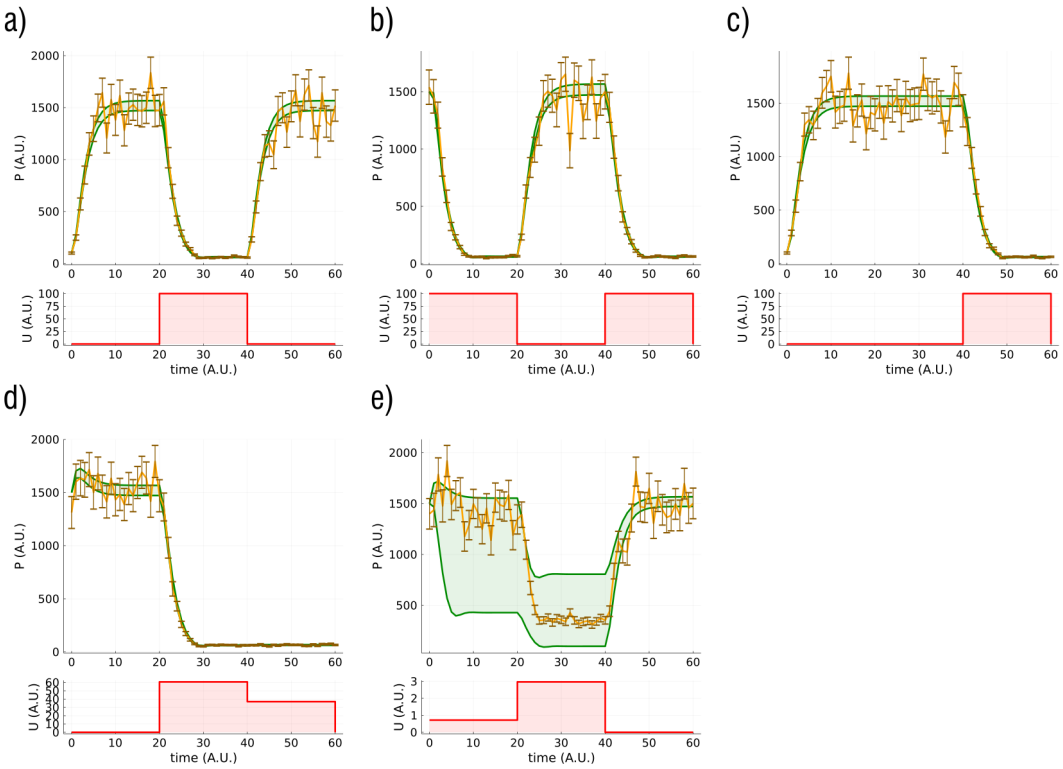


Figure S25: Pseudo-data and optimally designed posterior predictions used for the Bayesian OED proof of concept. Pseudo-data and optimally designed posterior model predictions from the intuition-driven experiment 1 (a) and 2 (b), the optimally designed experiment 1 (c) and 2 (d) and the sequential optimal experiment (e). The bottom panels indicate the input U profiles.

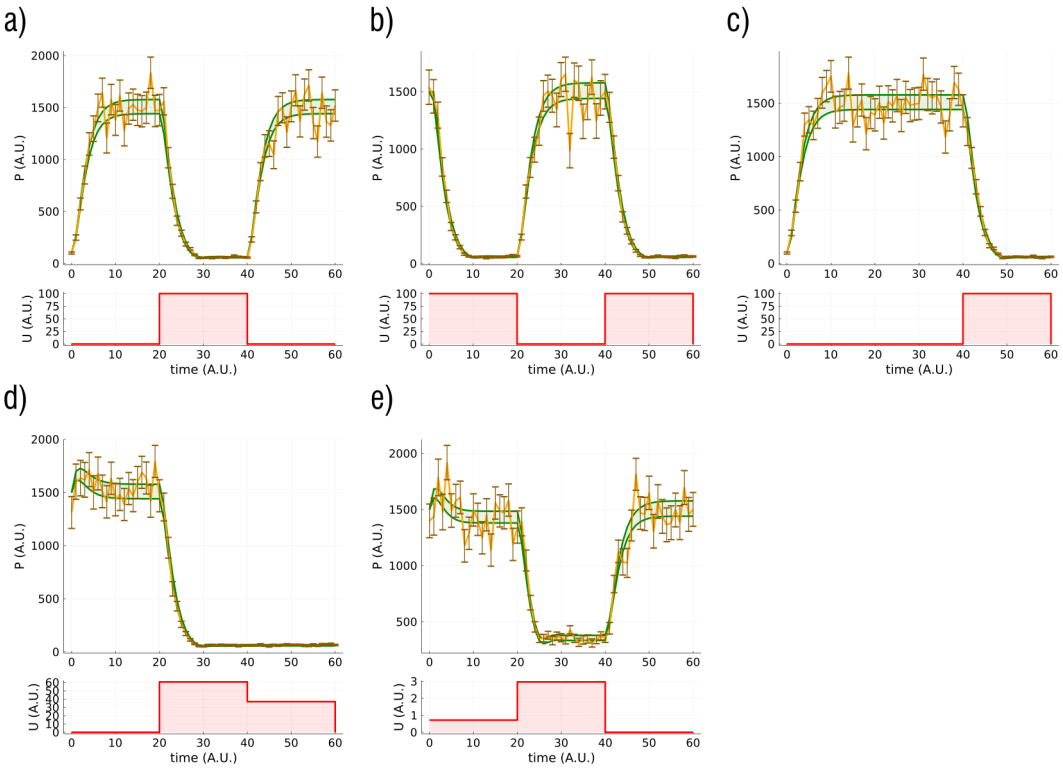


Figure S26: Pseudo-data and sequential optimally designed posterior predictions used for the Bayesian OED proof of concept. Pseudo-data and sequential optimally designed posterior model predictions from the intuition-driven experiment 1 (a) and 2 (b), the optimally designed experiment 1 (c) and 2 (d) and the sequential optimal experiment (e). The bottom panels indicate the input U profiles.

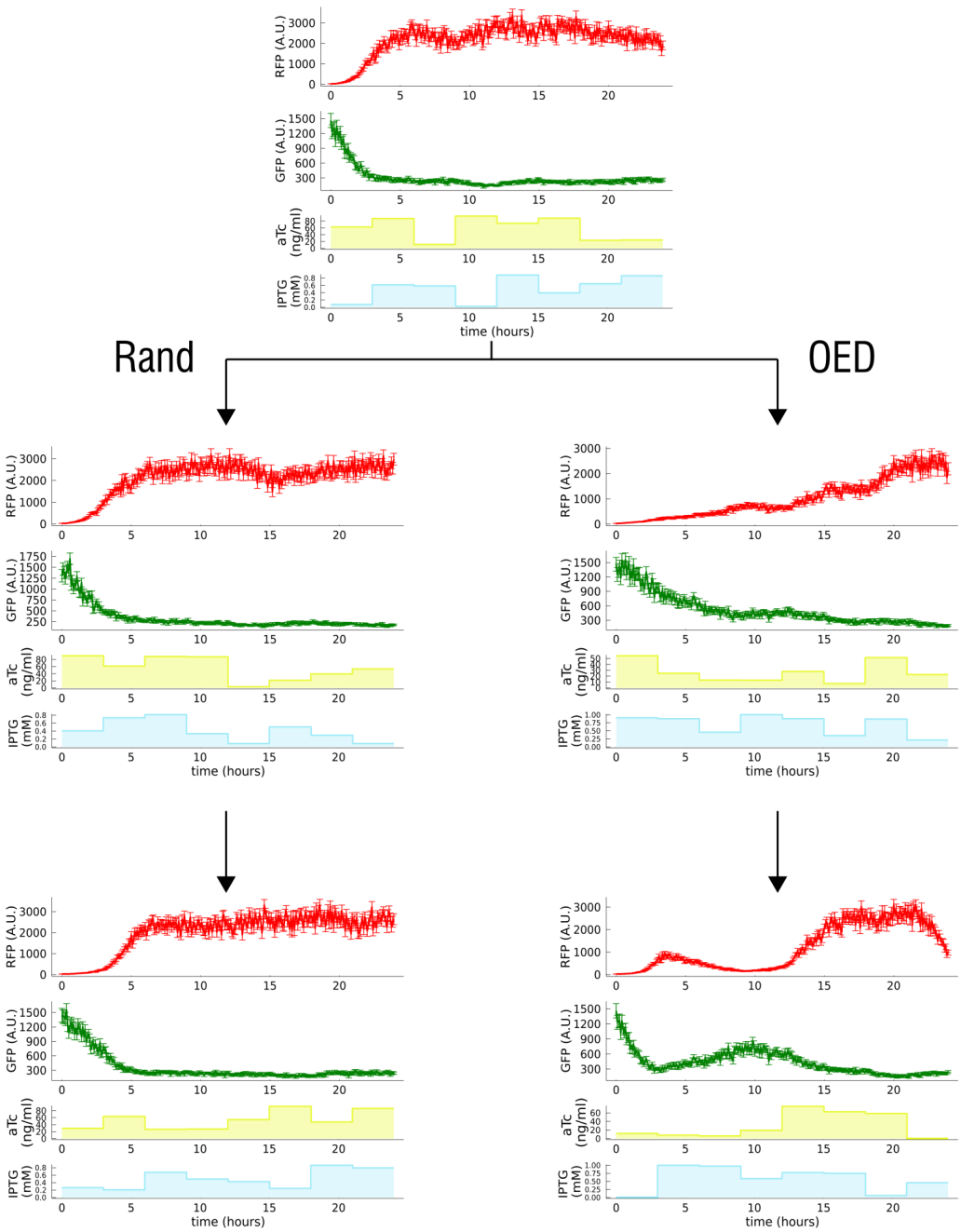


Figure S27: Pseudo-data and inducer input traces generated using the toggle switch model M_3 using the parameter vector Θ_{TS}^{T1} in the computational study for Bayesian optimal experimental design for parameter inference.

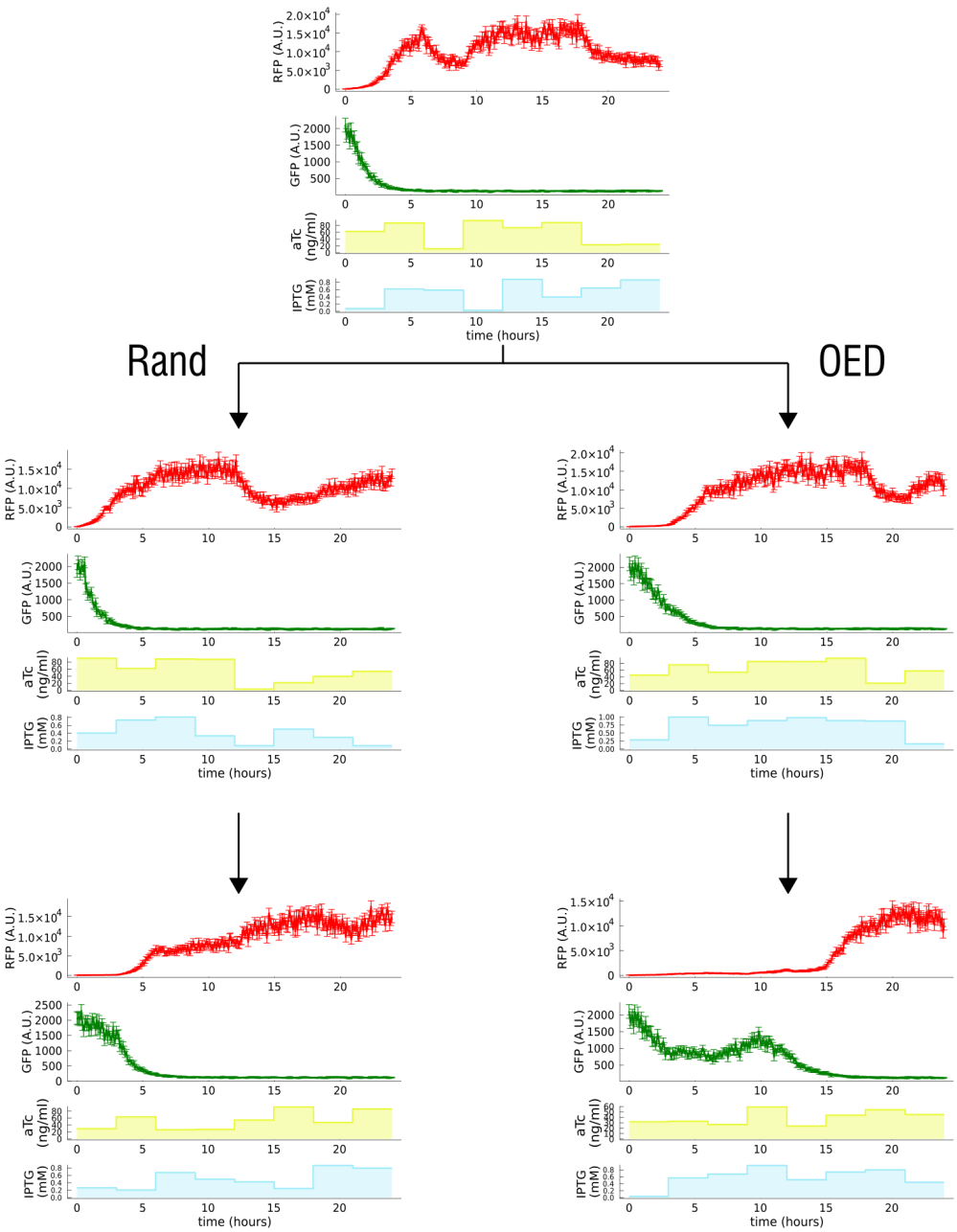


Figure S28: Pseudo-data and inducer input traces generated using the toggle switch model M_3 using the parameter vector Θ_{TS}^{T2} in the computational study for Bayesian optimal experimental design for parameter inference.

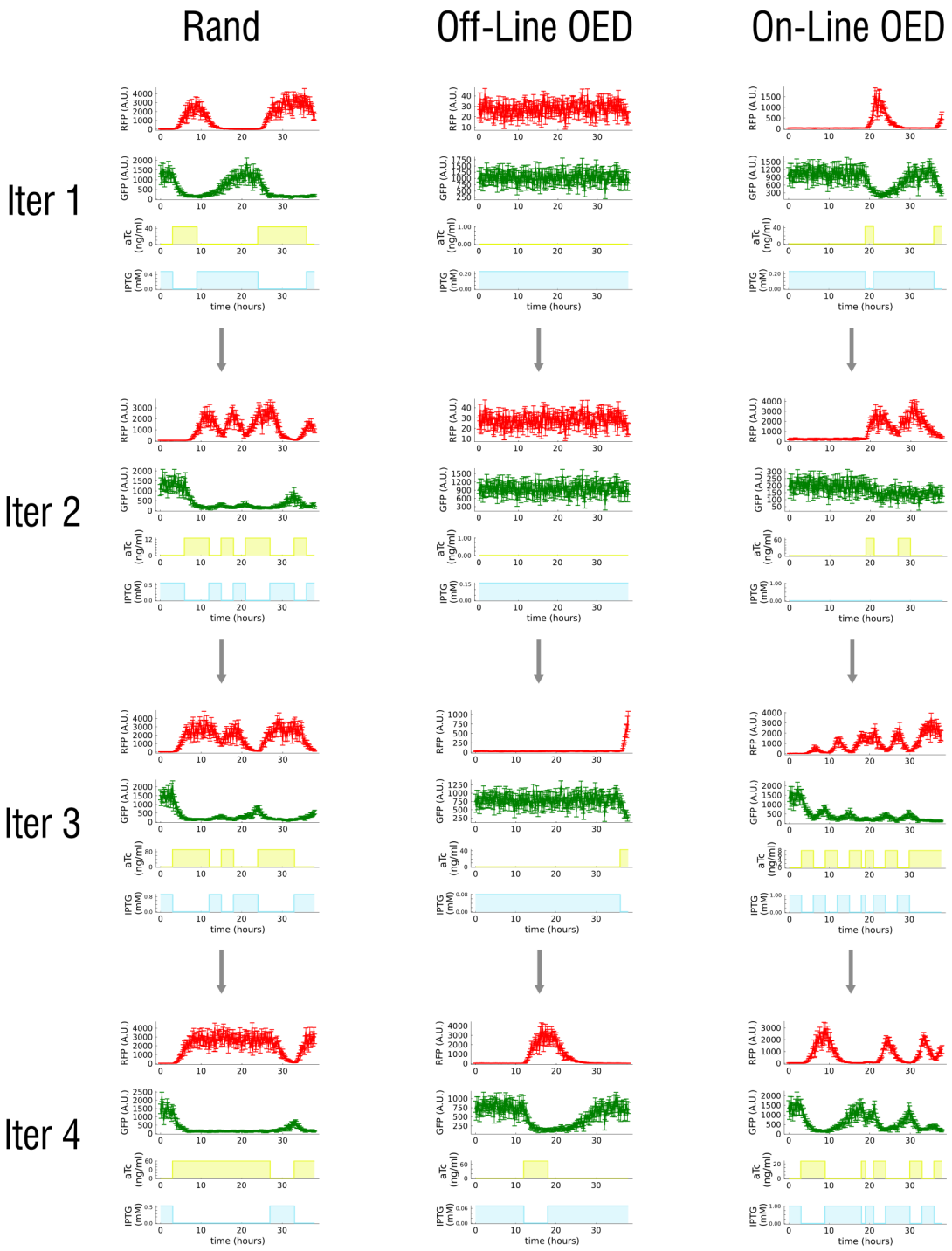


Figure S29: Pseudo-data and inducer input traces generated using the toggle switch model M_3 using the parameter vector Θ_{TS}^{T1} in the computational study for on-line Bayesian optimal experimental design for parameter inference.

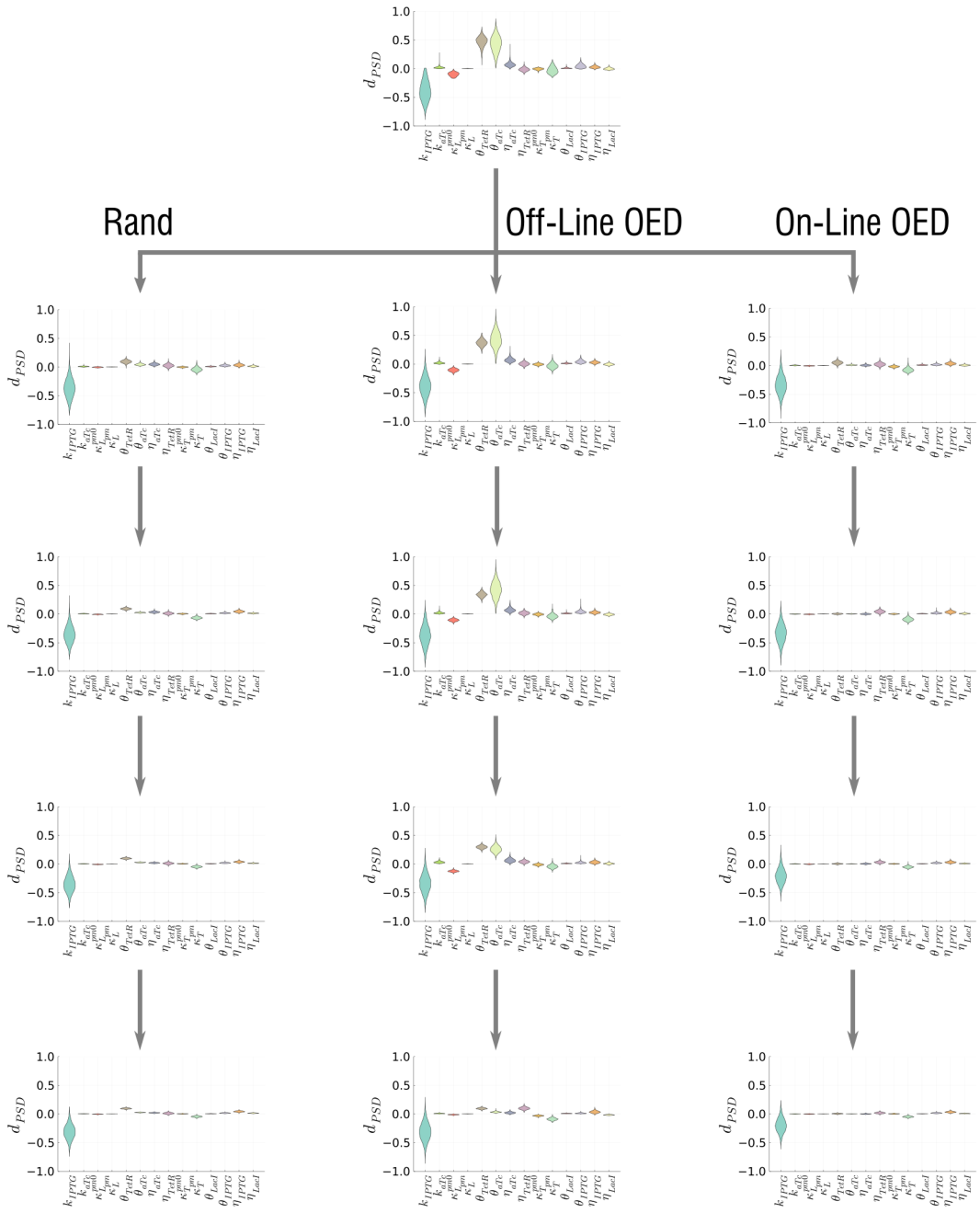


Figure S30: Posterior sample distances to Θ_{TS}^{T1} for each inferred parameter posterior obtained in the computational study for on-line Bayesian optimal experimental design for parameter inference.

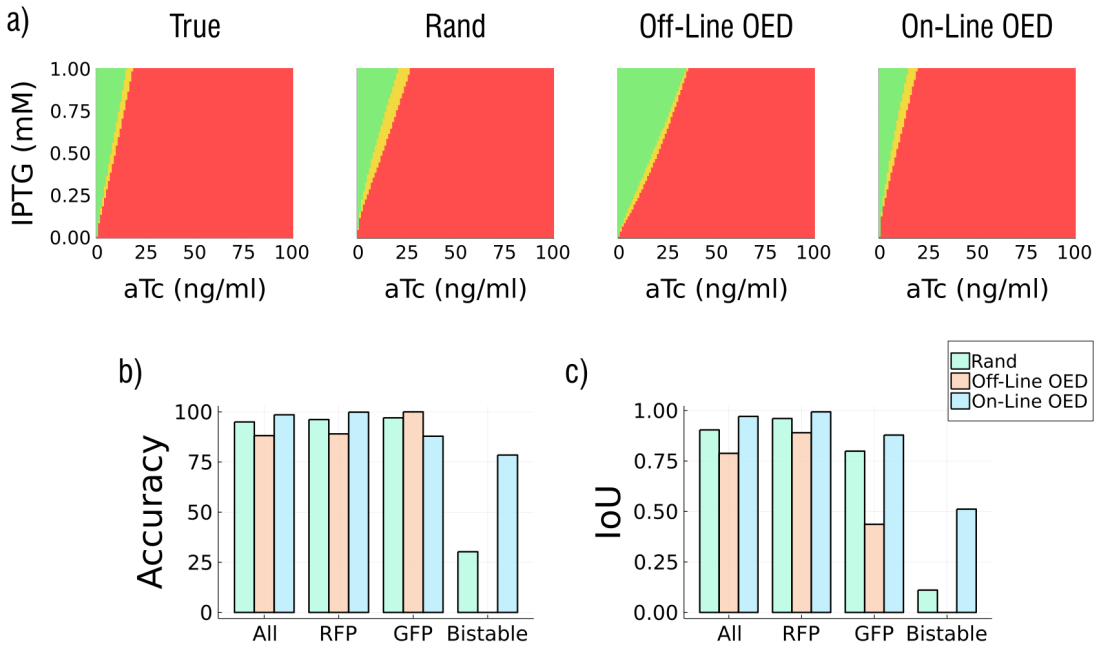


Figure S31: Stability analysis of the inferred parameter posteriors from the computational on-line OED for parameter inference study. a) Heat maps of the model M_3 stability matrices generated using Θ_{TS}^{T1} (True), the mean posterior parameter values after inference on random experiments (Rand) and the mean posterior parameter values after inference on off-line (Off-Line OED) and on-line (On-Line OED) optimal experiments as a function of aTc and IPTG. Colour green and red denotes monostability and the dominant fluorescent reporter at steady-state (GFP and RFP, respectively) while yellow colours indicate bistability. Prediction accuracy (b) and IoU (c) when considering the full matrices (All) or the individual regions (RFP, GFP, Bistable).

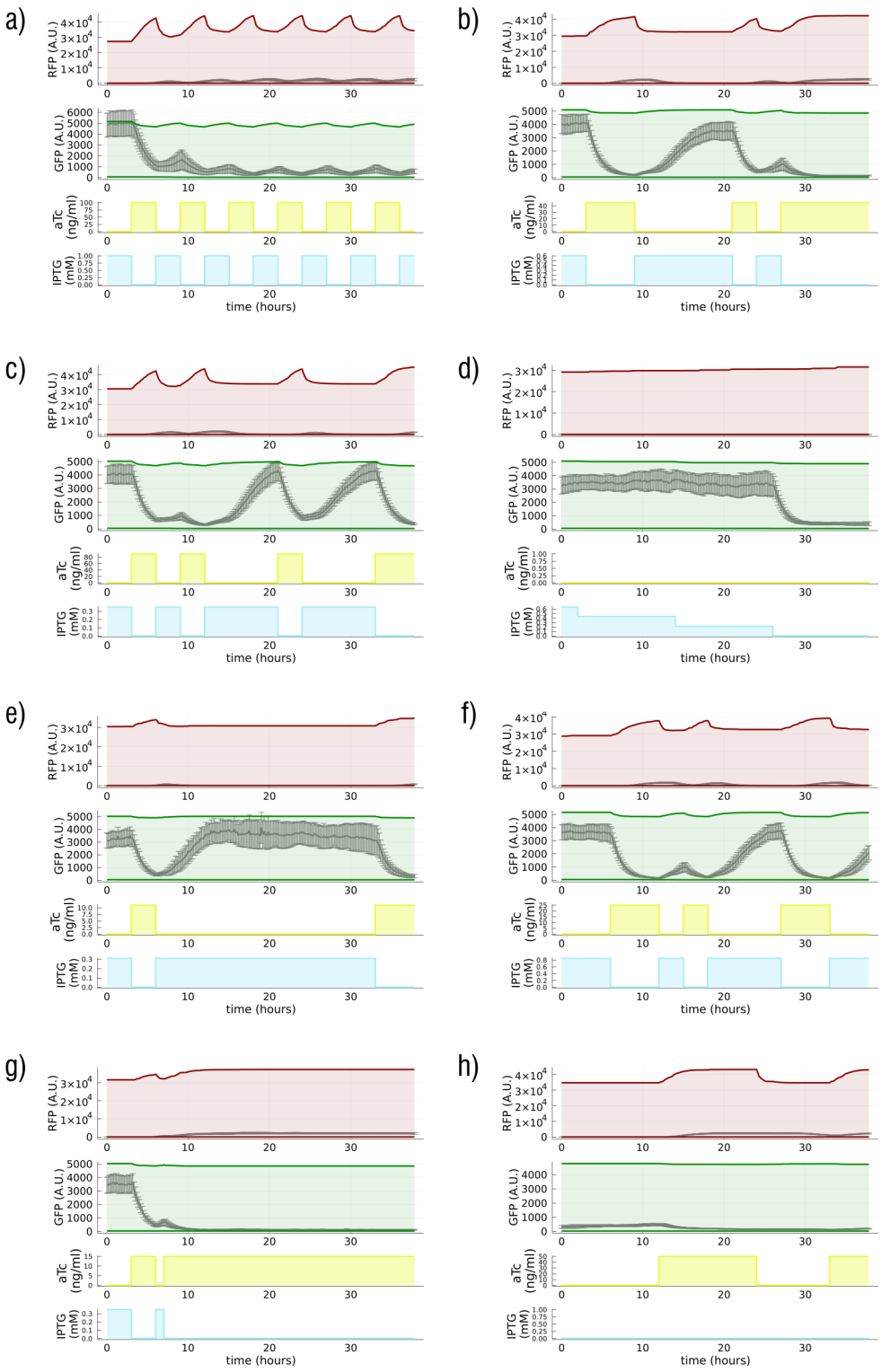


Figure S32: Toggle switch model M_3 prior predictions for the microfluidic experiments performed in this work. Prior predictions for the initial pulse-like experiment (a), the random experiments for iteration 2 (b) and iteration 3 (c), the optimal experiments for iteration 2 (d) and iteration 3 (e) and the three validation experiments (f,g,h).

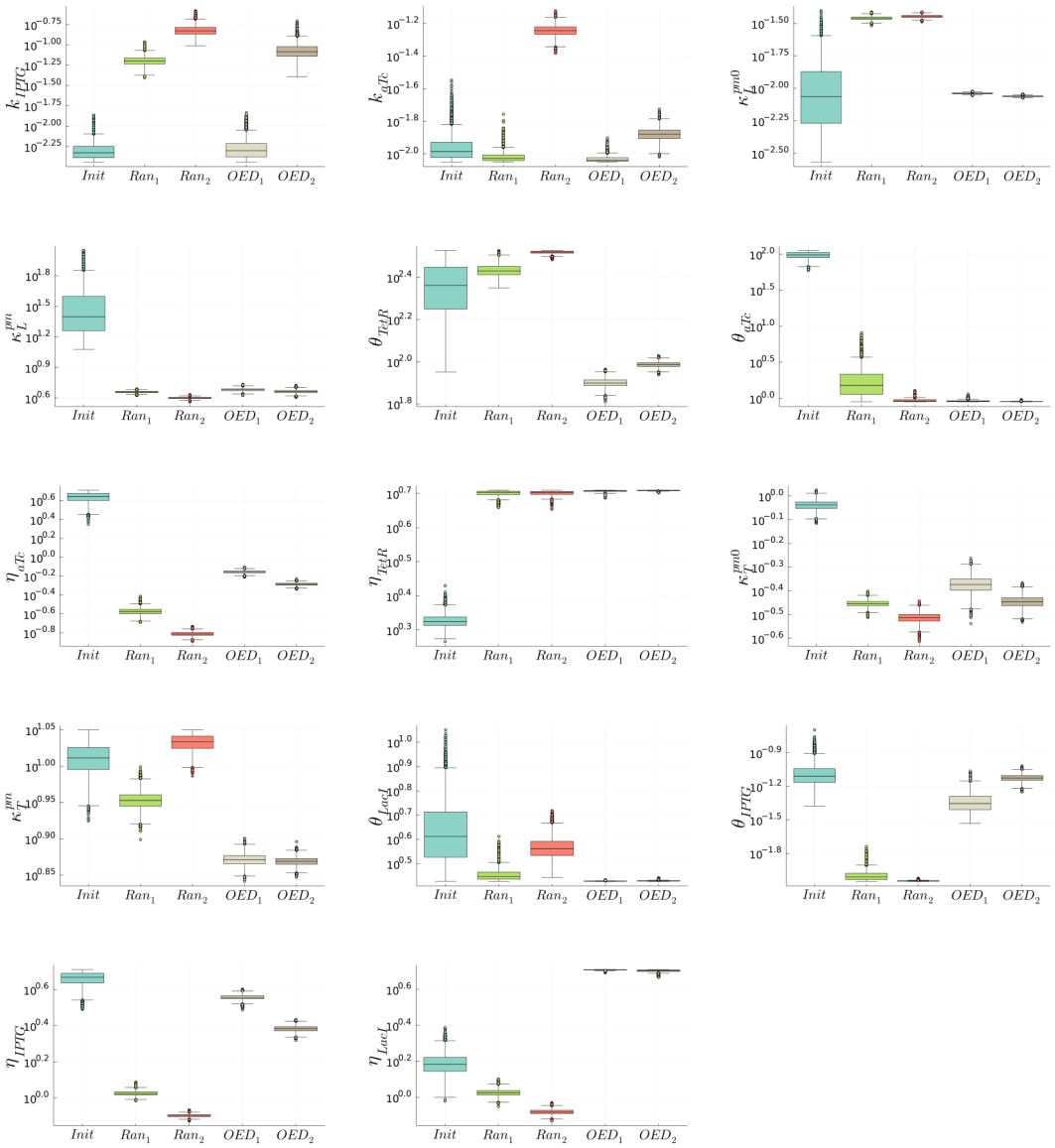


Figure S33: Posterior distributions for the mathematical model of the genetic toggle switch inferred from microfluidics experimental data. Box-plots for each inferred parameter posterior from the toggle switch model M_3 using a pulse-like experiment ($Init$), random experiments (Ran_1 for iteration 2 and Ran_2 for iteration 3) and optimally designed experiments (OED_1 for iteration 2 and OED_2 for iteration 3). Each inference used the uninformative priors described in Table 3.7 ($Prior_b$) performing a multi-experimental inference at every iteration.

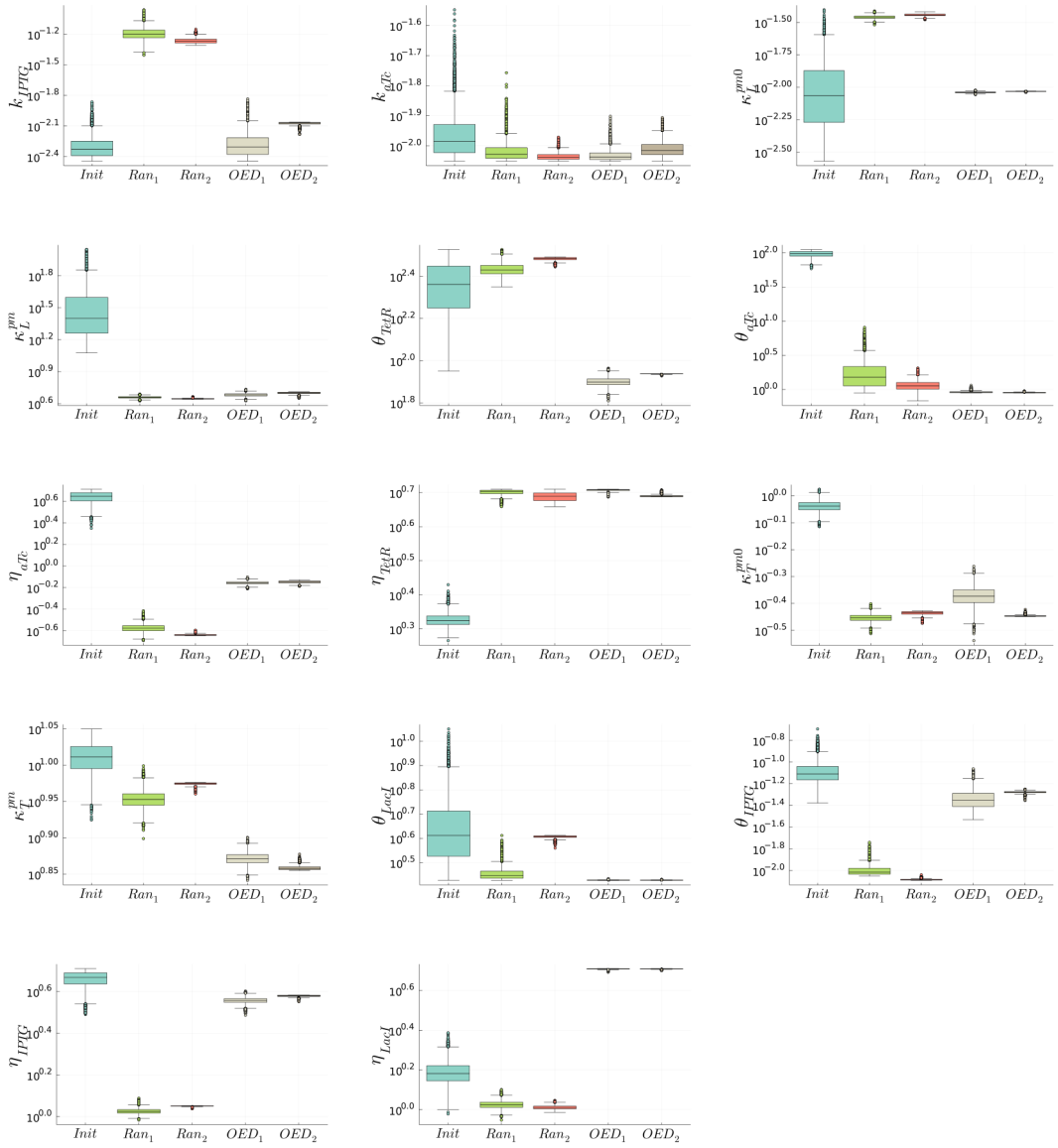


Figure S34: Posterior distributions for the mathematical model of the genetic toggle switch sequentially inferred from microfluidics experimental data. Box-plots for each inferred parameter posterior from the toggle switch model M_3 using a pulse-like experiment ($Init$), random experiments (Ran_1 for iteration 2 and Ran_2 for iteration 3) and optimally designed experiments (OED_1 for iteration 2 and OED_2 for iteration 3). Iteration 1 and 2 followed a multi-experimental inference using the uninformative priors described in Table 3.7 ($Prior_b$), while iteration 3 followed a single-experiment inference using as priors the posteriors obtained in iteration 2.

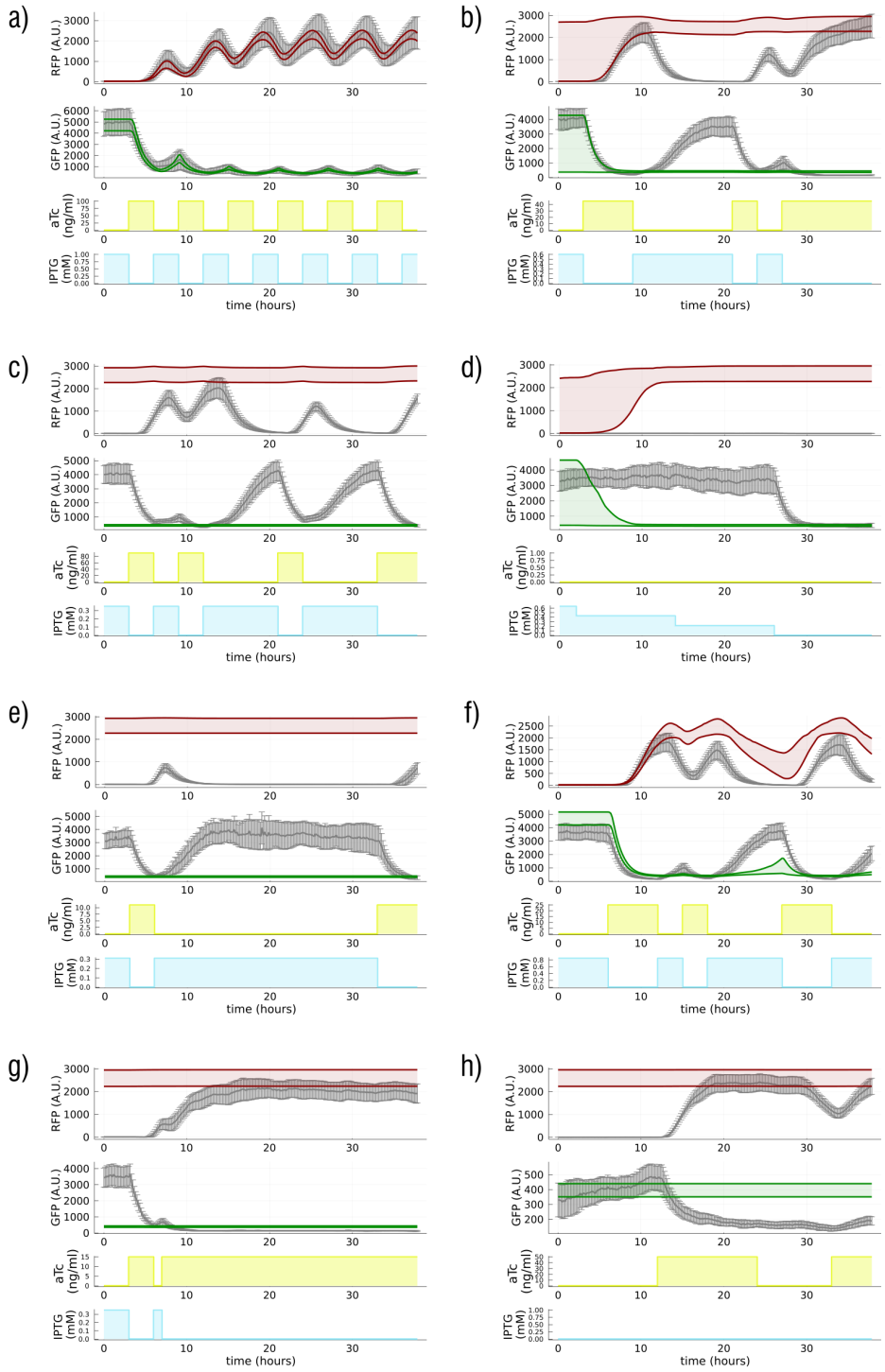


Figure S35: Toggle switch model M_3 predictions for the microfluidic experiments performed in this work using the parameter posterior obtained after inference using the initial pulse-like experiment. Posterior predictions for the initial pulse-like experiment (a), the random experiments for iteration 2 (b) and iteration 3 (c), the optimal experiments for iteration 2 (d) and iteration 3 (e) and the three validation experiments (f,g,h).

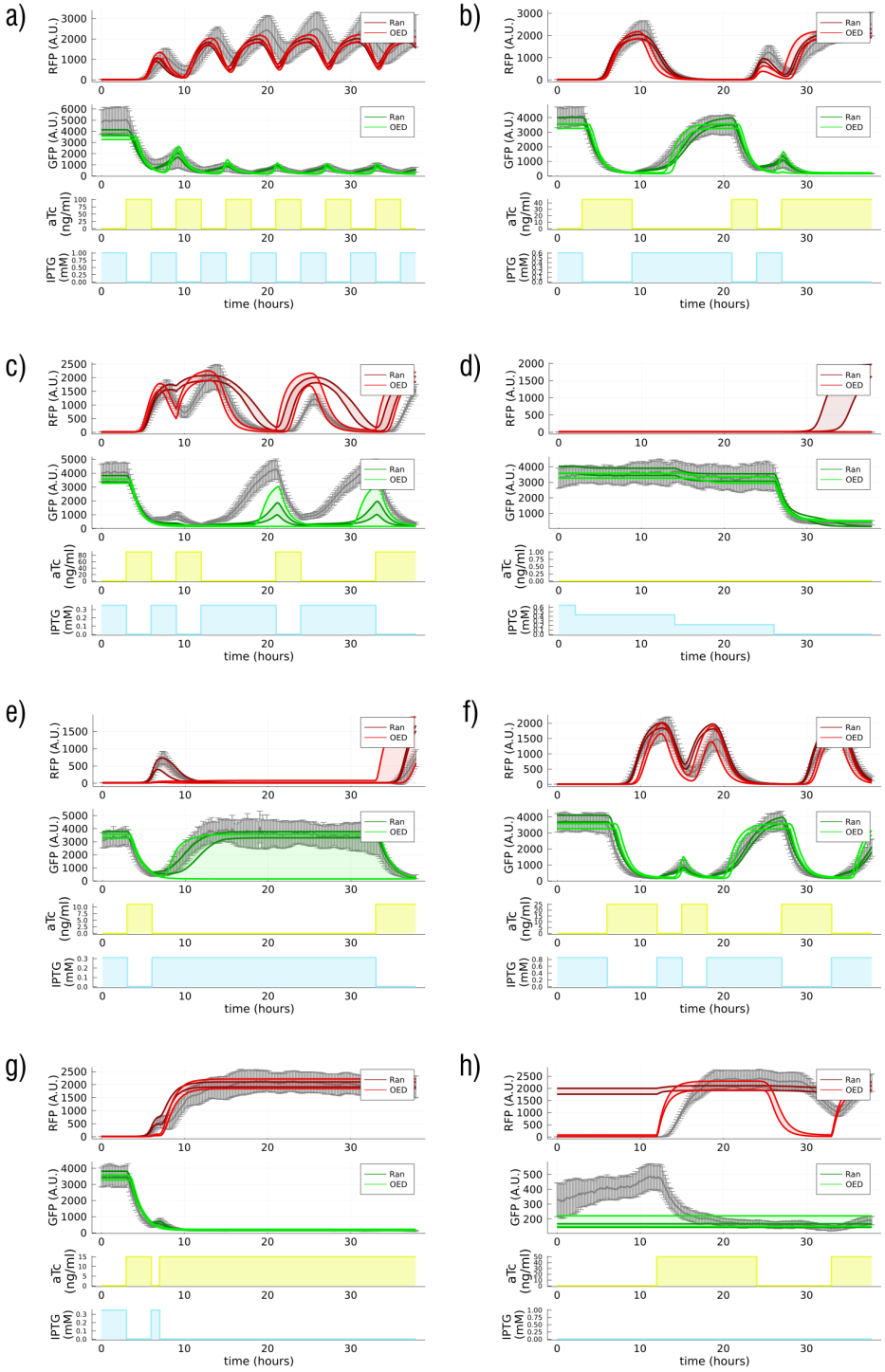


Figure S36: Toggle switch model M_3 predictions for the microfluidic experiments performed in this work using the parameter posteriors obtained in iteration 2. Posterior predictions for the initial pulse-like experiment (a), the random experiments for iteration 2 (b) and iteration 3 (c), the optimal experiments for iteration 2 (d) and iteration 3 (e) and the three validation experiments (f,g,h).

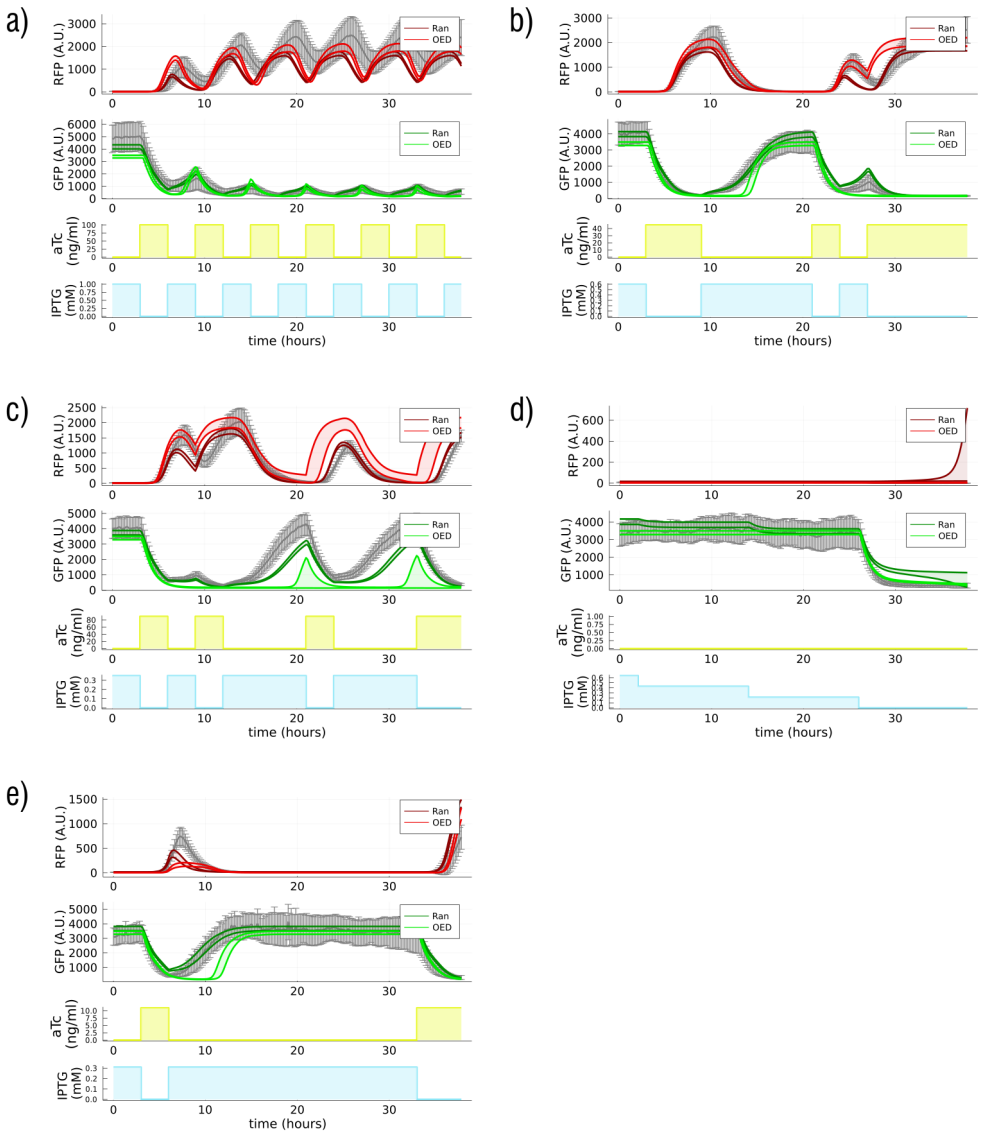


Figure S37: Toggle switch model M_3 predictions for the microfluidic experiments performed in this work using the parameter posteriors obtained in iteration 3 after multi-experimental inference using uninformative priors. Posterior predictions for the initial pulse-like experiment (a), the random experiments for iteration 2 (b) and iteration 3 (c) and the optimal experiments for iteration 2 (d) and iteration 3 (e).

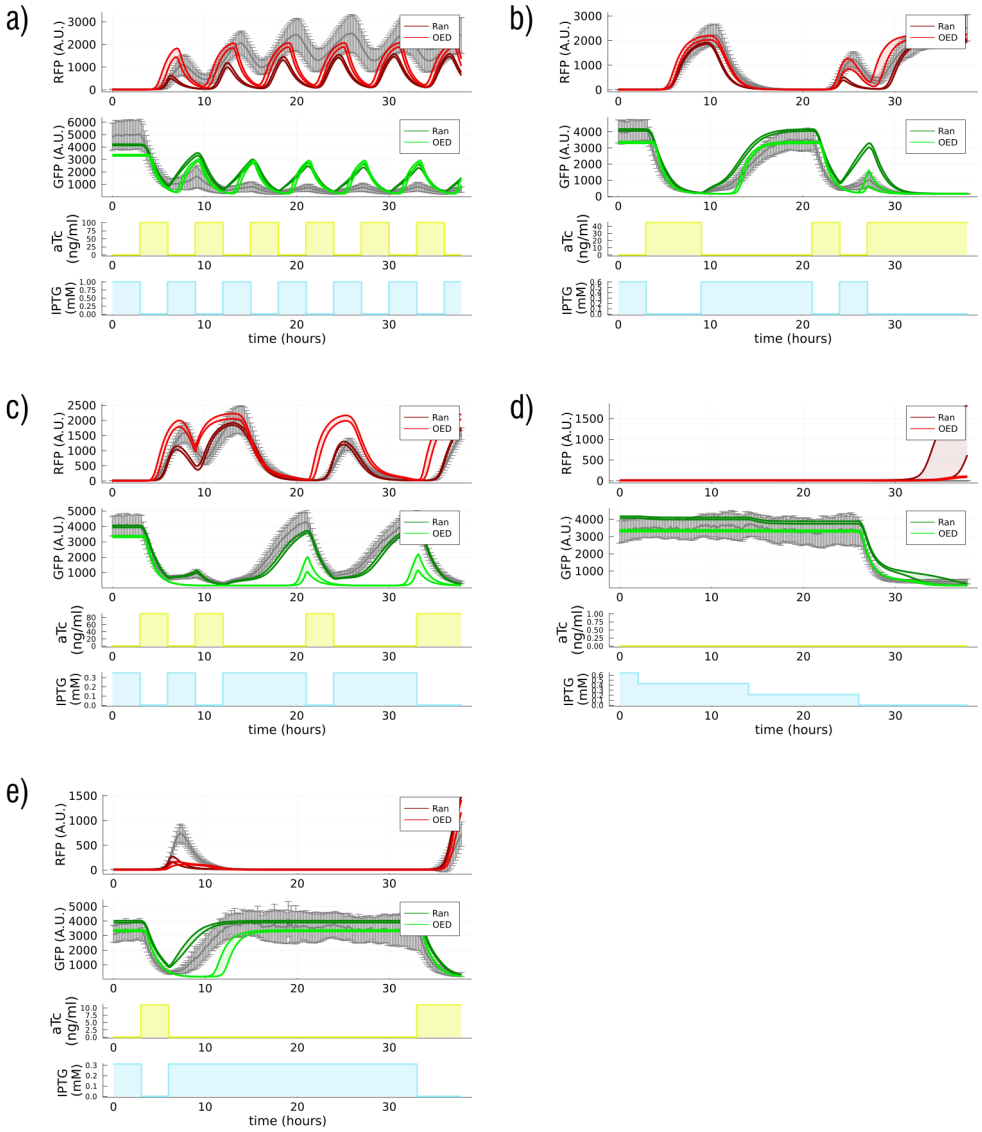


Figure S38: Toggle switch model M_3 predictions for the microfluidic experiments performed in this work using the parameter posteriors obtained in iteration 3 after a single-experimental inference using as priors the posteriors from iteration 2. Posterior predictions for the initial pulse-like experiment (a), the random experiments for iteration 2 (b) and iteration 3 (c) and the optimal experiments for iteration 2 (d) and iteration 3 (e).

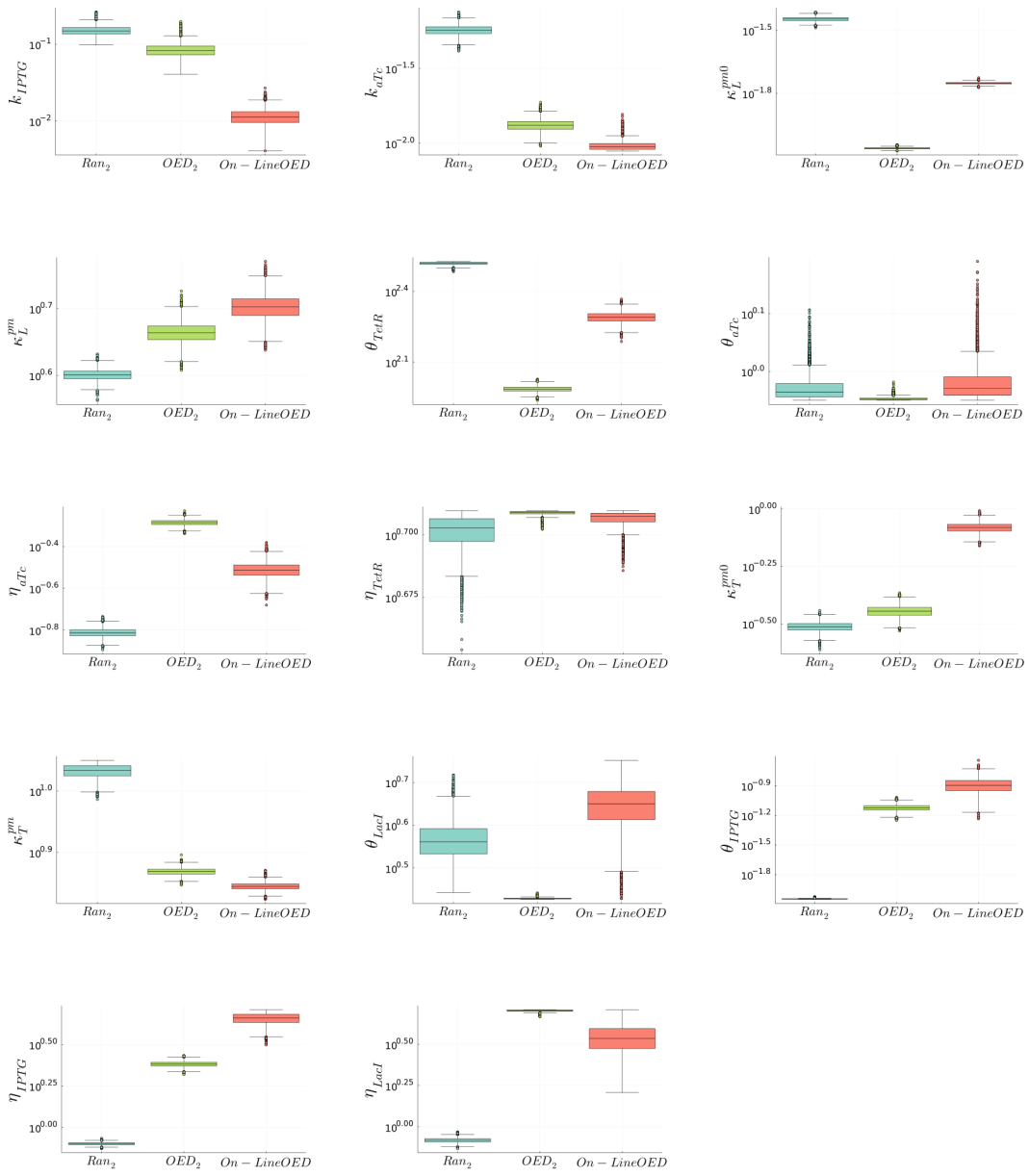


Figure S39: Posterior distribution comparison for the mathematical model of the genetic toggle switch inferred from microfluidics experimental data following random, off-line optimal and on-line optimal experimental designs. Box-plots for each inferred parameter posterior from the toggle switch model M_3 using random experiments (Ran_2 for iteration 3), off-line optimally designed experiments (OED_2 for iteration 3) and an on-line optimally designed one. Each inference used the uninformative priors described in Table 3.7 ($Prior_b$) performing a multi-experimental inference at every iteration including the initial pulse-like experiment in it.

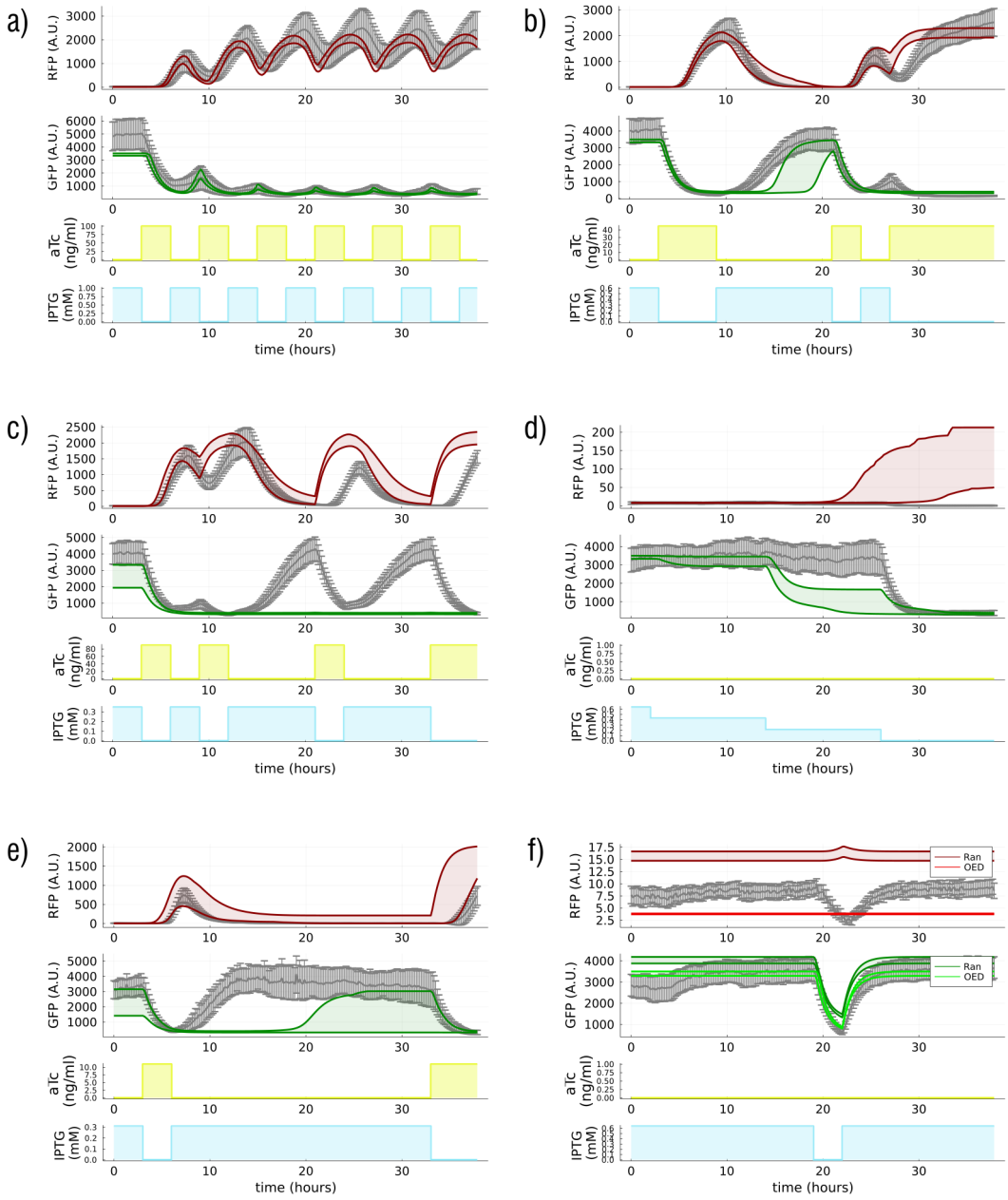


Figure S40: Toggle switch model M_3 predictions for the microfluidic experiments after inference including an on-line OED. Model predictions for the initial pulse-like experiment (a), the random experiments for iteration 2 (b) and iteration 3 (c), the optimal experiments for iteration 2 (d) and iteration 3 (e) using the posterior obtained after inference using the on-line OED. f) Model predictions of the on-line optimal experiment using the posteriors obtained after inference on random and off-line optimal experiments.

IPTG (mM)	aTc (ng/ml)	True	Rand	Off-Line OED	On-Line OED
1.0	100				
0.2	60				
1.0	90				
0.2	40				
0.2	10				
1.0	30				
0.9	40				
0.8	10				
0.2	90				
1.0	50				
0.1	70				
0.5	0				
0.0	30				
0.0	40				
0.8	90				
0.6	50				
0.6	30				
0.5	70				

Figure S41: Steady-state model validation analysis for the genetic toggle switch at iteration 2. Table containing 18 randomly selected IPTG and aTc concentration mixes between the allowed bounds (0-1 mM and 0-100 ng/ml, respectively), with the experimental steady-state for the system (True) and the predicted steady-state of model M_3 when using the mean of the posterior inferred from one random (Rand) and one optimal (off-line and on-line OED) experiments. Color red indicates a monostable RFP-LacI dominant state, green a monostable GFP-TetR dominant state and yellow bistability.

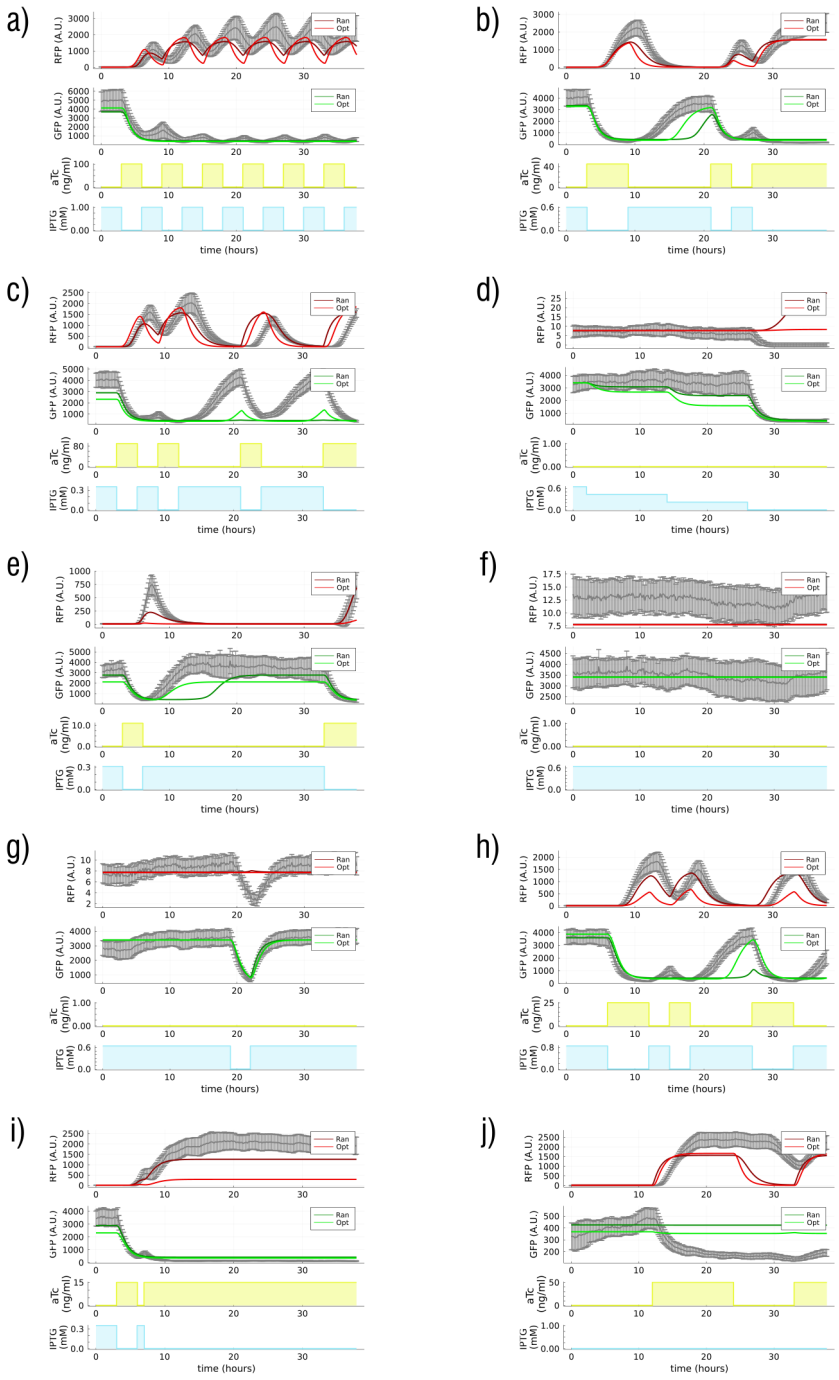


Figure S42: Model predictions of microfluidic after parameter MLE using the on-line OED with steady-state information as inference constraints. Model predictions using the best parameter vector according to MLE of the initial pulse-like experiment (a), the two random experiments used for Bayesian inference (b,c), the two optimal experiments used for Bayesian inference (d,e), the originally designed first OED (f), the on-line OED (g) and the three validation experiments (h,i,j). For MLE, the two models used the initial pulse-like and on-line OED data including randomly (Ran) or optimally (Opt) selected steady-state constraints.

IPTG (mM)	aTc (ng/ml)	True	Iter 1	Iter 2	Iter 3
1.0	100				
0.2	60				
1.0	90				
0.2	40				
0.2	10				
1.0	30				
0.9	40				
0.8	10				
0.2	90				
1.0	50				
0.1	70				
0.5	0				
0.0	30				
0.0	40				
0.8	90				
0.6	50				
0.6	30				
0.5	70				

Figure S43: Steady-state model validation analysis for the genetic toggle switch after MLE using randomly designed microfluidic experiments and randomly selected steady-state conditions as constraints. Table containing 18 randomly selected IPTG and aTc concentration mixes between the allowed bounds (0-1 mM and 0-100 ng/ml, respectively), with the experimental steady-state for the system (True) and the predicted steady-state when using the best parameter estimate according to MLE for the initial pulse-like experiment (Iter 1), the first (Iter 2) and the second random (Iter 3) experiments in an additive manner. Color red indicates a monostable RFP-LacI dominant state, green a monostable GFP-TetR dominant state and yellow bistability.

IPTG (mM)	aTc (ng/ml)	True	Iter 1	Iter 2	Iter 3
1.0	100				
0.2	60				
1.0	90				
0.2	40				
0.2	10				
1.0	30				
0.9	40				
0.8	10				
0.2	90				
1.0	50				
0.1	70				
0.5	0				
0.0	30				
0.0	40				
0.8	90				
0.6	50				
0.6	30				
0.5	70				

Figure S44: Steady-state model validation analysis for the genetic toggle switch after MLE using optimally designed microfluidic experiments and randomly selected steady-state conditions as constraints. Table containing 18 randomly selected IPTG and aTc concentration mixes between the allowed bounds (0-1 mM and 0-100 ng/ml, respectively), with the experimental steady-state for the system (True) and the predicted steady-state when using the best parameter estimate according to MLE for the initial pulse-like experiment (Iter 1), the first (Iter 2) and the second off-line optimal (Iter 3) experiments in an additive manner. Color red indicates a monostable RFP-LacI dominant state, green a monostable GFP-TetR dominant state and yellow bistability.

IPTG (mM)	aTc (ng/ml)	True	Iter 1	Iter 2	Iter 3
1.0	100				
0.2	60				
1.0	90				
0.2	40				
0.2	10				
1.0	30				
0.9	40				
0.8	10				
0.2	90				
1.0	50				
0.1	70				
0.5	0				
0.0	30				
0.0	40				
0.8	90				
0.6	50				
0.6	30				
0.5	70				

Figure S45: Steady-state model validation analysis for the genetic toggle switch after MLE using randomly designed microfluidic experiments and optimally selected steady-state conditions as constraints. Table containing 18 randomly selected IPTG and aTc concentration mixes between the allowed bounds (0-1 mM and 0-100 ng/ml, respectively), with the experimental steady-state for the system (True) and the predicted steady-state when using the best parameter estimate according to MLE for the initial pulse-like experiment (Iter 1), the first (Iter 2) and the second random (Iter 3) experiments in an additive manner. Color red indicates a monostable RFP-LacI dominant state, green a monostable GFP-TetR dominant state and yellow bistability.

IPTG (mM)	aTc (ng/ml)	True	Iter 1	Iter 2	Iter 3
1.0	100				
0.2	60				
1.0	90				
0.2	40				
0.2	10				
1.0	30				
0.9	40				
0.8	10				
0.2	90				
1.0	50				
0.1	70				
0.5	0				
0.0	30				
0.0	40				
0.8	90				
0.6	50				
0.6	30				
0.5	70				

Figure S46: Steady-state model validation analysis for the genetic toggle switch after MLE using optimally designed microfluidic experiments and optimally selected steady-state conditions as constraints. Table containing 18 randomly selected IPTG and aTc concentration mixes between the allowed bounds (0-1 mM and 0-100 ng/ml, respectively), with the experimental steady-state for the system (True) and the predicted steady-state when using the best parameter estimate according to MLE for the initial pulse-like experiment (Iter 1), the first (Iter 2) and the second off-line optimal (Iter 3) experiments in an additive manner. Color red indicates a monostable RFP-LacI dominant state, green a monostable GFP-TetR dominant state and yellow bistability.

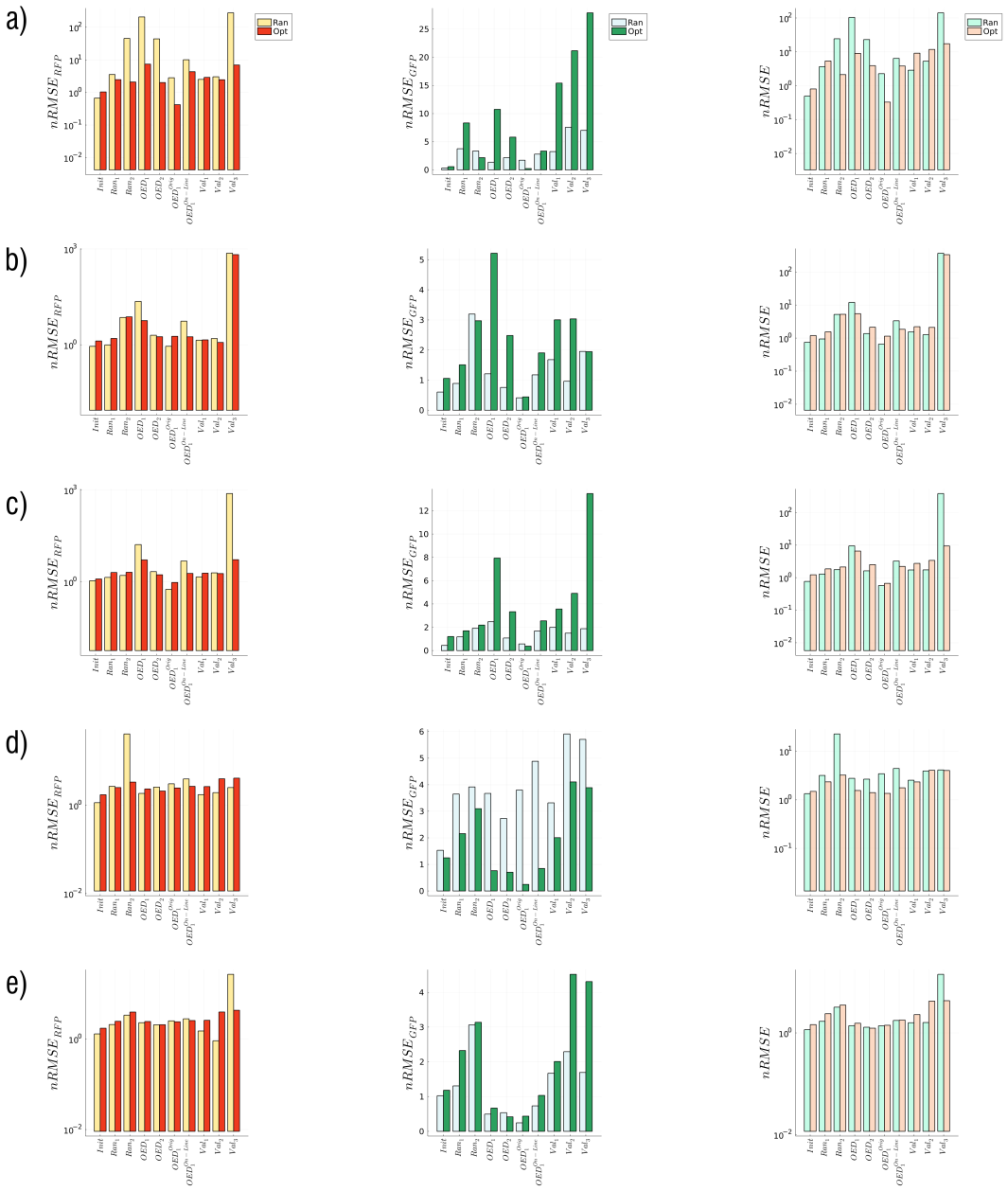


Figure S47: nRMSE of the genetic toggle switch model using the best parameter estimate according to MLE including steady-state observable rations as constraints when predicting microfluidic experiments. nRMSE for RFP-LacI (left), GFP-TetR (middle) or the mean of both observables (right) including randomly (Ran) or optimally (Opt) selected steady-state constraints after parameter estimation using the initial pulse-like microfluidic experiment (a), the first (b) and second (c) random microfluidic experiments (in an additive manner) and the first (d) and second (e) off-line optimal microfluidic experiments (in an additive manner).

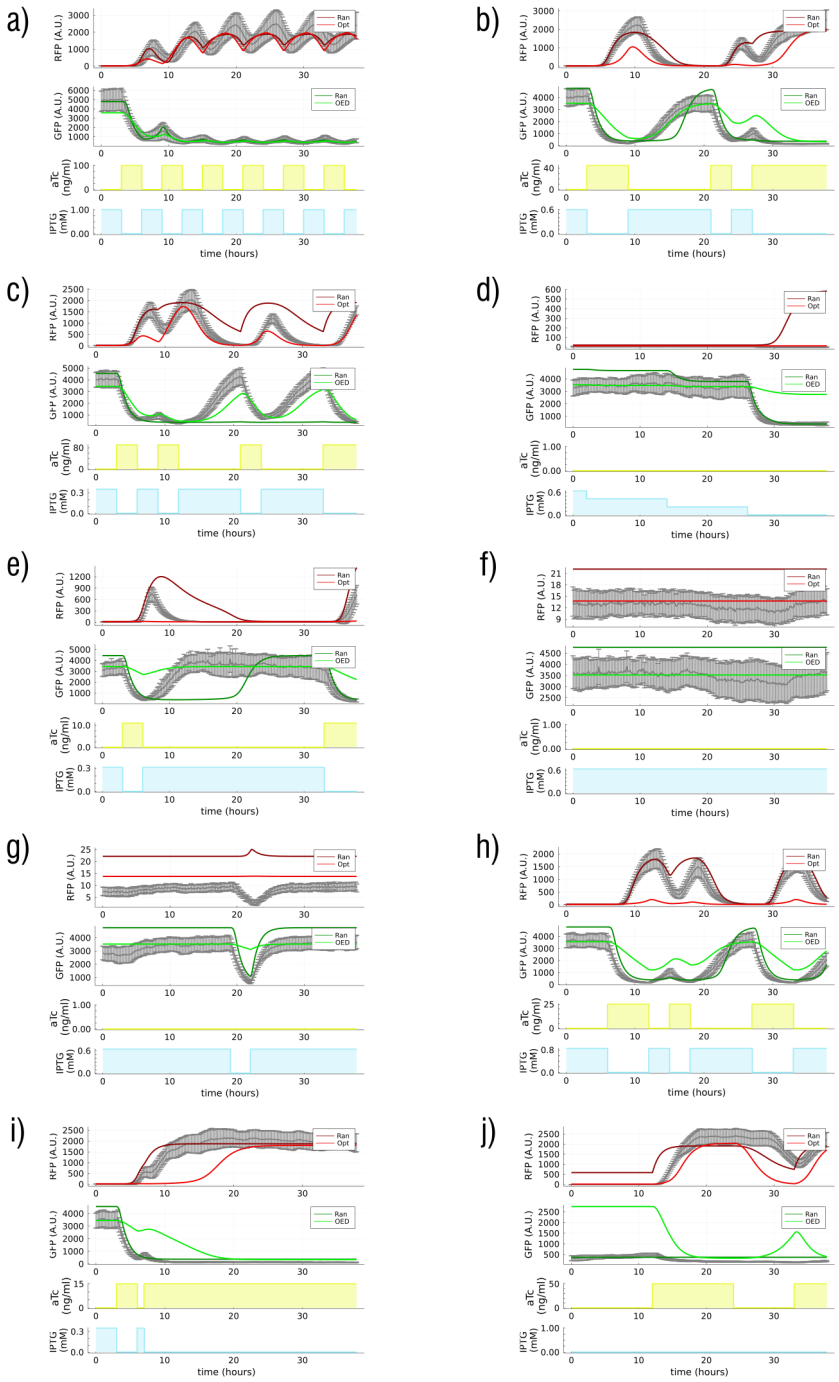


Figure S48: Model predictions of microfluidic experiments after parameter MLE using the initial pulse-like experiment with steady-state information as inference constraints. Model predictions of the initial pulse-like experiment (a), the two random experiments used for Bayesian inference (b,c), the two optimal experiments used for Bayesian inference (d,e), the originally designed first OED (f), the on-line OED (g) and the three validation experiments (h,i,j). For MLE the two models used the initial pulse-like experiment including random (Ran) or optimal (Opt) steady-state constraints.

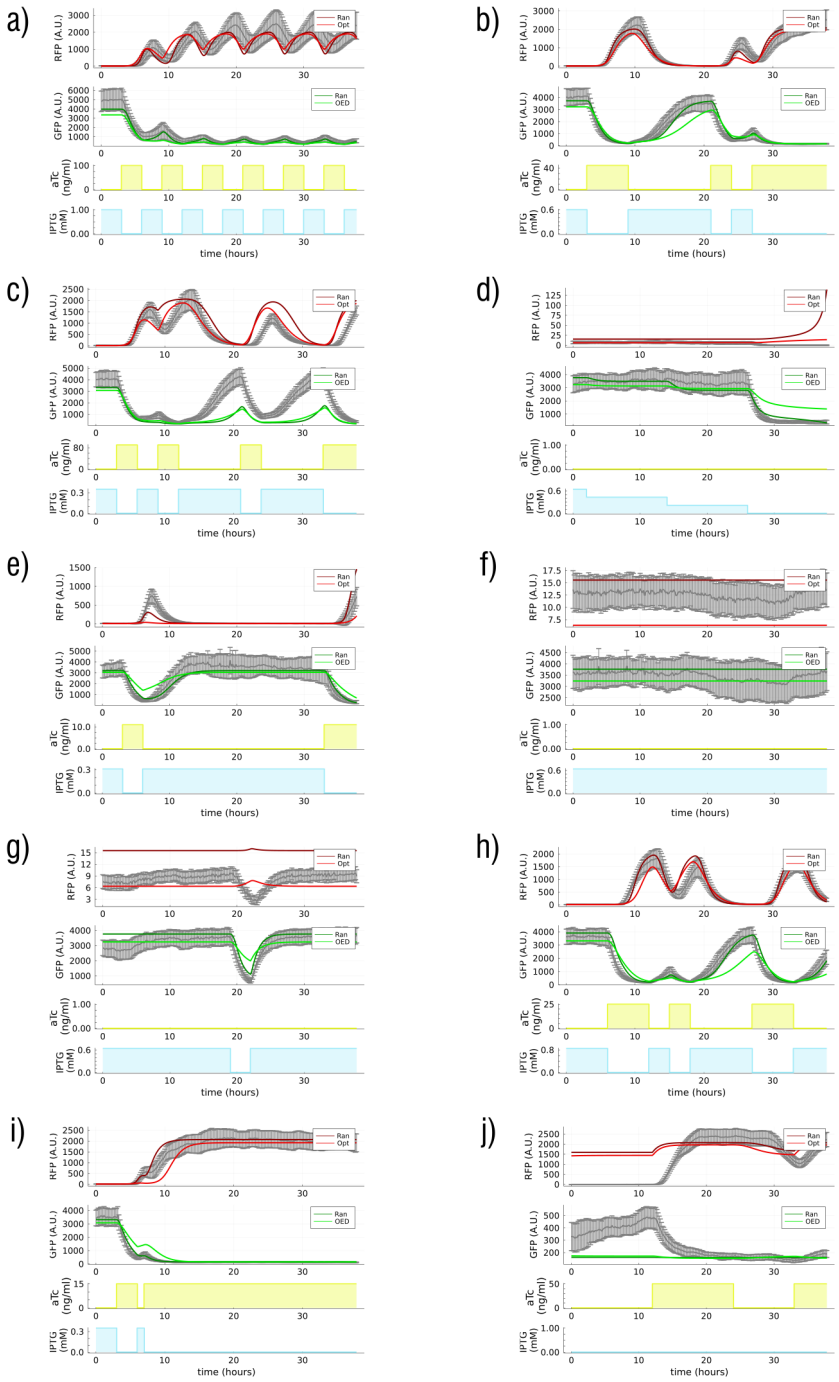


Figure S49: Model predictions of microfluidic experiments after parameter MLE using the first random experiment with steady-state information as inference constraints. Model predictions of the initial pulse-like experiment (a), the two random experiments used for Bayesian inference (b,c), the two optimal experiments used for Bayesian inference (d,e), the originally designed first OED (f), the on-line OED (g) and the three validation experiments (h,i,j). For MLE the two models used the initial pulse-like and first random experiments including random (Ran) or optimal (Opt) steady-state constraints.

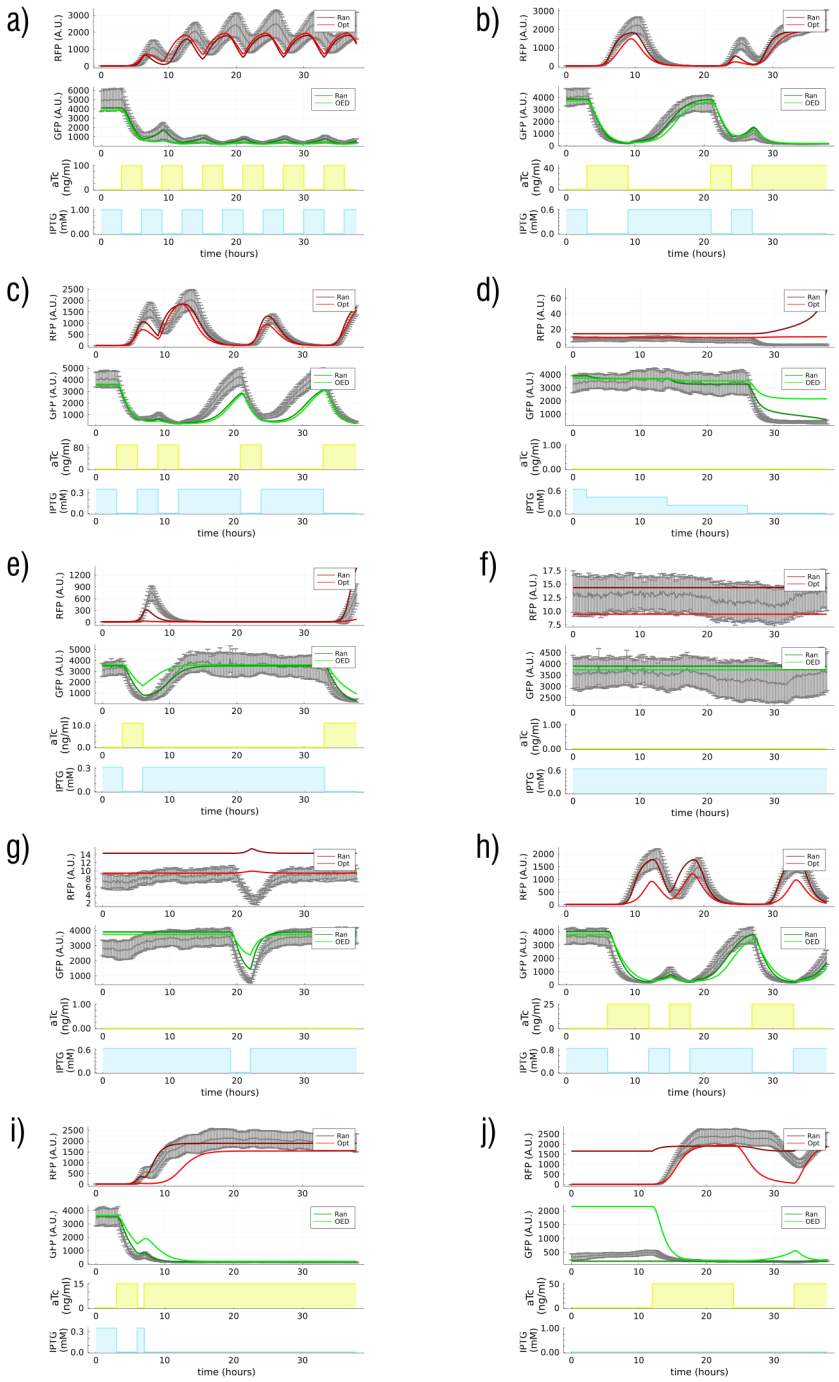


Figure S50: Model predictions of microfluidic experiments after parameter MLE using the first and second random experiments with steady-state information as inference constraints. Model predictions of the initial pulse-like experiment (a), the two random experiments used for Bayesian inference (b,c), the two optimal experiments used for Bayesian inference (d,e), the originally designed first OED (f), the on-line OED (g) and the three validation experiments (h,i,j). For MLE the two models used the initial pulse-like, first and second random experiments including random (Ran) or optimal (Opt) steady-state constraints.

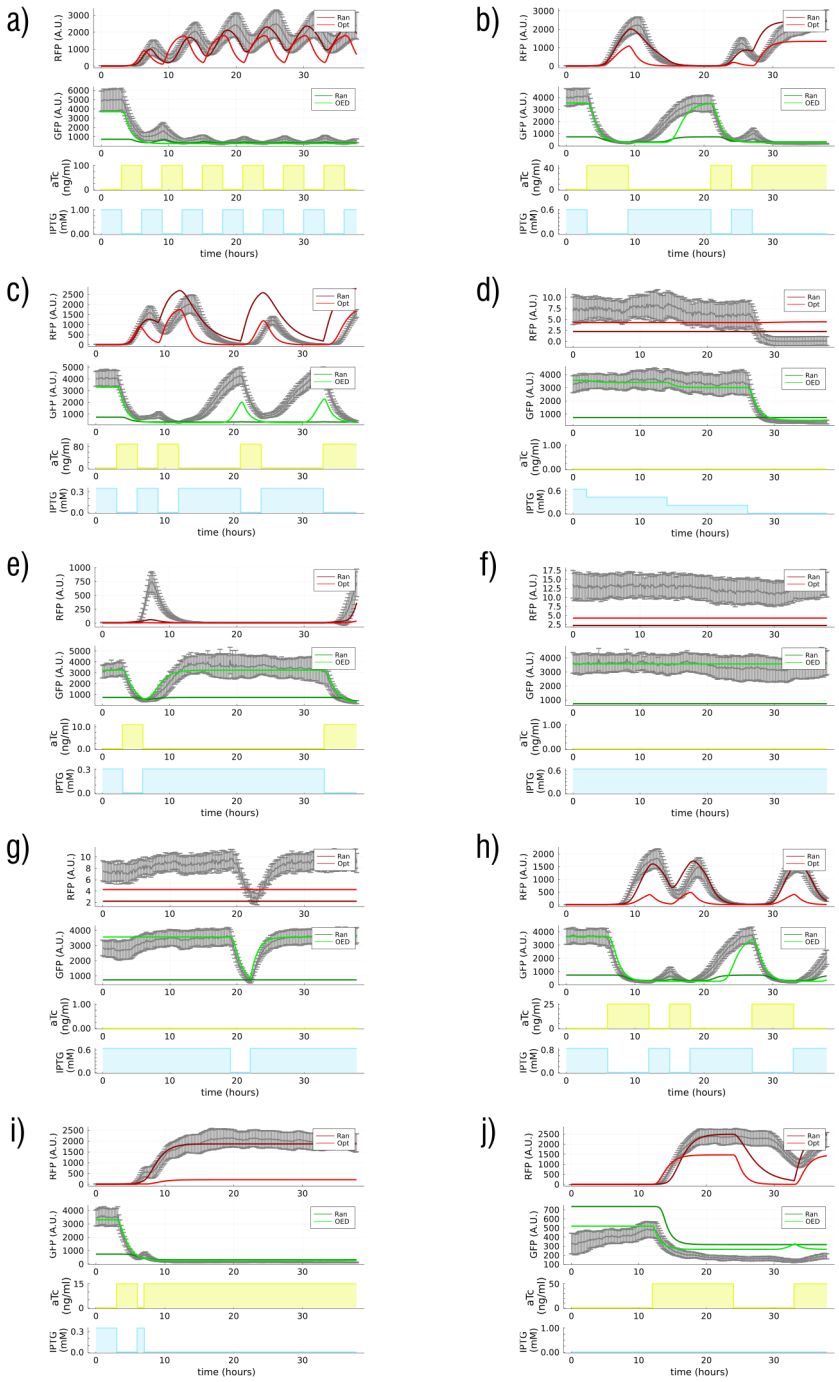


Figure S51: Model predictions of microfluidic experiments after parameter MLE using the first off-line optimal experiment with steady-state information as inference constraints. Model predictions of the initial pulse-like experiment (a), the two random experiments used for Bayesian inference (b,c), the two optimal experiments used for Bayesian inference (d,e), the originally designed first OED (f), the on-line OED (g) and the three validation experiments (h,i,j). For MLE the two models used the initial pulse-like and first optimal experiments including random (Ran) or optimal (Opt) steady-state constraints.

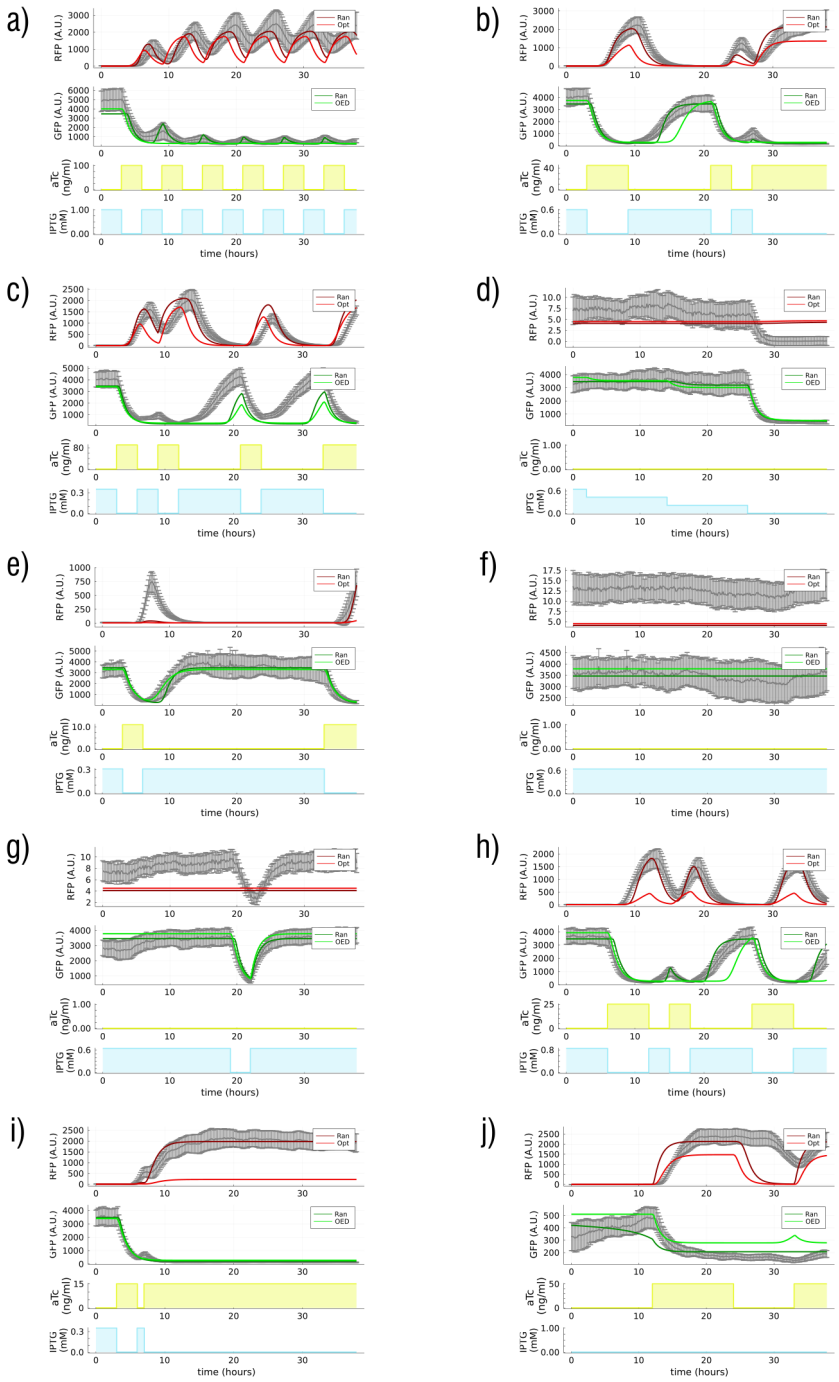


Figure S52: Model predictions of microfluidic experiments after parameter MLE using the first and second off-line optimal experiments with steady-state information as inference constraints. Model predictions of the initial pulse-like experiment (a), the two random experiments used for Bayesian inference (b,c), the two optimal experiments used for Bayesian inference (d,e), the originally designed first OED (f), the on-line OED (g) and the three validation experiments (h,i,j). For MLE the two models used initial the pulse-like, first and second optimal experiments including random (Ran) or optimal (Opt) steady-state constraints.

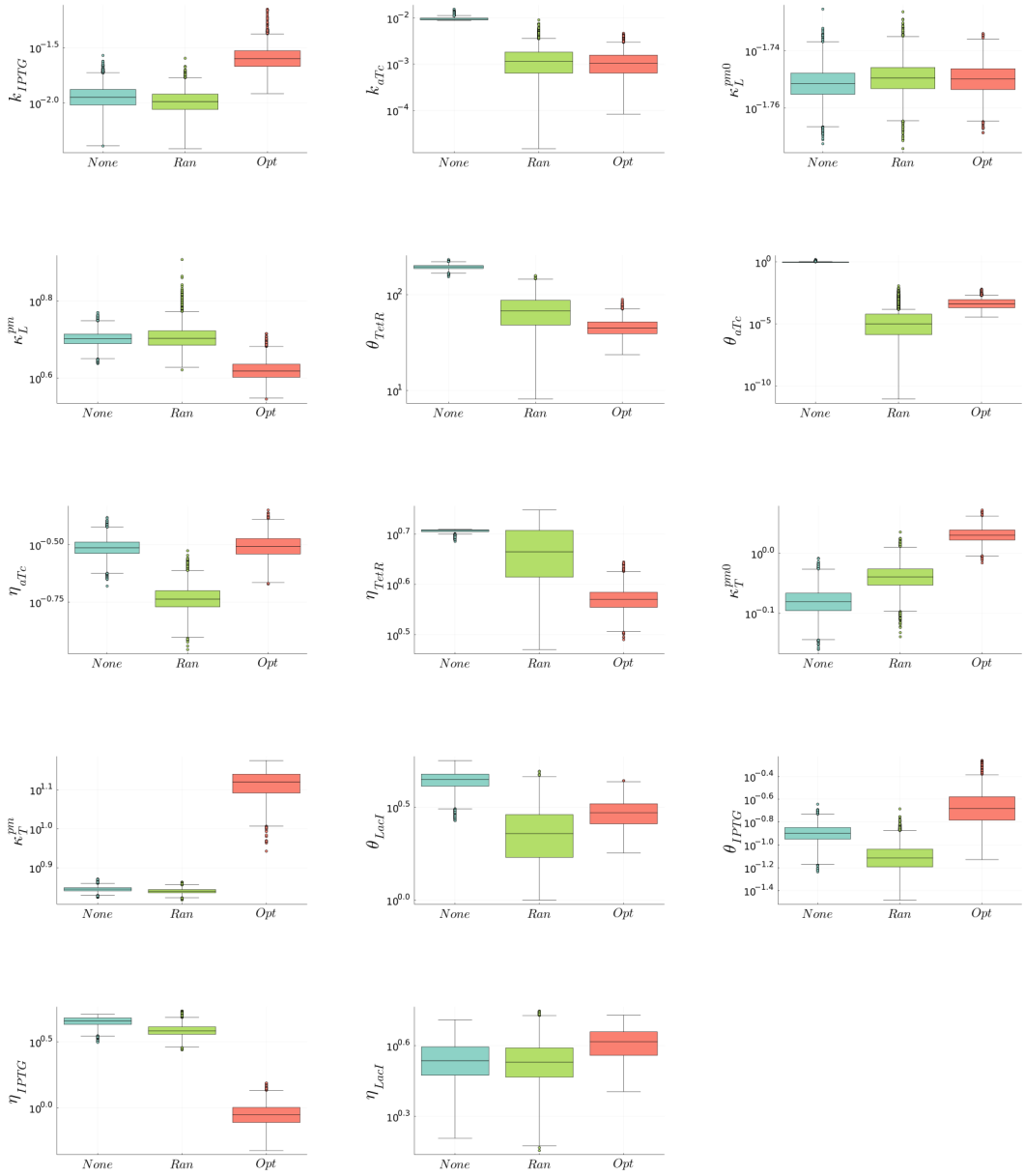


Figure S53: Posterior distribution comparison for a mathematical model of the genetic toggle switch inferred using the on-line OED microfluidics data including randomly or optimally selected steady-state observable ratios as constraints. Box-plots for each inferred parameter posterior from the toggle switch model M_3 where the inference included randomly (Ran) or optimally (Opt) selected steady-state constraints or excluded them (None). Each inference used the uninformative priors described in Table 3.7 ($Prior_b$) performing a multi-experimental inference including the initial pulse-like and the on-line optimal experiments in it.

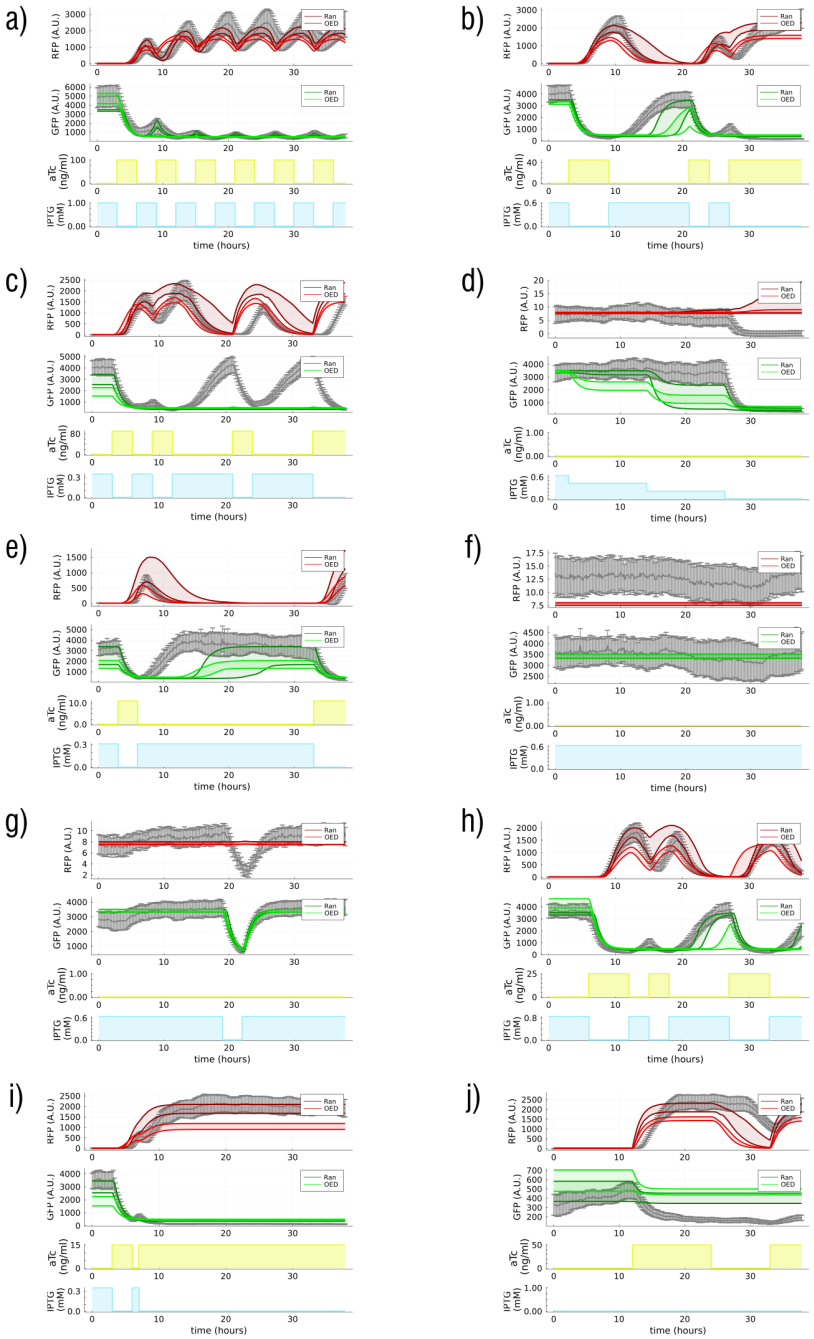


Figure S54: Model predictions of microfluidic experiments after Bayesian inference using the on-line OED with steady-state observable ratio information as inference constraints. Model predictions indicated as confidence intervals from predictions using randomly sampled posterior vectors for the initial pulse-like experiment (a), the two random experiments used for Bayesian inference (b,c), the two optimal experiments used for Bayesian inference (d,e), the originally designed first OED (f), the on-line OED (g) and the three validation experiments (h,i,j). For inference, the two models used initial pulse-like and the on-line OED data including randomly (Ran) or optimally (Opt) selected steady-state constraints.

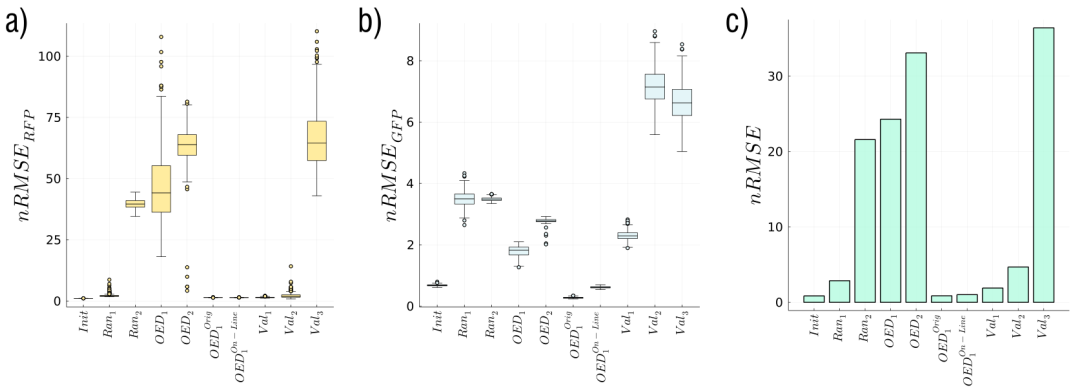


Figure S55: Model validation analysis for a model describing the genetic toggle switch employing microfluidics data for Bayesian inference without using observable ratios as constraints. Box plots containing the nRMSE for model M_3 using posterior samples to predict RFP-LacI (a) and GFP-TetR (b) and the average nRMSE (c) between posterior prediction samples and observables. Both inferences used the initial pulse-like experiment and the on-line OED microfluidics data. For the optimally designed microfluidic experiments, OED_1 is the model-based modified one, and OED_1^{Orig} is the one without modifications (output from the optimisation scheme).

Bibliography

[1] Allison J Lopatkin and James J Collins. Predictive biology: modelling, understanding and harnessing microbial complexity. *Nature Reviews Microbiology*, 18(9):507–520, 2020.

[2] Jeff Gauthier, Antony T Vincent, Steve J Charette, and Nicolas Derome. A brief history of bioinformatics. *Briefings in bioinformatics*, 20(6):1981–1996, 2019.

[3] Bruno César Feltes, Joice de Faria Poloni, Itamar José Guimarães Nunes, Sara Socorro Faria, and Marcio Dorn. Multi-approach bioinformatics analysis of curated omics data provides a gene expression panorama for multiple cancer types. *Frontiers in genetics*, page 1354, 2020.

[4] Han-Yu Chuang, Matan Hofree, and Trey Ideker. A decade of systems biology. *Annual review of cell and developmental biology*, 26:721–744, 2010.

[5] Alan Aderem. Systems biology: its practice and challenges. *Cell*, 121(4):511–513, 2005.

[6] David Angeli, James E Ferrell, and Eduardo D Sontag. Detection of multistability, bifurcations, and hysteresis in a large class of biological positive-feedback systems. *Proceedings of the National Academy of Sciences*, 101(7):1822–1827, 2004.

[7] James E Ferrell Jr, Joseph R Pomerening, Sun Young Kim, Nikki B Trunnell, Wen Xiong, Chi-Ying Frederick Huang, and Eric M Machleder. Simple, realistic models of complex biological processes: positive feedback and bistability in a cell fate switch and a cell cycle oscillator. *FEBS letters*, 583(24):3999–4005, 2009.

[8] Ahmad S Khalil and James J Collins. Synthetic biology: applications come of age. *Nature Reviews Genetics*, 11(5):367–379, 2010.

[9] Timothy S Gardner, Charles R Cantor, and James J Collins. Construction of a genetic toggle switch in escherichia coli. *Nature*, 403(6767):339–342, 2000.

[10] Michael B Elowitz and Stanislas Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338, 2000.

[11] M Omar Din, Tal Danino, Arthur Prindle, Matt Skalak, Jangir Selimkhanov, Kaitlin Allen, Ellixis Julio, Eta Atolia, Lev S Tsimring, Sangeeta N Bhatia, et al. Synchronized cycles of bacterial lysis for in vivo delivery. *Nature*, 536(7614):81–85, 2016.

[12] Sang Tae Park and Jayoung Kim. Trends in next-generation sequencing and a new era for whole genome sequencing. *International neurourology journal*, 20(Suppl 2):S76, 2016.

[13] Claudia Gonzaga-Jauregui, James R Lupski, and Richard A Gibbs. Human genome sequencing in health and disease. *Annual review of medicine*, 63:35–61, 2012.

[14] Jeff W Lichtman and José-Angel Conchello. Fluorescence microscopy. *Nature methods*, 2(12):910–919, 2005.

[15] Armen R Kherlopian, Ting Song, Qi Duan, Mathew A Neimark, Ming J Po, John K Gohagan, and Andrew F Laine. A review of imaging techniques for systems biology. *BMC systems biology*, 2(1):1–18, 2008.

[16] Ertugrul M Ozbudak, Mukund Thattai, Han N Lim, Boris I Shraiman, and Alexander Van Oudenaarden. Multistability in the lactose utilization network of escherichia coli. *Nature*, 427(6976):737–740, 2004.

[17] Robert M Hoffman. The multiple uses of fluorescent proteins to visualize cancer in vivo. *Nature Reviews Cancer*, 5(10):796–806, 2005.

[18] Dmitriy M Chudakov, Mikhail V Matz, Sergey Lukyanov, and Konstantin A Lukyanov. Fluorescent proteins and their applications in imaging living cells and tissues. *Physiological reviews*, 90(3):1103–1163, 2010.

[19] Xiaowei Yan, Tim A Hoek, Ronald D Vale, and Marvin E Tanenbaum. Dynamics of translation of single mrna molecules in vivo. *Cell*, 165(4):976–989, 2016.

[20] Xianjun Chen, Dasheng Zhang, Ni Su, Bingkun Bao, Xin Xie, Fangting Zuo, Lipeng Yang, Hui Wang, Li Jiang, Qiuning Lin, et al. Visualizing rna dynamics in live cells with bright and stable fluorescent rnas. *Nature Biotechnology*, 37(11):1287–1293, 2019.

[21] Nathan Braniff and Brian Ingalls. New opportunities for optimal design of dynamic experiments in systems and synthetic biology. *Current Opinion in Systems Biology*, 9:42–48, 2018.

[22] Marc Rullan, Dirk Benzinger, Gregor W Schmidt, Andreas Milias-Argeitis, and Mustafa Khammash. An optogenetic platform for real-time, single-cell interrogation of stochastic transcriptional regulation. *Molecular cell*, 70(4):745–756, 2018.

[23] Jing Dai, Sung Ho Yoon, Hye Young Sim, Yoon Sun Yang, Tae Kwang Oh, Jihyun F Kim, and Jong Wook Hong. Charting microbial phenotypes in multiplex nanoliter batch bioreactors. *Analytical chemistry*, 85(12):5892–5899, 2013.

[24] Todd A Duncombe, Augusto M Tentori, and Amy E Herr. Microfluidics: reframing biological enquiry. *Nature Reviews Molecular Cell Biology*, 16(9):554–567, 2015.

[25] Emily E Kempa, Katherine A Hollywood, Clive A Smith, and Perdita E Barran. High throughput screening of complex biological samples with mass spectrometry–from bulk measurements to single cell analysis. *Analyst*, 144(3):872–891, 2019.

[26] Kyung-Jin Jang, Monicah A Otieno, Janey Ronxhi, Heng-Keang Lim, Lorna Ewart, Konstantia R Kodella, Debora B Petropolis, Gauri Kulkarni, Jonathan E Rubins, David Conegliano, et al. Reproducing human and cross-species drug toxicities using a liver-chip. *Science translational medicine*, 11(517):eaax5516, 2019.

[27] Claire J Tomlin and Jeffrey D Axelrod. Biology by numbers: mathematical modelling in developmental biology. *Nature reviews genetics*, 8(5):331–340, 2007.

[28] Alec AK Nielsen, Bryan S Der, Jonghyeon Shin, Prashant Vaidyanathan, Vanya Paralanov, Elizabeth A Strychalski, David Ross, Douglas Densmore,

and Christopher A Voigt. Genetic circuit design automation. *Science*, 352(6281):aac7341, 2016.

[29] Alfio Quarteroni. Mathematical models in science and engineering. *Notices of the AMS*, 56(1):10–19, 2009.

[30] GK Batchelor. *An introduction to fluid dynamics*. Cambridge university press, 2000.

[31] Siegfried Flügge. *Practical quantum mechanics*. Springer Science & Business Media, 2012.

[32] JR Kittrell. Mathematical modeling of chemical reactions. In *Advances in Chemical Engineering*, volume 8, pages 97–183. Elsevier, 1970.

[33] G. A. Hazelrigg. On the Role and Use of Mathematical Models in Engineering Design. *Journal of Mechanical Design*, 121(3):336–341, 09 1999.

[34] Mike Gillman. *An introduction to mathematical models in ecology and evolution: time and space*, volume 4. John Wiley & Sons, 2009.

[35] D Mary Celin Sharmila, T Praveen, and L Rajendran. Mathematical modeling and analysis of nonlinear enzyme catalyzed reaction processes. *Journal of Theoretical Chemistry*, 2013, 2013.

[36] Fred Brauer, Carlos Castillo-Chavez, and Zhilan Feng. *Mathematical models in epidemiology*, volume 32. Springer, 2019.

[37] D Chandran, WB Copeland, SC Sleight, and HM Sauro. Mathematical modeling and synthetic biology. *Drug Discovery Today: Disease Models*, 5(4):299–309, 2008.

[38] Kate Harline, Jesús Martínez-Gómez, Chelsea D Specht, and Adrienne HK Roeder. A life cycle for modeling biology at different scales. *Frontiers in Plant Science*, page 1724, 2021.

[39] Jeremy Gunawardena. Models in biology:'accurate descriptions of our pathetic thinking'. *BMC biology*, 12(1):1–11, 2014.

[40] Santo Motta and Francesco Pappalardo. Mathematical modeling of biological systems. *Briefings in Bioinformatics*, 14(4):411–422, 2013.

[41] Brian P Ingalls. *Mathematical modeling in systems biology: an introduction*. MIT press, 2013.

[42] Haluk Resat, Linda Petzold, and Michel F. Pettigrew. *Kinetic Modeling of Biological Systems*, pages 311–335. Humana Press, Totowa, NJ, 2009.

[43] Bruce Hannon and Matthias Ruth. *Modeling Dynamic Biological Systems*, pages 3–28. Springer International Publishing, Cham, 2014.

[44] Andreas Deutsch, Lutz Brusch, Helen Byrne, Gerda De Vries, and Hanspeter Herzel. *Mathematical Modeling of Biological Systems, Volume I.* Springer, 2007.

[45] Vijayalakshmi Chelliah, Camille Laibe, and Nicolas Le Novère. *BioModels Database: A Repository of Mathematical Models of Biological Processes*, pages 189–199. Humana Press, Totowa, NJ, 2013.

[46] Rafael S Costa, André Veríssimo, and Susana Vinga. Ki mosys: a web-based repository of experimental data for kinetic models of biological systems. *BMC systems biology*, 8(1):1–10, 2014.

[47] M. T. Cooling, V. Rouilly, G. Misirli, J. Lawson, T. Yu, J. Hallinan, and A. Wipat. Standard virtual biological parts: a repository of modular modeling components for synthetic biology. *Bioinformatics*, 26(7):925–931, 02 2010.

[48] Michael Hucka, Andrew Finney, Herbert M Sauro, Hamid Bolouri, John C Doyle, Hiroaki Kitano, Adam P Arkin, Benjamin J Bornstein, Dennis Bray, Athel Cornish-Bowden, et al. The systems biology markup language (sbml): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003.

[49] Sebastian Mirschel, Katrin Steinmetz, Michael Rempel, Martin Ginkel, and Ernst Dieter Gilles. ProMoT: modular modeling for systems biology. *Bioinformatics*, 25(5):687–689, 01 2009.

[50] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the national academy of sciences*, 113(15):3932–3937, 2016.

[51] Gabriele Lillacci and Mustafa Khammash. Parameter estimation and model selection in computational biology. *PLOS Computational Biology*, 6(3):1–17, 03 2010.

[52] Eric-Jan Wagenmakers, Michael Lee, Tom Lodewyckx, and Geoffrey J. Iverson. *Bayesian Versus Frequentist Inference*, pages 181–207. Springer New York, New York, NY, 2008.

[53] Frank DeVilbiss and Doraiswami Ramkrishna. Addressing the need for a model selection framework in systems biology using information theory. *Proceedings of the IEEE*, 105(2):330–339, 2017.

[54] Paul Kirk, Thomas Thorne, and Michael PH Stumpf. Model selection in systems and synthetic biology. *Current opinion in biotechnology*, 24(4):767–774, 2013.

[55] James Gilman, Laura Walls, Lucia Bandiera, and Filippo Menolascina. Statistical design of experiments for synthetic biology. *ACS Synthetic Biology*, 10(1):1–18, 2021.

[56] Juliane Liepe, Sarah Filippi, Michał Komorowski, and Michael PH Stumpf. Maximizing the information content of experiments in systems biology. *PLoS computational biology*, 9(1):e1002888, 2013.

[57] Eva Balsa-Canto, Antonio A Alonso, and Julio R Banga. Computational procedures for optimal experimental design in biological systems. *IET systems biology*, 2(4):163–172, 2008.

[58] Lucia Bandiera, David Gomez-Cabeza, James Gilman, Eva Balsa-Canto, and Filippo Menolascina. Optimally designed model selection for synthetic biology. *ACS Synthetic Biology*, 9(11):3134–3144, 2020.

[59] Byran Smucker, Martin Krzywinski, and Naomi Altman. Optimal experimental design. *Nat. Methods*, 15(8):559–560, 2018.

[60] Luc Pronzato. Optimal experimental design and some related control problems. *Automatica*, 44(2):303–325, 2008.

[61] Valerii Fedorov. Optimal experimental design. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(5):581–589, 2010.

[62] J. Kiefer. Optimum experimental designs. *Journal of the Royal Statistical Society. Series B (Methodological)*, 21(2):272–319, 1959.

[63] Peter Whittle. Some general points in the theory of optimal experimental design. *Journal of the Royal Statistical Society: Series B (Methodological)*, 35(1):123–130, 1973.

[64] A Philip Dawid and Paola Sebastiani. Coherent dispersion criteria for optimal experimental design. *Annals of Statistics*, pages 65–81, 1999.

[65] Jose López-Fidalgo, Ch Tommasi, and PC Trandafir. An optimal experimental design criterion for discriminating between non-normal models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(2):231–242, 2007.

[66] Anthony C. Atkinson and Randall D. Tobias. Optimal experimental design in chromatography. *Journal of Chromatography A*, 1177(1):1–11, 2008.

[67] Lorenz Fleitmann, Jan Pyschik, Ludger Wolff, Johannes Schilling, and André Bardow. Optimal experimental design of physical property measurements for optimal chemical process simulations. *Fluid Phase Equilibria*, 557:113420, 2022.

[68] Roozbeh Dehghannasiri, Dezhen Xue, Prasanna V. Balachandran, Mohammadmahdi R. Yousefi, Lori A. Dalton, Turab Lookman, and Edward R. Dougherty. Optimal experimental design for materials discovery. *Computational Materials Science*, 129:311–322, 2017.

[69] Julien Berger, Denys Dutykh, and Nathan Mendes. On the optimal experiment design for heat and moisture parameter estimation. *Experimental Thermal and Fluid Science*, 81:109–122, 2017.

[70] Joep Vanlier, Christian A Tiemann, Peter AJ Hilbers, and Natal AW van Riel. Optimal experiment design for model selection in biochemical networks. *BMC systems biology*, 8(1):1–16, 2014.

[71] Xiaokang Wang, Navneet Rai, Beatriz Merchel Piovesan Pereira, Ameen Eetemadi, and Ilias Tagkopoulos. Accelerated knowledge discovery from omics data by optimal experimental design. *Nature communications*, 11(1):1–9, 2020.

[72] Eva Balsa-Canto, Lucia Bandiera, and Filippo Menolascina. Optimal experimental design for systems and synthetic biology using amigo2. In *Synthetic Gene Circuits*, pages 221–239. Springer, 2021.

[73] Tim Litwin, Jens Timmer, and Clemens Kreutz. Optimal experimental design based on two-dimensional likelihood profiles. *Frontiers in molecular biosciences*, 9, 2022.

[74] Eva Balsa-Canto, David Henriques, Attila Gábor, and Julio R Banga. Amigo2, a toolbox for dynamic modeling, optimization and control in systems biology. *Bioinformatics*, 32(21):3357–3359, 2016.

[75] Daniel Faller, Ursula Klingmüller, and Jens Timmer. Simulation methods for optimal experimental design in systems biology. *Simulation*, 79(12):717–725, 2003.

[76] Neythen J. Treloar, Nathan Braniff, Brian Ingalls, and Chris P. Barnes. Deep reinforcement learning for optimal experimental design in biology. *bioRxiv*, 2022.

[77] Dennis V Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005, 1956.

[78] Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995.

[79] Alejandro F Villaverde, Elba Raimúndez, Jan Hasenauer, and Julio R Banga. A comparison of methods for quantifying prediction uncertainty in systems biology. *IFAC-PapersOnLine*, 52(26):45–51, 2019.

[80] Oana-Teodora Chis, Alejandro F Villaverde, Julio R Banga, and Eva Balsa-Canto. On the relationship between sloppiness and identifiability. *Mathematical biosciences*, 282:147–161, 2016.

[81] Zachary R Fox and Brian Munsky. The finite state projection based fisher information matrix approach to estimate information and optimize single-cell experiments. *PLoS computational biology*, 15(1):e1006365, 2019.

[82] Elizabeth G Ryan, Christopher C Drovandi, James M McGree, and Anthony N Pettitt. A review of modern computational algorithms for bayesian optimal design. *International Statistical Review*, 84(1):128–154, 2016.

[83] Stan Development Team. Stan modeling language users guide and reference manual, 2.28, 2022.

[84] Pejman Honarmandi and Raymundo Arróyave. Uncertainty quantification and propagation in computational materials science and simulation-assisted materials design. *Integrating Materials and Manufacturing Innovation*, 9(1):103–143, 2020.

[85] Jean-Baptiste Lugagne, Sebastián Sosa Carrillo, Melanie Kirch, Agnes Köhler, Gregory Batt, and Pascal Hersen. Balancing a genetic toggle switch by real-time feedback control and periodic forcing. *Nature communications*, 8(1):1–8, 2017.

[86] Lucia Bandiera, Zhaozheng Hou, Varun B Kothamachu, Eva Balsa-Canto, Peter S Swain, and Filippo Menolascina. On-line optimal input design increases the efficiency and accuracy of the modelling of an inducible synthetic promoter. *Processes*, 6(9):148, 2018.

[87] Suliman Khan, Muhammad Wajid Ullah, Rabeea Siddique, Ghulam Nabi, Sehrish Manan, Muhammad Yousaf, and Hongwei Hou. Role of recombinant dna technology to improve life. *International journal of genomics*, 2016, 2016.

[88] Tie Koide, Wyming Lee Pang, and Nitin S Baliga. The role of predictive modelling in rationally re-engineering biological systems. *Nature Reviews Microbiology*, 7(4):297–305, 2009.

[89] Richard J Roberts. How restriction enzymes became the workhorses of molecular biology. *Proceedings of the National Academy of Sciences*, 102(17):5905–5908, 2005.

[90] Behnam Enghiad and Huimin Zhao. Programmable dna-guided artificial restriction enzymes. *ACS synthetic biology*, 6(5):752–757, 2017.

[91] FAFA Ran, Patrick D Hsu, Jason Wright, Vineeta Agarwala, David A Scott, and Feng Zhang. Genome engineering using the crispr-cas9 system. *Nature protocols*, 8(11):2281–2308, 2013.

[92] Darshana Gupta, Oindrila Bhattacharjee, Drishti Mandal, Madhab Kumar Sen, Dhritiman Dey, Adhiraj Dasgupta, Tawsif Ahmed Kazi, Rahul Gupta, Senjuti Sinharoy, Krishnendu Acharya, et al. Crispr-cas9 system: A new-fangled dawn in gene editing. *Life sciences*, 232:116636, 2019.

[93] Jose-Luis Adrio and Arnold L Demain. Recombinant organisms for production of industrial products. *Bioengineered bugs*, 1(2):116–131, 2010.

[94] Natalia Pogrebnyak, Maxim Golovkin, Vyacheslav Andrianov, Sergei Spitsin, Yuriy Smirnov, Richard Egolf, and Hilary Koprowski. Severe acute respiratory syndrome (sars) s protein production in plants: development

of recombinant vaccine. *Proceedings of the National Academy of Sciences*, 102(25):9062–9067, 2005.

[95] Nabih A Baeshen, Mohammed N Baeshen, Abdullah Sheikh, Roop S Bora, Mohamed Morsi M Ahmed, Hassan AI Ramadan, Kulvinder Singh Saini, and Elrashdy M Redwan. Cell factories for insulin production. *Microbial cell factories*, 13(1):1–9, 2014.

[96] Theo Dingermann. Recombinant therapeutic proteins: production platforms and challenges. *Biotechnology Journal: Healthcare Nutrition Technology*, 3(1):90–97, 2008.

[97] Carolyn M Bonin and Frederic L Kolb. Resistance to fusarium head blight and kernel damage in a winter wheat recombinant inbred line population. *Crop Science*, 49(4):1304–1312, 2009.

[98] Sarah M Pearsall, Christopher N Rowley, and Alan Berry. Advances in pathway engineering for natural product biosynthesis. *ChemCatChem*, 7(19):3078–3093, 2015.

[99] CA Merrick, C Wardrope, JE Paget, SD Colloms, and SJ Rosser. Rapid optimization of engineered metabolic pathways with serine integrase recombinational assembly (sira). In *Methods in Enzymology*, volume 575, pages 285–317. Elsevier, 2016.

[100] Jing Du, Yongbo Yuan, Tong Si, Jiazhang Lian, and Huimin Zhao. Customized optimization of metabolic pathways by combinatorial transcriptional engineering. *Nucleic acids research*, 40(18):e142–e142, 2012.

[101] Shasha Zhang, Xuejin Zhao, Yong Tao, and Chunbo Lou. A novel approach for metabolic pathway optimization: Oligo-linker mediated assembly (olma) method. *Journal of biological engineering*, 9(1):1–10, 2015.

[102] Gita Naseri and Mattheos AG Koffas. Application of combinatorial optimization strategies in synthetic biology. *Nature communications*, 11(1):1–14, 2020.

[103] J Andrew Jones and Mattheos AG Koffas. Optimizing metabolic pathways for the improved production of natural products. In *Methods in Enzymology*, volume 575, pages 179–193. Elsevier, 2016.

[104] Tae Seok Moon, Chunbo Lou, Alvin Tamsir, Brynne C Stanton, and Christopher A Voigt. Genetic programs constructed from layered logic gates in single cells. *Nature*, 491(7423):249–253, 2012.

[105] Meriem El Karoui, Monica Hoyos-Flight, and Liz Fletcher. Future trends in synthetic biology—a report. *Frontiers in bioengineering and biotechnology*, page 175, 2019.

[106] Drew Endy. Foundations for engineering biology. *Nature*, 438(7067):449–453, 2005.

[107] Sascha Rollié, Michael Mangold, and Kai Sundmacher. Designing biological systems: systems engineering meets synthetic biology. *Chemical Engineering Science*, 69(1):1–29, 2012.

[108] MAJ Roberts, RM Cranenburgh, MP Stevens, and PCF Oyston. Synthetic biology: biology by design. *Microbiology*, 159(Pt 7):1219, 2013.

[109] Timothy Frei, Federica Cella, Fabiana Tedeschi, Joaquín Gutiérrez, Guy-Bart Stan, Mustafa Khammash, and Velia Siciliano. Characterization and mitigation of gene expression burden in mammalian cells. *Nature communications*, 11(1):1–14, 2020.

[110] Dirk Benzinger, Serguei Ovinnikov, and Mustafa Khammash. Synthetic gene networks recapitulate dynamic signal decoding and differential gene expression. *Cell Systems*, 2022.

[111] Kathakali Sarkar and Sangram Bagh. Synthetic gene circuits for higher-order information processing. In *New Frontiers and Applications of Synthetic Biology*, pages 373–395. Elsevier, 2022.

[112] Ari E Friedland, Timothy K Lu, Xiao Wang, David Shi, George Church, and James J Collins. Synthetic gene networks that count. *science*, 324(5931):1199–1202, 2009.

[113] Domitilla Del Vecchio, Aaron J Dy, and Yili Qian. Control theory meets synthetic biology. *Journal of The Royal Society Interface*, 13(120):20160380, 2016.

[114] Jesse Stricker, Scott Cookson, Matthew R Bennett, William H Mather, Lev S Tsimring, and Jeff Hasty. A fast, robust and tunable synthetic gene oscillator. *Nature*, 456(7221):516–519, 2008.

[115] Pavithran T Ravindran, Sarah McFann, Richard H Thornton, and Jared E Toettcher. A synthetic gene circuit for imaging-free detection of signaling pulses. *Cell Systems*, 13(2):131–142, 2022.

[116] Baojun Wang, Richard I Kitney, Nicolas Joly, and Martin Buck. Engineering modular and orthogonal genetic logic gates for robust digital-like synthetic biology. *Nature communications*, 2(1):1–9, 2011.

[117] Tal Danino, Octavio Mondragón-Palomino, Lev Tsimring, and Jeff Hasty. A synchronized quorum of genetic clocks. *Nature*, 463(7279):326–330, 2010.

[118] Lara Tess Bereza-Malcolm, Gülay Mann, and Ashley Edwin Franks. Environmental sensing of heavy metals through whole cell microbial biosensors: a synthetic biology approach. *ACS synthetic biology*, 4(5):535–546, 2015.

[119] Jan Claesen and Michael A Fischbach. Synthetic microbes as drug delivery systems. *ACS synthetic biology*, 4(4):358–364, 2015.

[120] Hannes Mutschler, Tom Robinson, T-Y Dora Tang, and Seraphine Wegner. Special issue on bottom-up synthetic biology, 2019.

[121] Nadanai Laohakunakorn, Laura Grasemann, Barbora Lavickova, Grégoire Michielin, Amir Shahein, Zoe Swank, and Sebastian J Maerkl. Bottom-up construction of complex biomolecular systems with cell-free synthetic biology. *Frontiers in bioengineering and biotechnology*, 8:213, 2020.

[122] Nadanai Laohakunakorn. Cell-free systems: a proving ground for rational biodesign. *Frontiers in Bioengineering and Biotechnology*, 8:788, 2020.

[123] Aidan Tinafar, Katariina Jaenes, and Keith Pardee. Synthetic biology goes cell-free. *BMC biology*, 17(1):1–14, 2019.

[124] Richard JR Kelwick, Alexander J Webb, and Paul S Freemont. Biological materials: the next frontier for cell-free synthetic biology. *Frontiers in Bioengineering and Biotechnology*, 8:399, 2020.

[125] Mitchell S Weisenberger and Tara L Deans. Bottom-up approaches in synthetic biology and biomaterials for tissue engineering applications. *Journal of Industrial Microbiology and Biotechnology*, 45(7):599–614, 2018.

[126] Keith Pardee, Alexander A Green, Tom Ferrante, D Ewen Cameron, Ajay DaleyKeyser, Peng Yin, and James J Collins. Paper-based synthetic gene networks. *Cell*, 159(4):940–954, 2014.

[127] Peter Q Nguyen, Luis R Soenksen, Nina M Donghia, Nicolaas M Angenent-Mari, Helena de Puig, Ally Huang, Rose Lee, Shimyn Slomovic, Tommaso Galbersanini, Geoffrey Lansberry, et al. Wearable materials with embedded synthetic biology sensors for biomolecule detection. *Nature Biotechnology*, 39(11):1366–1374, 2021.

[128] Junbiao Dai, Jef D Boeke, Zhouqing Luo, Shuangying Jiang, and Yizhi Cai. Sc3. 0: Revamping and minimizing the yeast genome, 2020.

[129] Jean-Christophe Lachance, Sébastien Rodrigue, and Bernhard O Palsson. Synthetic biology: minimal cells, maximal knowledge. *Elife*, 8:e45379, 2019.

[130] Christopher M Whitford, Pablo Cruz-Morales, Jay D Keasling, and Tilmann Weber. The design-build-test-learn cycle for metabolic engineering of streptomycetes. *Essays in Biochemistry*, 65(2):261–275, 2021.

[131] Christopher A Voigt. Genetic parts to program bacteria. *Current opinion in biotechnology*, 17(5):548–557, 2006.

[132] Priscilla EM Purnick and Ron Weiss. The second wave of synthetic biology: from modules to systems. *Nature reviews Molecular cell biology*, 10(6):410–422, 2009.

[133] Hsin-Ho Huang, Daniel Camsund, Peter Lindblad, and Thorsten Heidorn. Design and characterization of molecular tools for a synthetic biology approach towards developing cyanobacterial biotechnology. *Nucleic acids research*, 38(8):2577–2593, 2010.

[134] Reshma P Shetty, Drew Endy, and Thomas F Knight. Engineering biobrick vectors from biobrick parts. *Journal of biological engineering*, 2(1):1–12, 2008.

[135] Mariana Gómez-Schiavon, Galen Dods, Hana El-Samad, and Andrew H Ng. Multidimensional characterization of parts enhances modeling accuracy in genetic circuits. *ACS Synthetic Biology*, 9(11):2917–2926, 2020.

[136] Yadira Boada, Alejandro Vignoni, Iván Alarcon-Ruiz, Carlos Andreu-Vilarroig, Roger Monfort-Llorens, Adrián Requena, and Jesús Picó.

Characterization of gene circuit parts based on multiobjective optimization by using standard calibrated measurements. *ChemBioChem*, 20(20):2653–2665, 2019.

[137] Thomas E Gorochowski, Amin Espah Borujeni, Yongjin Park, Alec AK Nielsen, Jing Zhang, Bryan S Der, D Benjamin Gordon, and Christopher A Voigt. Genetic circuit characterization and debugging using rna-seq. *Molecular systems biology*, 13(11):952, 2017.

[138] Paul Opgenorth, Zak Costello, Takuya Okada, Garima Goyal, Yan Chen, Jennifer Gin, Veronica Benites, Markus de Raad, Trent R Northen, Kai Deng, et al. Lessons from two design–build–test–learn cycles of dodecanol production in escherichia coli aided by machine learning. *ACS Synthetic Biology*, 8(6):1337–1351, 2019.

[139] Pablo Carbonell, Adrian J Jervis, Christopher J Robinson, Cunyu Yan, Mark Dunstan, Neil Swainston, Maria Vinaixa, Katherine A Hollywood, Andrew Currin, Nicholas JW Rattray, et al. An automated design-build-test-learn pipeline for enhanced microbial production of fine chemicals. *Communications biology*, 1(1):1–10, 2018.

[140] Dmitriy M Chudakov, Sergey Lukyanov, and Konstantin A Lukyanov. Fluorescent proteins as a toolkit for in vivo imaging. *Trends in biotechnology*, 23(12):605–613, 2005.

[141] Stephen B Rigoulot, Tayler M Schimel, Jun Hyung Lee, Robert G Sears, Holly Brabazon, Jessica S Layton, Li Li, Kerry A Meier, Magen R Poindexter, Manuel J Schmid, et al. Imaging of multiple fluorescent proteins in canopies enables synthetic biology in plants. *Plant biotechnology journal*, 19(4):830–843, 2021.

[142] Roberta Lentini, Michele Forlin, Laura Martini, Cristina Del Bianco, Amy C Spencer, Domenica Torino, and Sheref S Mansy. Fluorescent proteins and in vitro genetic organization for cell-free synthetic biology. *ACS synthetic biology*, 2(9):482–489, 2013.

[143] Roger Y Tsien. The green fluorescent protein. *Annual review of biochemistry*, 67(1):509–544, 1998.

[144] Eric Lubeck and Long Cai. Single-cell systems biology by super-resolution imaging and combinatorial labeling. *Nature methods*, 9(7):743–748, 2012.

[145] Jaroslav Icha, Michael Weber, Jennifer C Waters, and Caren Norden. Phototoxicity in live fluorescence microscopy, and how to avoid it. *BioEssays*, 39(8):1700003, 2017.

[146] Michael W Davidson and Robert E Campbell. Engineered fluorescent proteins: innovations and applications. *Nature methods*, 6(10):713–717, 2009.

[147] Kevin M Dean and Amy E Palmer. Advances in fluorescence labeling strategies for dynamic cellular imaging. *Nature chemical biology*, 10(7):512–523, 2014.

[148] Nika Shakiba, Ross D Jones, Ron Weiss, and Domitilla Del Vecchio. Context-aware synthetic biology by controller design: Engineering the mammalian cell. *Cell Systems*, 12(6):561–592, 2021.

[149] Stefano Cardinale and Adam Paul Arkin. Contextualizing context for synthetic biology–identifying causes of failure of synthetic biological systems. *Biotechnology journal*, 7(7):856–866, 2012.

[150] Ye Chen, Joanne ML Ho, David L Shis, Chinmaya Gupta, James Long, Daniel S Wagner, William Ott, Krešimir Josić, and Matthew R Bennett. Tuning the dynamic range of bacterial promoters regulated by ligand-inducible transcription factors. *Nature communications*, 9(1):1–8, 2018.

[151] Alan Costello and Ahmed H Badran. Synthetic biological circuits within an orthogonal central dogma. *Trends in biotechnology*, 39(1):59–71, 2021.

[152] Ying-Ja Chen, Peng Liu, Alec AK Nielsen, Jennifer AN Brophy, Kevin Clancy, Todd Peterson, and Christopher A Voigt. Characterization of 582 natural and synthetic terminators and quantification of their design constraints. *Nature methods*, 10(7):659–664, 2013.

[153] Jennifer AN Brophy and Christopher A Voigt. Principles of genetic circuit design. *Nature methods*, 11(5):508–520, 2014.

[154] Mirna Kheir Gouda, Michael Manhart, and Gábor Balázsi. Evolutionary regain of lost gene circuit function. *Proceedings of the National Academy of Sciences*, 116(50):25162–25171, 2019.

[155] Alexander PS Darlington, Juhyun Kim, José I Jiménez, and Declan G Bates. Engineering translational resource allocation controllers: mechanistic

models, design guidelines, and potential biological implementations. *ACS Synthetic Biology*, 7(11):2485–2496, 2018.

[156] Alice Boo, Tom Ellis, and Guy-Bart Stan. Host-aware synthetic biology. *Current Opinion in Systems Biology*, 14:66–72, 2019.

[157] Quentin M Dudley, Ashty S Karim, and Michael C Jewett. Cell-free metabolic engineering: biomanufacturing beyond the cell. *Biotechnology journal*, 10(1):69–82, 2015.

[158] Fangzhong Wang and Weiwen Zhang. Synthetic biology: recent progress, biosafety and biosecurity concerns, and possible solutions. *Journal of Biosafety and Biosecurity*, 1(1):22–30, 2019.

[159] Eugene Fletcher, Anastasia Krivoruchko, and Jens Nielsen. Industrial systems biology and its impact on synthetic biology of yeast cell factories. *Biotechnology and bioengineering*, 113(6):1164–1170, 2016.

[160] Hong Zeng, Reza Rohani, Wei E Huang, and Aidong Yang. Understanding and mathematical modelling of cellular resource allocation in microorganisms: a comparative synthesis. *BMC bioinformatics*, 22(1):1–22, 2021.

[161] Thomas Decoene, Brecht De Paepe, Jo Maertens, Pieter Coussement, Gert Peters, Sofie L De Maeseneire, and Marjan De Mey. Standardization in synthetic biology: an engineering discipline coming of age. *Critical reviews in biotechnology*, 38(5):647–656, 2018.

[162] Kevin Clancy and Christopher A Voigt. Programming cells: towards an automated 'genetic compiler'. *Current Opinion in Biotechnology*, 21(4):572–581, 2010.

[163] Olivia Gallup, Hia Ming, and Tom Ellis. Ten future challenges for synthetic biology. *Engineering Biology*, 5(3):51–59, 2021.

[164] Katharina F Sonnen and Christoph A Merten. Microfluidics as an emerging precision tool in developmental biology. *Developmental cell*, 48(3):293–311, 2019.

[165] Vera Ortseifen, Martina Viefhues, Lutz Wobbe, and Alexander Grünberger. Microfluidics for biotechnology: bridging gaps to foster microfluidic applications. *Frontiers in Bioengineering and Biotechnology*, page 1324, 2020.

[166] Guilhem Velve-Casquillas, Maël Le Berre, Matthieu Piel, and Phong T Tran. Microfluidic tools for cell biological research. *Nano today*, 5(1):28–47, 2010.

[167] Mikihiro Hashimoto, Takashi Nozoe, Hidenori Nakaoka, Reiko Okura, Sayo Akiyoshi, Kunihiko Kaneko, Edo Kussell, and Yuichi Wakamoto. Noise-driven growth rate gain in clonal cellular populations. *Proceedings of the National Academy of Sciences*, 113(12):3251–3256, 2016.

[168] Elisa Pedone, Lorena Postiglione, Francesco Aulicino, Dan L Rocca, Sandra Montes-Olivas, Mahmoud Khazim, Diego di Bernardo, Maria Pia Cosma, and Lucia Marucci. A tunable dual-input system for on-demand dynamic gene expression regulation. *Nature communications*, 10(1):1–13, 2019.

[169] Matthias Kaiser, Florian Jug, Thomas Julou, Siddharth Deshpande, Thomas Pfohl, Olin K Silander, Gene Myers, and Erik Van Nimwegen. Monitoring single-cell gene regulation under dynamically controllable conditions with integrated microfluidics and software. *Nature communications*, 9(1):1–16, 2018.

[170] Mike S Ferry, Ivan A Razinkov, and Jeff Hasty. Microfluidics for synthetic biology: from design to execution. *Methods in enzymology*, 497:295–372, 2011.

[171] David J Beebe, Glennys A Mensing, and Glenn M Walker. Physics and applications of microfluidics in biology. *Annual review of biomedical engineering*, 4(1):261–286, 2002.

[172] David Holmes and Shady Gawad. The application of microfluidics in biology. *Microengineering in biotechnology*, pages 55–80, 2010.

[173] Kangning Ren, Yin Chen, and Hongkai Wu. New materials for microfluidics in biology. *Current opinion in biotechnology*, 25:78–85, 2014.

[174] Yang Bai, Meng Gao, Lingling Wen, Caiyun He, Yuan Chen, Chenli Liu, Xiongfei Fu, and Shuqiang Huang. Applications of microfluidics in quantitative biology. *Biotechnology Journal*, 13(5):1700170, 2018.

[175] Laila A Damiati, Safa A Damiati, and Samar Damiati. Developments in the use of microfluidics in synthetic biology. In *New Frontiers and Applications of Synthetic Biology*, pages 423–435. Elsevier, 2022.

[176] Kristina Woodruff and Sebastian J Maerkl. A high-throughput microfluidic platform for mammalian cell transfection and culturing. *Scientific reports*, 6(1):1–12, 2016.

[177] Filippo Menolascina, Gianfranco Fiore, Emanuele Orabona, Luca De Stefano, Mike Ferry, Jeff Hasty, Mario di Bernardo, and Diego di Bernardo. In-vivo real-time control of protein expression from endogenous and synthetic gene networks. *PLoS computational biology*, 10(5):e1003625, 2014.

[178] Samar Damiati, Rami Mhanna, Rimantas Kodzius, and Eva-Kathrin Ehmoser. Cell-free approaches in synthetic biology utilizing microfluidics. *Genes*, 9(3):144, 2018.

[179] Michael J Lawson, Daniel Camsund, Jimmy Larsson, Özden Baltekin, David Fange, and Johan Elf. In situ genotyping of a pooled strain library after characterizing complex phenotypes. *Molecular systems biology*, 13(10):947, 2017.

[180] Sandra Pérez-Rodríguez, José Manuel García-Aznar, and Jesús Gonzalo-Asensio. Microfluidic devices for studying bacterial taxis, drug testing and biofilm formation. *Microbial Biotechnology*, 15(2):395–414, 2022.

[181] Hyun Jung Kim, James Q Boedicker, Jang Wook Choi, and Rustem F Ismagilov. Defined spatial structure stabilizes a synthetic multispecies bacterial community. *Proceedings of the National Academy of Sciences*, 105(47):18188–18193, 2008.

[182] Larry J Millet, Jayde Aufrecht, Jessy Labbé, Jessie Uehling, Rytas Vilgalys, Myka L Estes, Cora Miquel Guennoc, Aurélie Deveau, Stefan Olsson, Gregory Bonito, et al. Increasing access to microfluidics for studying fungi and other branched biological structures. *Fungal biology and biotechnology*, 6(1):1–14, 2019.

[183] Ningren Han, Oliver Purcell, Fahim Farzadfard, Kevin S Lee, Timothy K Lu, and Rajeev J Ram. Microfluidics for control in synthetic biology. In *presentation at the 18th International Conference on Miniaturized Systems for Chemistry and Life Sciences, MicroTAS*, pages 312–314, 2014.

[184] Leo Baumgart, William Mather, and Jeff Hasty. Synchronized dna cycling across a bacterial population. *Nature genetics*, 49(8):1282–1285, 2017.

[185] Tim Prangemeier, François-Xavier Lehr, Rogier M Schoeman, and Heinz Koeppl. Microfluidic platforms for the dynamic characterisation of synthetic circuitry. *Current opinion in biotechnology*, 63:167–176, 2020.

[186] Gregory Linshiz, Erik Jensen, Nina Stawski, Changhao Bi, Nick Elsbree, Hong Jiao, Jungkyu Kim, Richard Mathies, Jay D Keasling, and Nathan J Hillson. End-to-end automated microfluidic platform for synthetic biology: from design to functional analysis. *Journal of biological engineering*, 10(1):1–15, 2016.

[187] Philip C Gach, Kosuke Iwai, Peter W Kim, Nathan J Hillson, and Anup K Singh. Droplet microfluidics for synthetic biology. *Lab on a Chip*, 17(20):3388–3400, 2017.

[188] Yuliya Khilko, Philip D Weyman, John I Glass, Mark D Adams, Melanie A McNeil, and Peter B Griffin. Dna assembly with error correction on a droplet digital microfluidics platform. *BMC biotechnology*, 18(1):1–14, 2018.

[189] Tao Geng and Chang Lu. Microfluidic electroporation for cellular analysis and delivery. *Lab on a Chip*, 13(19):3803–3821, 2013.

[190] Jungkyu Kim, Inseong Hwang, Derek Britain, Taek Dong Chung, Yu Sun, and Deok-Ho Kim. Microfluidic approaches for gene delivery and gene therapy. *Lab on a Chip*, 11(23):3941–3948, 2011.

[191] Mira T Guo, Assaf Rotem, John A Heyman, and David A Weitz. Droplet microfluidics for high-throughput biological assays. *Lab on a Chip*, 12(12):2146–2155, 2012.

[192] Yigang Shen, Yaxiaer Yalikun, and Yo Tanaka. Recent advances in microfluidic cell sorting systems. *Sensors and Actuators B: Chemical*, 282:268–281, 2019.

[193] Rachelle Prantil-Baun, Richard Novak, Debarun Das, Mahadevabharath R Somayaji, Andrzej Przekwas, and Donald E Ingber. Physiologically based pharmacokinetic and pharmacodynamic analysis enabled by microfluidically linked organs-on-chips. *Annual review of pharmacology and toxicology*, 58:37–64, 2018.

[194] Qirui Wu, Jinfeng Liu, Xiaohong Wang, Lingyan Feng, Jinbo Wu, Xiaoli Zhu, Weijia Wen, and Xiuqing Gong. Organ-on-a-chip: Recent breakthroughs and future prospects. *Biomedical engineering online*, 19(1):1–19, 2020.

[195] Lucie A Low, Christine Mummery, Brian R Berridge, Christopher P Austin, and Danilo A Tagle. Organs-on-chips: into the next decade. *Nature Reviews Drug Discovery*, 20(5):345–361, 2021.

[196] Alex Ede Danku, Eva-H Dulf, Cornelia Braicu, Ancuta Jurj, and Ioana Berindan-Neagoe. Organ-on-a-chip: A survey of technical results and problems. *Frontiers in Bioengineering and Biotechnology*, 10, 2022.

[197] Kenneth A Johnson and Roger S Goody. The original michaelis constant: translation of the 1913 michaelis–menten paper. *Biochemistry*, 50(39):8264–8269, 2011.

[198] Sylvain Goutelle, Michel Maurin, Florent Rougier, Xavier Barbaut, Laurent Bourguignon, Michel Ducher, and Pascal Maire. The hill equation: a review of its capabilities in pharmacological modelling. *Fundamental & clinical pharmacology*, 22(6):633–648, 2008.

[199] Nestor V Torres and Guido Santos. The (mathematical) modeling process in biosciences. *Frontiers in Genetics*, 6:354, 2015.

[200] M Traykov and Iv Trenchev. Mathematical models in genetics. *Russian Journal of Genetics*, 52(9):985–992, 2016.

[201] Miklós Farkas. *Dynamical models in biology*. Academic press, 2001.

[202] Christoph P Bagowski and James E Ferrell Jr. Bistability in the jnk cascade. *Current Biology*, 11(15):1176–1182, 2001.

[203] David G García Münzer, Margaritis Kostoglou, Michael C Georgiadis, Efstratios N Pistikopoulos, and Athanasios Mantalaris. Cyclin and dna distributed cell cycle model for gs-ns0 cells. *PLoS computational biology*, 11(2):e1004062, 2015.

[204] Jan Ewald, Patricia Sieber, Ravindra Garde, Stefan N Lang, Stefan Schuster, and Bashar Ibrahim. Trends in mathematical modeling of host–pathogen interactions. *Cellular and Molecular Life Sciences*, 77(3):467–480, 2020.

[205] Xin Lai, Olaf Wolkenhauer, and Julio Vera. Understanding microrna-mediated gene regulatory networks through mathematical modelling. *Nucleic acids research*, 44(13):6019–6035, 2016.

[206] Nathan Crook and Hal S Alper. Model-based design of synthetic, biological systems. *Chemical Engineering Science*, 103:2–11, 2013.

[207] Alan U Sabino, Miguel FS Vasconcelos, Misaki Yamada Sittoni, Willian W Lautenschlager, Alexandre S Queiroga, Mauro CC Morais, and Alexandre F Ramos. Lessons and perspectives for applications of stochastic models in biological and cancer research. *Clinics*, 73, 2018.

[208] Darren J Wilkinson. Stochastic modelling for quantitative description of heterogeneous biological systems. *Nature Reviews Genetics*, 10(2):122–133, 2009.

[209] Verena Wolf, Rushil Goel, Maria Mateescu, and Thomas A Henzinger. Solving the chemical master equation using sliding windows. *BMC systems biology*, 4(1):1–19, 2010.

[210] Ian S Hsu and Alan M Moses. Stochastic models for single-cell data: Current challenges and the way forward. *The FEBS Journal*, 289(3):647–658, 2022.

[211] Sayuri K Hahl and Andreas Kremling. A comparison of deterministic and stochastic modeling approaches for biochemical reaction systems: on fixed points, means, and modes. *Frontiers in genetics*, 7:157, 2016.

[212] Jasia King, Kerbaï Saïd Eroumé, Roman Truckenmüller, Stefan Giselbrecht, Ann E Cowan, Leslie Loew, and Aurélie Carlier. Ten steps to investigate a cellular system with mathematical modeling. *PLoS computational biology*, 17(5):e1008921, 2021.

[213] Brian P Ingalls. *Mathematical modeling in systems biology: an introduction*. MIT press, 2013.

[214] Johannes Müller and Christina Kuttler. Methods and models in mathematical biology. *Lecture Notes on Mathematical Modelling in Life Sciences, Springer, Berlin*, 2015.

[215] Jamie Twycross, Leah R Band, Malcolm J Bennett, John R King, and Natalio Krasnogor. Stochastic and deterministic multiscale models for systems biology: an auxin-transport case study. *BMC Systems Biology*, 4(1):1–11, 2010.

[216] Jack A Tuszynski, Philip Winter, Diana White, Chih-Yuan Tseng, Kamlesh K Sahu, Francesco Gentile, Ivana Spasevska, Sara Ibrahim Omar, Niloofar Nayebi, Cassandra DM Churchill, et al. Mathematical and computational modeling in biology at multiple scales. *Theoretical Biology and Medical Modelling*, 11(1):1–42, 2014.

[217] Angélique Stéphanou and Vitaly Volpert. Hybrid modelling in biology: a classification review. *Mathematical Modelling of Natural Phenomena*, 11(1):37–48, 2016.

[218] Enrico Gavagnin and Christian A Yates. Stochastic and deterministic modeling of cell migration. In *Handbook of statistics*, volume 39, pages 37–91. Elsevier, 2018.

[219] Arjan van der Schaft, Shodhan Rao, and Bayu Jayawardhana. On the mathematical structure of balanced chemical reaction networks governed by mass action kinetics. *SIAM Journal on Applied Mathematics*, 73(2):953–973, 2013.

[220] Darren J Wilkinson. Stochastic modelling for quantitative description of heterogeneous biological systems. *Nature Reviews Genetics*, 10(2):122–133, 2009.

[221] Blake S Pollard. Open markov processes: A compositional perspective on non-equilibrium steady states in biology. *Entropy*, 18(4):140, 2016.

[222] Mohieddin Jafari, Naser Ansari-Pour, Sadegh Azimzadeh, and Mehdi Mirzaie. A logic-based dynamic modeling approach to explicate the evolution of the central dogma of molecular biology. *PloS one*, 12(12):e0189922, 2017.

[223] Hong Qian and Hao Ge. Kinetics of the central dogma of molecular cell biology. In *Stochastic Chemical Reaction Systems in Biology*, pages 259–280. Springer, 2021.

[224] Ting Chen, Hongyu L He, and George M Church. Modeling gene expression with differential equations. In *Biocomputing'99*, pages 29–40. World Scientific, 1999.

[225] David S Glass, Xiaofan Jin, and Ingmar H Riedel-Kruse. Nonlinear delay differential equations and their application to modeling biological network motifs. *Nature communications*, 12(1):1–19, 2021.

[226] Andres Ortiz. Derivation of hill's equation from scale invariance. *Journal of Uncertain Systems*, 7(3):198–202, 2013.

[227] Moises Santillán. On the use of the hill functions in mathematical models of gene regulatory networks. *Mathematical Modelling of Natural Phenomena*, 3(2):85–97, 2008.

[228] Robert Gnügge, Lekshmi Dharmarajan, Moritz Lang, and Jörg Stelling. An orthogonal permease–inducer–repressor feedback loop shows bistability. *ACS synthetic biology*, 5(10):1098–1107, 2016.

[229] Eva Balsa-Canto, Antonio A Alonso, and Julio R Banga. An iterative identification procedure for dynamic modeling of biochemical networks. *BMC systems biology*, 4(1):1–18, 2010.

[230] Clemens Kreutz. A new approximation approach for transient differential equation models. *Frontiers in Physics*, 8:70, 2020.

[231] Philipp Städter, Yannik Schälte, Leonard Schmiester, Jan Hasenauer, and Paul L Stapor. Benchmarking of numerical integration methods for ode models of biological systems. *Scientific reports*, 11(1):1–11, 2021.

[232] Jan Rombouts and Lendert Gelens. Dynamic bistable switches enhance robustness and accuracy of cell cycle transitions. *PLoS Computational Biology*, 17(1):e1008231, 2021.

[233] Lei Wang, Brandon L Walker, Stephen Iannaccone, Devang Bhatt, Patrick J Kennedy, and T Tse William. Bistable switches control memory and plasticity in cellular differentiation. *Proceedings of the National Academy of Sciences*, 106(16):6638–6643, 2009.

[234] Joseph R Pomerening. Uncovering mechanisms of bistability in biological systems. *Current opinion in biotechnology*, 19(4):381–388, 2008.

[235] Tetsuya Shiraishi, Shinako Matsuyama, and Hiroaki Kitano. Large-scale analysis of network bistability for human cancers. *PLoS computational biology*, 6(7):e1000851, 2010.

[236] Michael Bednarz, Jennifer A Halliday, Christophe Herman, and Ido Golding. Revisiting bistability in the lysis/lysogeny circuit of bacteriophage lambda. *PLoS one*, 9(6):e100876, 2014.

[237] Luca Cardelli, Rosa D Hernansaiz-Ballesteros, Neil Dalchau, and Attila Csikász-Nagy. Efficient switches in biology and computer science. *PLoS computational biology*, 13(1):e1005100, 2017.

[238] Shintaro Nagata and Macoto Kikuchi. Emergence of cooperative bistability and robustness of gene regulatory networks. *PLoS computational biology*, 16(6):e1007969, 2020.

[239] Albert Goldbeter. Dissipative structures in biological systems: bistability, oscillations, spatial patterns and waves. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2124):20170376, 2018.

[240] Thomas Wilhelm. The smallest chemical reaction system with bistability. *BMC systems biology*, 3(1):1–9, 2009.

[241] Abhinav Tiwari, J Christian J Ray, Jatin Narula, and Oleg A Igoshin. Bistable responses in bacterial genetic networks: designs and dynamical consequences. *Mathematical biosciences*, 231(1):76–89, 2011.

[242] Qiang Zhang, Sudin Bhattacharya, Melvin E Andersen, and Rory B Conolly. Computational systems biology and dose-response modeling in relation to new directions in toxicity testing. *Journal of Toxicology and Environmental Health, Part B*, 13(2-4):253–276, 2010.

[243] Kirk D Dolan, Dharmendra K Mishra, et al. Parameter estimation in food science. *Annual review of food science and technology*, 4(1):401–422, 2013.

[244] Attila Gábor and Julio R Banga. Robust and efficient parameter estimation in dynamic models of biological systems. *BMC systems biology*, 9(1):1–25, 2015.

[245] Einav Simon and Daniel Kornitzer. Pulse-chase analysis to measure protein degradation. In *Methods in enzymology*, volume 536, pages 65–75. Elsevier, 2014.

[246] Kate Sidaway-Lee, Maria J Costa, David A Rand, Bärbel Finkenstadt, and Steven Penfield. Direct measurement of transcription rates reveals multiple mechanisms for configuration of the arabidopsisambient temperature response. *Genome biology*, 15(3):1–10, 2014.

[247] David R Penas, Patricia González, Jose A Egea, Ramón Doallo, and Julio R Banga. Parameter estimation in large-scale systems biology models: a parallel and self-adaptive cooperative strategy. *BMC bioinformatics*, 18(1):1–24, 2017.

[248] Christophe Andrieu and Johannes Thoms. A tutorial on adaptive mcmc. *Statistics and computing*, 18(4):343–373, 2008.

[249] Jianyong Sun, Jonathan M Garibaldi, and Charlie Hodgman. Parameter estimation using metaheuristics in systems biology: a comprehensive review. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 9(1):185–202, 2011.

[250] Eshan D Mitra and William S Hlavacek. Parameter estimation and uncertainty quantification for systems biology models. *Current opinion in systems biology*, 18:9–18, 2019.

[251] Carmen G Moles, Pedro Mendes, and Julio R Banga. Parameter estimation in biochemical pathways: a comparison of global optimization methods. *Genome research*, 13(11):2467–2474, 2003.

[252] Jose A Egea, Eva Balsa-Canto, Maria-Sonia G Garcia, and Julio R Banga. Dynamic optimization of nonlinear processes with an enhanced scatter search method. *Industrial & Engineering Chemistry Research*, 48(9):4388–4401, 2009.

[253] Kenneth V Price. Differential evolution. In *Handbook of optimization*, pages 187–214. Springer, 2013.

[254] Riccardo Poli, James Kennedy, and Tim Blackwell. Particle swarm optimization. *Swarm intelligence*, 1(1):33–57, 2007.

[255] Gloria I Valderrama-Bahamóndez and Holger Fröhlich. Mcmc techniques for parameter estimation of ode based models in systems biology. *Frontiers in Applied Mathematics and Statistics*, page 55, 2019.

[256] Paul Stapor, Daniel Weindl, Benjamin Ballnus, Sabine Hug, Carolin Loos, Anna Fiedler, Sabrina Krause, Sabrina Hroß, Fabian Fröhlich, and Jan Hasenauer. Pesto: parameter estimation toolbox. *Bioinformatics*, 34(4):705–707, 2018.

[257] Eshan D Mitra, Ryan Suderman, Joshua Colvin, Alexander Ionkov, Andrew Hu, Herbert M Sauro, Richard G Posner, and William S Hlavacek. Pybionetfit and the biological property specification language. *IScience*, 19:1012–1036, 2019.

[258] Juliane Liepe, Chris Barnes, Erika Cule, Kamil Erguler, Paul Kirk, Tina Toni, and Michael PH Stumpf. Abc-sysbio—approximate bayesian computation in python with gpu support. *Bioinformatics*, 26(14):1797–1799, 2010.

[259] Juliane Liepe, Paul Kirk, Sarah Filippi, Tina Toni, Chris P Barnes, and Michael PH Stumpf. A framework for parameter estimation and model selection from experimental data in systems biology using approximate bayesian computation. *Nature protocols*, 9(2):439–456, 2014.

[260] Francisco J Samaniego. *A comparison of the Bayesian and frequentist approaches to estimation*, volume 24. Springer, 2010.

[261] M Jésus Bayarri and James O Berger. The interplay of bayesian and frequentist analysis. *Statistical Science*, 19(1):58–80, 2004.

[262] David L. Banks and Stephen E. Fienberg. Statistics, multivariate. In Robert A. Meyers, editor, *Encyclopedia of Physical Science and Technology (Third Edition)*, pages 851–889. Academic Press, New York, third edition edition, 2003.

[263] Maria Rodriguez-Fernandez, Jose A Egea, and Julio R Banga. Novel metaheuristic for parameter estimation in nonlinear dynamic biological systems. *BMC bioinformatics*, 7(1):1–18, 2006.

[264] Allan Donner and GY Zou. Closed-form confidence intervals for functions of the normal mean and standard deviation. *Statistical Methods in Medical Research*, 21(4):347–359, 2012.

[265] Avijit Hazra. Using the confidence interval confidently. *Journal of thoracic disease*, 9(10):4125, 2017.

[266] Kwanghee Jung, Jaehoon Lee, Vibhuti Gupta, and Gyeongcheol Cho. Comparison of bootstrap confidence interval methods for gsca using a monte carlo simulation. *Frontiers in psychology*, 10:2215, 2019.

[267] Jerzy Neyman. Frequentist probability and frequentist statistics. *Synthese*, pages 97–131, 1977.

[268] Autcha Araveeporn. Comparing parameter estimation of random coefficient autoregressive model by frequentist method. *Mathematics*, 8(1):62, 2020.

[269] David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[270] Satoshi Morita, Peter F Thall, and Peter Müller. Evaluating the impact of prior assumptions in bayesian biostatistics. *Statistics in biosciences*, 2(1):1–17, 2010.

[271] Sarah Depaoli, Sonja D Winter, and Marieke Visser. The importance of prior sensitivity analysis in bayesian statistics: demonstrations using an interactive shiny app. *Frontiers in psychology*, 11:608045, 2020.

[272] Darren J Wilkinson. Bayesian methods in bioinformatics and computational systems biology. *Briefings in bioinformatics*, 8(2):109–116, 2007.

[273] Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I Jordan. An introduction to mcmc for machine learning. *Machine learning*, 50(1):5–43, 2003.

[274] Siddhartha Chib and Edward Greenberg. Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4):327–335, 1995.

[275] Tina Toni, David Welch, Natalja Strelkowa, Andreas Ipsen, and Michael PH Stumpf. Approximate bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface*, 6(31):187–202, 2009.

[276] Wangang Xie, Paul O Lewis, Yu Fan, Lynn Kuo, and Ming-Hui Chen. Improving marginal likelihood estimation for bayesian phylogenetic model selection. *Systematic biology*, 60(2):150–160, 2011.

[277] Kevin H Knuth, Michael Habeck, Nabin K Malakar, Asim M Mubeen, and Ben Placek. Bayesian evidence and model selection. *Digital Signal Processing*, 47:50–67, 2015.

[278] Nial Friel and Anthony N Pettitt. Marginal likelihood estimation via power posteriors. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(3):589–607, 2008.

[279] Tina Toni, Yu-ichi Ozaki, Paul Kirk, Shinya Kuroda, and Michael PH Stumpf. Elucidating the in vivo phosphorylation dynamics of the erk map kinase using quantitative proteomics data and bayesian model selection. *Molecular BioSystems*, 8(7):1921–1929, 2012.

[280] Rainer Opgen-Rhein and Korbinian Strimmer. Learning causal networks from systems biology time course data: an effective model selection procedure for the vector autoregressive process. *BMC bioinformatics*, 8(2):1–8, 2007.

[281] Laura Salter Kubatko. Identifying hybridization events in the presence of coalescence via model selection. *Systematic Biology*, 58(5):478–488, 2009.

[282] Kwangbom Choi, Yang Chen, Daniel A Skelly, and Gary A Churchill. Bayesian model selection reveals biological origins of zero inflation in single-cell transcriptomics. *Genome biology*, 21(1):1–16, 2020.

[283] Shawn J Leroux. On the prevalence of uninformative parameters in statistical models applying model selection in applied ecology. *PLoS One*, 14(2):e0206711, 2019.

[284] Wouter JP Kuijpers, Marinus JG van de Molengraft, Simon van Mourik, Albertus van't Ooster, Silke Hemming, and Eldert J van Henten. Model selection with a common structure: Tomato crop growth models. *biosystems engineering*, 187:247–257, 2019.

[285] Kwangbom Choi, Yang Chen, Daniel A Skelly, and Gary A Churchill. Bayesian model selection reveals biological origins of zero inflation in single-cell transcriptomics. *Genome biology*, 21:1–16, 2020.

[286] Andreas Raue, Clemens Kreutz, Thomas Maiwald, Julie Bachmann, Marcel Schilling, Ursula Klingmüller, and Jens Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–1929, 2009.

[287] Alejandro F Villaverde. Observability and structural identifiability of nonlinear biological systems. *Complexity*, 2019, 2019.

[288] Alejandro F Villaverde, Antonio Barreiro, and Antonis Papachristodoulou. Structural identifiability of dynamic systems biology models. *PLoS computational biology*, 12(10):e1005153, 2016.

[289] Franz-Georg Wieland, Adrian L Hauber, Marcus Rosenblatt, Christian Tönsing, and Jens Timmer. On structural and practical identifiability. *Current Opinion in Systems Biology*, 25:60–69, 2021.

[290] James WT Yates, Neil D Evans, and Michael J Chappell. Structural identifiability analysis via symmetries of differential equations. *Automatica*, 45(11):2585–2591, 2009.

[291] Johan Karlsson, Milena Anguelova, and Mats Jirstrand. An efficient method for structural identifiability analysis of large dynamic systems. *IFAC proceedings volumes*, 45(16):941–946, 2012.

[292] D Joubert, JD Stigter, and J Molenaar. An efficient procedure to assist in the re-parametrization of structurally unidentifiable models. *Mathematical biosciences*, 323:108328, 2020.

[293] Thomas S Ligon, Fabian Fröhlich, Oana T Chiş, Julio R Banga, Eva Balsa-Canto, and Jan Hasenauer. Genssi 2.0: multi-experiment structural identifiability analysis of sbml models. *Bioinformatics*, 34(8):1421–1423, 2018.

[294] Matthew J Simpson, Ruth E Baker, Sean T Vittadello, and Oliver J Maclaren. Practical parameter identifiability for spatio-temporal models of cell invasion. *Journal of the Royal Society Interface*, 17(164):20200055, 2020.

[295] Şebnem Yılmaz Balaman. Chapter 5 - uncertainty issues in biomass-based production chains. In Şebnem Yılmaz Balaman, editor, *Decision-Making for Biomass-Based Production Chains*, pages 113–142. Academic Press, 2019.

[296] Simeone Marino, Ian B Hogue, Christian J Ray, and Denise E Kirschner. A methodology for performing global uncertainty and sensitivity analysis in systems biology. *Journal of theoretical biology*, 254(1):178–196, 2008.

[297] Eva Balsa-Canto and Julio R Banga. Amigo: A model identification toolbox based on global optimization and its applications in biosystems. *IFAC Proceedings Volumes*, 43(6):132–137, 2010.

[298] Jeremy E Oakley and Anthony O'Hagan. Probabilistic sensitivity analysis of complex models: a bayesian approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(3):751–769, 2004.

[299] Pavel Loskot, Komlan Atitey, and Lyudmila Mihaylova. Comprehensive review of models and methods for inferences in bio-chemical reaction networks. *Frontiers in genetics*, page 549, 2019.

[300] W Becker, J Rowson, JE Oakley, Alaster Yoxall, G Manson, and K Worden. Bayesian sensitivity analysis of a model of the aortic valve. *Journal of biomechanics*, 44(8):1499–1506, 2011.

[301] Gaia Franceschini and Sandro Macchietto. Model-based design of experiments for parameter precision: State of the art. *Chemical Engineering Science*, 63(19):4846–4872, 2008.

[302] Clemens Kreutz and Jens Timmer. Systems biology: experimental design. *The FEBS journal*, 276(4):923–942, 2009.

[303] Roger Mead, Robert N Curnow, and Anne M Hasted. *Statistical methods in agriculture and experimental biology*. Chapman and Hall/CRC, 2017.

[304] Lev S Tsimring. Noise in biology. *Reports on Progress in Physics*, 77(2):026601, 2014.

[305] Roozbeh Dehghannasiri, Byung-Jun Yoon, and Edward R Dougherty. Optimal experimental design for gene regulatory networks in the presence of uncertainty. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 12(4):938–950, 2014.

[306] Samuel Bandara, Johannes P Schlöder, Roland Eils, Hans Georg Bock, and Tobias Meyer. Optimal experimental design for parameter estimation of a cell signaling model. *PLoS computational biology*, 5(11):e1000558, 2009.

[307] Fergal P Casey, Dan Baird, Qiyu Feng, Ryan N Gutenkunst, Joshua J Waterfall, Christopher R Myers, Kevin S Brown, Richard A Cerione, and James P Sethna. Optimal experimental design in an epidermal growth factor receptor signalling and down-regulation model. *IET systems biology*, 1(3):190–202, 2007.

[308] Olaf Wolkenhauer, Peter Wellstead, Kwang-Hyun Cho, Julio R Banga, and Eva Balsa-Canto. Parameter estimation and optimal experimental design. *Essays in biochemistry*, 45:195–210, 2008.

[309] Clemens Kreutz and Jens Timmer. *Optimal Experiment Design, Fisher Information*, pages 1576–1579. Springer New York, New York, NY, 2013.

[310] Myrta Rodriguez, Bradley Jones, Connie M Borror, and Douglas C Montgomery. Generating and assessing exact g-optimal designs. *Journal of quality technology*, 42(1):3–20, 2010.

[311] Hongyan Jiang and Yichuan Zhao. A review of bayesian optimal experimental design on different models. *Modern Statistical Methods for Health Research*, pages 205–220, 2021.

[312] Adam Foster, Martin Jankowiak, Elias Bingham, Paul Horsfall, Yee Whye Teh, Thomas Rainforth, and Noah Goodman. Variational bayesian optimal

experimental design. *Advances in Neural Information Processing Systems*, 32, 2019.

[313] MR Khodja, MD Prange, and HA Djikpesse. Guided bayesian optimal experimental design. *Inverse Problems*, 26(5):055008, 2010.

[314] Alen Alexanderian, Philip J Gloor, and Omar Ghattas. On bayesian a-and d-optimal experimental designs in infinite dimensions. *Bayesian Analysis*, 11(3):671–695, 2016.

[315] Costas Papadimitriou and Costas Argyris. Bayesian optimal experimental design for parameter estimation and response predictions in complex dynamical systems. *Procedia engineering*, 199:972–977, 2017.

[316] Xun Huan and Youssef M Marzouk. Simulation-based optimal bayesian experimental design for nonlinear systems. *Journal of Computational Physics*, 232(1):288–317, 2013.

[317] Claas Michalik, Maxim Stuckert, and Wolfgang Marquardt. Optimal experimental design for discriminating numerous model candidates: the awdc criterion. *Industrial & engineering chemistry research*, 49(2):913–919, 2010.

[318] Dominik Skanda and Dirk Lebiedz. An optimal experimental design approach to model discrimination in dynamic biochemical systems. *Bioinformatics*, 26(7):939–945, 2010.

[319] Daniel Silk, Paul DW Kirk, Chris P Barnes, Tina Toni, and Michael PH Stumpf. Model selection in systems biology depends on experimental design. *PLoS computational biology*, 10(6):e1003650, 2014.

[320] Simon Olofsson, Marc Peter Deisenroth, and Ruth Misener. Design of experiments for model discrimination using gaussian process surrogate models. In *Computer Aided Chemical Engineering*, volume 44, pages 847–852. Elsevier, 2018.

[321] Jesús López-Fidalgo and Chiara Tommasi. Optimal experimental design for model selection: a partial review. *The Mathematics of the Uncertain*, pages 253–263, 2018.

[322] Brecht MR Donckels, Dirk JW De Pauw, Bernard De Baets, Jo Maertens, and Peter A Vanrolleghem. An anticipatory approach to optimal

experimental design for model discrimination. *Chemometrics and Intelligent Laboratory Systems*, 95(1):53–63, 2009.

[323] D Bingham and H Chipman. Optimal designs for model selection. *Technometrics*, pages 1–20, 2002.

[324] Chiara Tommasi and Jesús López-Fidalgo. Bayesian optimum designs for discriminating between models with any distribution. *Computational Statistics & Data Analysis*, 54(1):143–150, 2010.

[325] Daniel Silk, Paul DW Kirk, Chris P Barnes, Tina Toni, Anna Rose, Simon Moon, Margaret J Dallman, and Michael PH Stumpf. Designing attractive models via automated identification of chaotic and oscillatory dynamical regimes. *Nature communications*, 2(1):1–6, 2011.

[326] Radu Serban and Alan C Hindmarsh. Cvodes: the sensitivity-enabled ode solver in sundials. In *ASME 2005 international design engineering technical conferences and computers and information in engineering conference*, pages 257–269. American Society of Mechanical Engineers, 2005.

[327] John R Dormand and Peter J Prince. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

[328] Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. [Online; accessed ¡today¿].

[329] Andrew Gelman, Hal S Stern, John B Carlin, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 2013.

[330] Michael Betancourt. A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434*, 2017.

[331] Matthew D Hoffman and Andrew Gelman. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.

[332] A.V. Lazo and P. Rathie. On the entropy of continuous probability distributions (corresp.). *IEEE Transactions on Information Theory*, 24(1):120–122, 1978.

[333] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical review E*, 69(6):066138, 2004.

[334] Marco F Huber, Tim Bailey, Hugh Durrant-Whyte, and Uwe D Hanebeck. On entropy approximation for gaussian mixture random vectors. In *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 181–188. IEEE, 2008.

[335] Wentian Li and Dale R Nyholt. Marker selection by akaike information criterion and bayesian information criterion. *Genetic Epidemiology*, 21(S1):S272–S277, 2001.

[336] Giovanni Fasano and Alberto Franceschini. A multidimensional version of the kolmogorov–smirnov test. *Monthly Notices of the Royal Astronomical Society*, 225(1):155–170, 1987.

[337] Thomas J DiCiccio and Bradley Efron. Bootstrap confidence intervals. *Statistical science*, 11(3):189–228, 1996.

[338] Chiara Gastaldi, Samuel Muscinelli, and Wulfram Gerstner. Optimal stimulation protocol in a bistable synaptic consolidation model. *Frontiers in Computational Neuroscience*, 13:78, 2019.

[339] Euisun Choi and Chulhee Lee. Feature extraction based on the bhattacharyya distance. *Pattern Recognition*, 36(8):1703–1709, 2003.

[340] Scott A Sarra. Regularized symmetric positive definite matrix factorizations for linear systems arising from rbf interpolation and differentiation. *Engineering Analysis with Boundary Elements*, 44:76–86, 2014.

[341] Peter I Frazier. Bayesian optimization. In *Recent Advances in Optimization and Modeling of Contemporary Problems*, pages 255–278. INFORMS, 2018.

[342] Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On bayesian upper confidence bounds for bandit problems. In *Artificial intelligence and statistics*, pages 592–600. PMLR, 2012.

[343] Jose A Egea, David Henriques, Thomas Cokelaer, Alejandro F Villaverde, Aidan MacNamara, Diana-Patricia Danciu, Julio R Banga, and Julio Saez-Rodriguez. Meigo: an open-source software suite based on metaheuristics for global optimization in systems biology and bioinformatics. *BMC bioinformatics*, 15(1):1–9, 2014.

[344] Miguel Viveiros, Ana Jesus, Mafalda Brito, Clara Leandro, Marta Martins, Diane Ordway, Ana Maria Molnar, Joseph Molnar, and Leonard Amaral. Inducement and reversal of tetracycline resistance in escherichia coli k-12 and expression of proton gradient-dependent multidrug efflux pump genes. *Antimicrobial agents and chemotherapy*, 49(8):3578–3582, 2005.

[345] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In Nassir Navab, Joachim Hornegger, William M. Wells, and Alejandro F. Frangi, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing.

[346] Thorsten Falk, Dominic Mai, Robert Bensch, Özgün Çiçek, Ahmed Abdulkadir, Yassine Marrakchi, Anton Böhm, Jan Deubner, Zoe Jäckel, Katharina Seiwald, et al. U-net: deep learning for cell counting, detection, and morphometry. *Nature methods*, 16(1):67–70, 2019.

[347] Jean-Baptiste Lugagne, Haonan Lin, and Mary J. Dunlop. Delta: Automated cell segmentation, tracking, and lineage reconstruction using deep learning. *PLOS Computational Biology*, 16(4):1–18, 04 2020.

[348] Robert E Kass and Adrian E Raftery. Bayes factors. *Journal of the american statistical association*, 90(430):773–795, 1995.

[349] Iacopo Ruolo, Sara Napolitano, Davide Salzano, Mario di Bernardo, and Diego di Bernardo. Control engineering meets synthetic biology: Foundations and applications. *Current Opinion in Systems Biology*, 28:100397, 2021.

[350] Johannes Schindelin, Ignacio Arganda-Carreras, Erwin Frise, Verena Kaynig, Mark Longair, Tobias Pietzsch, Stephan Preibisch, Curtis Rueden, Stephan Saalfeld, Benjamin Schmid, et al. Fiji: an open-source platform for biological-image analysis. *Nature methods*, 9(7):676–682, 2012.

[351] Cristian Versari, Szymon Stoma, Kirill Batmanov, Artémis Llamosi, Filip Mroz, Adam Kaczmarek, Matt Deyell, Cédric Lhoussaine, Pascal Hersen, and Gregory Batt. Long-term tracking of budding yeast cells in brightfield microscopy: Cellstar and the evaluation platform. *Journal of The Royal Society Interface*, 14(127):20160705, 2017.

[352] Stefan Van Dongen. Prior specification in bayesian statistics: three cautionary tales. *Journal of theoretical biology*, 242(1):90–100, 2006.

[353] Nathan P Lemoine. Moving beyond noninformative priors: why and how to choose weakly informative priors in bayesian analyses. *Oikos*, 128(7):912–928, 2019.

[354] Pandemic Influenza Outbreak Research Modelling Team (Pan-infORM et al. Modelling an influenza pandemic: A guide for the perplexed. *Cmaj*, 181(3-4):171–173, 2009.

[355] Sanae Lotfi, Pavel Izmailov, Gregory Benton, Micah Goldblum, and Andrew Gordon Wilson. Bayesian model selection, the marginal likelihood, and generalization. In *International Conference on Machine Learning*, pages 14223–14247. PMLR, 2022.

[356] Içvara Barbier, Rubén Perez-Carrasco, and Yolanda Schaerli. Controlling spatiotemporal pattern formation in a concentration gradient with a synthetic toggle switch. *Molecular systems biology*, 16(6):e9361, 2020.

Glossary

A-opt A-optimality. 48, 49

AIC Akaike information criterion. 42, 83, 142–144, 150, 155, 260, 261

aTc Anhydrotetracycline. 30, 60, 64, 66, 74, 86, 87, 93, 99, 101, 118, 119, 122, 126, 133, 135, 146, 148, 151–155, 168, 180, 191, 197, 198, 201, 207–210, 213, 215, 216, 219, 220, 222–225, 228, 234, 239, 245, 254, 302, 312, 314–317

BIC Bayesian information criterion. 42, 83, 143, 150, 155, 260, 261

CNN Convolutional neural network. xvi, 105, 161–164, 169

CRISPR-Cas9 Clustered Regularly Interspaced Short Palindromic Repeats-CRISPR-Associated protein 9. 11

D-opt D-optimality. 48, 49, 91, 92, 108, 175

DAW Dial-A-Wave. xi, xvi, 96, 98, 100, 103, 104, 106, 158–160, 265

DBTL Design-Build-Test-Learn. 13, 16, 20

DIC Differential interference contrast. 95, 100–106, 160–164, 167, 168

DNA Deoxyribonucleic acid. 11, 20, 65

E-opt E-optimality. 48, 49, 91, 92, 108, 175

eSS enhanced Scatter Search. 76, 89, 92, 107, 131, 146, 147

FIM Fisher information matrix. 4, 39, 47–49, 51, 85, 86, 91, 92, 108, 125, 128, 130, 132, 134, 137, 172, 175, 181, 259, 260, 263, 266

HMC Hamiltonian Monte Carlo. 78

IoU Intersection over union. 162, 164, 196, 197, 203, 302

IPTG Isopropyl ß-D-1-thiogalactopyranoside. 30, 60, 64, 66, 69, 74, 86, 87, 93, 95, 99, 101, 103, 115–119, 122, 126, 133, 135, 146, 148, 151–155, 167, 168, 175, 176, 180, 191, 197, 198, 201, 204, 207–211, 213, 217, 219, 222–225, 228, 234, 239, 245, 254, 276, 302, 312, 314–317

IVP Initial Value Problem. 33

MCMC Markov chain Monte Carlo. 41, 45, 78, 173, 178, 199, 232, 249, 255, 258

MINLP Mixed Integer Non-Linear Programming. 92, 107

MLE Maximum Likelihood Estimation. xvii, xxi, xxii, 78, 108, 109, 177, 178, 180, 199, 226–233, 235, 236, 238, 247, 248, 266, 313–323

mRNA Messenger ribonucleic acid. xix, 16, 64, 65, 69–72, 112–115, 126, 135, 136, 275, 281, 282, 284

nRMSE Normalised root mean square error. xxi, 83, 121, 122, 133, 142, 144, 150, 155, 213, 215, 222, 229–231, 235, 236, 261, 318, 326

NUTS No-U-Turn Sampler. 78, 255, 258

ODE Ordinary Differential Equation. 23, 24, 26, 30–33, 64, 65, 73, 84, 87, 112, 118, 221, 243, 244, 246, 254, 266

OED Optimal Experimental Design. xvii, xix–xxii, 3–6, 8–10, 47, 49–51, 71, 87, 88, 91, 92, 107–109, 146–148, 150–154, 156, 158, 170–181, 183–186, 188, 190–192, 194, 196, 198–223, 229, 231, 233, 235–237, 239, 240, 252–254, 257, 258, 261–266, 294–297, 311–313, 319–326

PDE Partial Differential Equation. 23

PDMS Polydimethylsiloxane. 18, 19, 93, 97, 98, 159

PE Parameter estimation. 132, 134, 171, 172, 175, 176

RMSE Root mean square error. 105

RNA Ribonucleic acid. 2

ROI Region of interest. 105, 106, 159–162, 164, 165, 167, 169, 172

SSE Sum squared error. 105