THÈSE
Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ SORBONNE PARIS CITÉ
Préparée à l'UNIVERSITÉ PARIS DIDEROT
Spécialité : biophysique
Ecole doctorale Interdisciplinaire Européenne Frontières du Vivant (ED 474)
Laboratoire Matière et Systèmes Complexes
INRIA Saclay, Ile-de-France

Real-time control of a genetic toggle switch

présentée par :

Jean-Baptiste LUGAGNE

Soutenue le 13 Décembre 2016 devant le jury composé de:

Hidde de Jong	Directeur de recherche, INRIA	Rapporteur
Sven van Teeffelen	Chargé de Recherche, Institut Pasteur	Rapporteur
Jérôme Bonnet	Chargé de Recherche, CNRS	Examinateur
Mario Di Bernardo	Professor, Université Frederico II, Italie	Examinateur
Meriem El Karoui	Professor, Université d'Edimbourg, Royaume-Uni	Présidente
Pascal Hersen	Directeur de Recherche, CNRS	Directeur de thèse
Grégory Batt	Chargé de Recherche, INRIA	Directeur de thèse

CC BY ND

Acknowledgements

I would like to thank all the interns who worked with me throughout this PhD. First of all, Agnes Köhler, Melanie Kirch, Catherine Eisenhauer and Jean-Baptiste Caron worked with me in close collaboration on the genetic control project presented here, and I benefitted greatly from their fresh views on the project. I would also like to thank Pierre Ivanovitch who worked with me on the image analysis method presented in this thesis and accomplished a formidable amount of work. I also collaborated with Rémi Sieskind, Amaury Monmeyran and Théo Maire who worked on a number of exciting side-projects.

I would of course like to thank all the members of the team and of the entire laboratory Matière et Systèmes Complexes for their help and all the fruitful exchanges we had, but also for the wonderful atmosphere in which I did my PhD. I would like to thank specifically Jannis Uhlendorf, Clément Vulin and Xavier Duportet for being excellent teachers, Zacchari Ben Meriem, Chiara Fracassi, Adrien Hallou, Fabien Furfaro, Max Piffoux and Marguerite Benony for their stimulating collaboration on a number of side-projects, and finally other members of the laboratory Artémis Llamosi, Zoran Marinković, Raphaël Ponthieu, Sara Bonavia, Adrien Marck and David Pereira for their support, both scientific and personal. Also a special thank you to Stéphane Pinhal and Antoine Decrulles who, while not being part of the lab, helped me get started with molecular biology and gave me excellent advice.

I greatly benefitted from the help, skills, and knowledge of technicians and engineers in the department, Carine Vias, Arnaud Grados, Laurent Réa, Matthieu Receveur, and Stéphan Suffit. Thank you also to Kévin Lhoste and Vladimir Hermand at the hackerspace and fab-lab l'Openlab.

I am very grateful for the exchange and collaborations I could have with other permanent researchers in the laboratory, namely Gaëlle Charron, Benoit Sorre and Jean-Marc di Meglio.

I would also like to thank all members of the Lifeware team at INRIA for all the very insightful discussions we had, especially François Fages and Sylvain Soliman.

Guy-Bart Stan and Hans Geiselmann were the two members of my thesis advisory committee and I thank them both for being excellent supports and mentors.

And, of course, I would like to deeply thank my two thesis advisors Pascal Hersen and Grégory Batt for their supervision, support and their invaluable contribution to the projects presented here. I was given a lot of freedom and support for experimentation, side-projects and collaborations, and I realize how exceptionally fortunate I was.

On a more personal level, I would like to thank the entire iGEM Grenoble team of 2011 and Diego Oyarzún for introducing me to synthetic biology and systems biology during my engineering studies.

I would finally like to thank my family and close friends for their unconditional support, open-mindedness and patience. An especially big thank you to Adélaïna, Alison, Anthony, Audrey, Camille, Cécilia, Fabien, Hélène, Léa, Luc, Lucien and Marion.

Table of Contents

Acknowledgements i

Table of Contents v

Abstract ix

Résumé xi

Foreword 1

Motivation: From steam to electronics to cybergenetics 2

Scientific problem: Long-term and real-time control of a bistable genetic system in single-cell bacteria 8

Approach 8

Contribution 9

Outline 11

Chapter I - State of the art 15

Genetic networks as dynamical systems 16

External control 17

The genetic toggle switch 25

Chapter II - Observing and actuating on trapped bacterial cells (materials and methods) 42

Introduction 43

Synthetic biology 45

Microfluidics & automation 59

Modeling 77

Conclusion 84

Chapter III – Toggle switch stabilization 93

Introduction 94

Characterization experiments 94

Fitted model 95

Single-cell control 102

Open-loop dynamic stabilization 108

Conclusion 112

Chapter IV – Stack segmentation 117

Introduction 119

Z-stacks and focal signatures 132
Training 134
Classification 140
Segmentation 148
Towards 3D segmentation 152
Conclusion 153

Chapter V – Conclusion and perspectives 161
Contributions 162
Limitations 163
Development of the control platform 164
Perspectives 166
Appendix A – List of plasmids and strains .....

Abstract

Recent progresses in microfluidics, synthetic biology, and microscopy automation now make it possible to control gene expression externally and in real time. Among the challenges facing the field of external real-time control of gene expression is the control of intricate, multistable gene regulation networks as well as the control of several target genes at the same time. To advance the domain in this direction we studied the controllability of a simple bistable two-genes network, the so-called genetic toggle switch, in the vicinity of its unstable equilibrium point for extended periods of time.

Throughout this document, we present the development of a custom control platform for external control of gene expression at the single-cell level as well as a bacterial cellular chassis and a library of toggle switch genetic circuits for us to control. We use the platform to drive and maintain our genetic system in its region of unstability with both closed-loop and open-loop strategies. Not only do we demonstrate that in silico control platforms can control genetic systems in out-of-equilibrium states, we also notably maintain a population of cells in their unstable area with open-loop periodic stimulations. These results suggest the possible emergence of different regimes of stability in gene regulation networks submitted to fluctuating environments, and can potential insights in the study of cellular decision making.

We also introduce a new approach for microscopy image analysis which exploits information hidden in several focal planes around the specimen instead of using only a single-plane image. The objective of this method is to automatically label different parts of an image with machine learning techniques inspired by hyperspectral imaging. The method is then shown to facilitate segmentation and be easily adaptable to various different organisms.

Keywords: Escherichia coli, Cybergenetics, Microfluidics, Control theory, Machine learning, Synthetic biology, Systems biology, Image analysis.

Résumé

Les progrès récents de la microfluidique, la biologie synthétique, la microscopie automatisée rendent aujourd'hui possible le contrôle externe de l'expression des gènes en temps réel. Parmi les défis que devra relever le domaine du contrôle externe et temps-réel de l'expression des gènes se trouve la possibilité de contrôler des réseaux de régulation génique aux dynamiques complexes et multi-stables, et le contrôle de multiple gènes en parallèle. Pour faire avancer le domaine dans cette direction nous avons étudié la contrôlabilité d'un réseau bistable composé de deux gènes, appelé genetic toggle switch, ou bascule génétique, autour de son point d'équilibre instable sur de longues périodes.

Dans ce document, nous présentons la mise en place d'une plateforme de contrôle externe de l'expression des gènes en cellule unique, ainsi que le développement d'un châssis cellulaire bactérien et d'une librairie de circuits de bascules génétique à contrôler. Le travail de développement technologique est présenté dans un premier temps : Une nouvelle implémentation en parallèle du système de microfluidique connu sous le nom de ‘mother machine’ y est présenté ainsi que le développement d’une solution de mélange d’inducteurs chimiques basée sur des valves interconnectées commandées en modulation de largeur d’impulsion sont présenté ainsi que des tests de validation. Le développement technique de l’automatisation de la plateforme de microscopie-microfluidique, notamment informatique, ainsi qu’une brève présentation de la stratégie d’analyse d’image en temps réel impliquée, sont discutés. La construction d’une librairie de toggle switch bistables basés sur la double répression LacI-TetR et différents par leurs force de liaison ribosomales est également détaillée ainsi qu’un nombre d’innovations techniques, nécessaires à cette fin, de la technique de clonage ‘MoClo’. Enfin, la caractérisation du système ainsi que le développement d’un modèle quantitatif, à la fois déterministe et stochastique, sont présentés afin de pouvoir à la fois estimer la position du point d’équilibre instable de notre toggle switch, et prédire la faisabilité de notre objectif de contrôler le toggle switch génétique autour de son point d’équilibre instable.

Une fois opérationnelle, nous utilisons la plateforme pour diriger et maintenir notre système génétique dans sa région d'instabilité avec des techniques de stabilisation à la fois en boucle ouverte et en boucle fermée. Dans un premier temps, une seule cellule est contrôlée et est entraînée vers le point d'équilibre instable du toggle switch, grâce à l'utilisation d'un contrôleur proportionnel-intégral. En augmentant le gain proportionnel de ce contrôleur nous observons que le reste de la population peut être entraînée. Dans le cas extrême d'un contrôleur bang-bang, cas limite du contrôleur proportionnel-intégral où le gain est infini, toute la population est entraînée avec la cellule effectivement contrôlée. Nous démontrons donc non seulement que les plateformes de contrôle in silico peuvent être utilisées pour contrôler un système génétique dans un état hors-équilibre, nous démontrons aussi la possibilité de maintenir une population de cellules dans leurs région d'instabilité à l'aide de stimulation dynamiques en boucle fermée. En étudiant la dynamique sous-jacente de ce phénomène populationnel grâce aux modèles théoriques développés et calibrés sur nos données, nous suggérons la possibilité d'un phénomène de stabilisation dynamique semblable au phénomène du pendule de Kapitza. Cette étude indique, lorsque les cellules sont soumises à des stimulations périodiques de fortes concentrations d'inducteurs, un changement dans le régime de stabilité du système et transforme la zone instable du toggle switch en attracteur global. Nous démontrons ensuite expérimentalement ce phénomène et l'observons pour une gamme de stimulations périodiques. Ces résultats suggèrent l'émergence de régimes de stabilité différents dans des réseaux de régulation génique lorsqu'ils sont soumis à des environnements fluctuants, et peuvent fournir de nouvelles perspectives dans l'étude de la prise de décision cellulaire. Notamment, de telles fluctuations périodiques sont observées dans des circuits de différentiations de précurseurs neuronaux en amont d'une prise décision cellulaire.

Enfin nous présentons aussi une nouvelle approche pour l'analyse d'images de microscopie qui exploite l'information cachée dans plusieurs plans focaux autour du spécimen au lieu d'utiliser seulement un seul plan focal. En effet, lorsque plusieurs images de microscopie sont acquises, pour un même objet, dans plusieurs plan focaux différents, l'intensité d'un même pixel va varier de façon différente selon la nature de l'objet présent sur ce pixel, pour des raisons de propriétés liées à la matière de l'objet ainsi qu'à sa forme et à la diffraction qu'elle entraîne. Par conséquent, cette « signature » de l'objet observé permet de l'identifier. Il s'agît donc d'un problème de classification, et nous utilisons des techniques classiques d'apprentissage machine, les machines à support de vecteurs, pour classifier ces signatures en fonction de la nature de l'objet observé. Nous démontrons la possibilité d'ainsi identifier dans des images de microscopie différentes cellules, comme la bactérie E. coli, la levure S. cerevisiae, ou des cellules cancéreuse humaines de type HeLa. Nous pouvons aussi identifier les sous-parties de ces cellules comme le cytoplasme ou la membrane, mais aussi identifier des structures de microfluidique ou des contaminants. La méthode facilite ensuite la segmentation de l'image et peut être facilement adaptée à différents organismes.

Mots-clés : Escherichia coli, Cybergénétique, Microfluidique, Automatique, Apprentissage machine, Biologie synthétique, Biologie des systèmes, Analyse d'images.

Foreword

Table of Contents

1 Motivation: From steam to electronics to cybergenetics. 2

1.1 Control Theory 2

1.2 Control engineering and biology 4

1.3 Externalizing Control 6

2 Scientific problem: Long-term and real-time control of a bistable genetic system in single-cell bacteria 8

3 Approach 8

4 Contributions 9

5 Outline 11

1 Motivation: From steam to electronics to cybergenetics.

1.1 Control Theory

As S. Bennet reminds us at the beginning of his book series A History of Control Engineering (Bennett 1979) the will to control natural elements and make them self-regulate dates back thousands of years. In Politics, Aristotle already understood the potential of what would later be called "control theory":

"...if every instrument could accomplish its own work, obeying or anticipating the will of others [...] if the shuttle weaved and the pick touched the lyre without a hand to guide them, *chief workmen would not need servants, nor masters slaves."*

Control theory is the science of regulating and driving physical processes to accomplish tasks that are important to us. It is used somewhat interchangeably to describe automatic feedback mechanisms in which the state of a process, for example, the water level in a tank, is first sensed, e.g. with an ultrasonic sensor or, more simply, a floater, and the process is acted upon on-the-fly to make it reach a desired state, such as, by opening or closing valves. The applications of control theory are ubiquitous in our day-to-day life; perhaps recently one of the most sensational being the technological prowess of the successful landing of SpaceX's falcon 9 rocket[1]. But aeronautics and astronautics are not the only domains in which control theory is of central importance; applications include abstract tasks such as monetary regulation in central banks or smart power grid management, to more mundane examples like air conditioning.

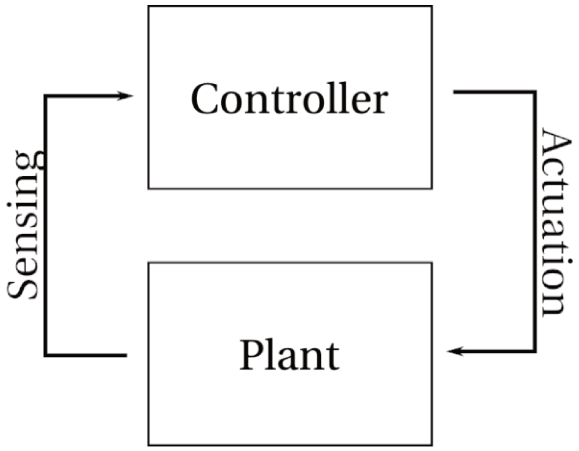

Figure 1-1 The general scheme of feedback control. A process of interest, usually called a "plant", is regulated in real time by a control apparatus. The state of the process is sensed through different sensors and a control algorithm computes the strategy to follow by influencing the process via various means of actuation to steer it in the desired direction.

It is in ancient Greece that antique implementations of control devices started appearing. The earliest documents relating water clocks and pneumatic control date back more than 23

[1] The Falcon 9 rocket, designed and manufactured by the company SpaceX, is a two-stage-to-orbit launch vehicle, designed for commercial cargo delivery into orbit. Its key feature is the use of a re-usable main stage, which can land autonomously after detachment of the payload, to reduce costs. On December 21st 2015, the company successfully landed the main stage for the first time, in what is seen as a major step forward and a technological tour de force.
http://www.theverge.com/2015/12/21/10640306/spacex-elon-musk-rocket-landing-success

centuries ago. Heron of Alexandria's float valve regulator for water level control can still be found in every modern toilet water tank. These early implementations spread through the Greek and Arab worlds and control theory remained largely confined to pneumatics and water level control until the 17th century. After the Renaissance, European scientists rediscovered the classical Hellenic knowledge and soon Cornelius Drebbel implemented the first thermostated incubators. But perhaps the most important landmark in early control theory history is the centrifugal governor. Invented by Thomas Mead in 1787(Mead 1787), it was soon applied by James Watt and Matthew Boulton to rotary steam engines to regulate their speed (see Figure 1-2).

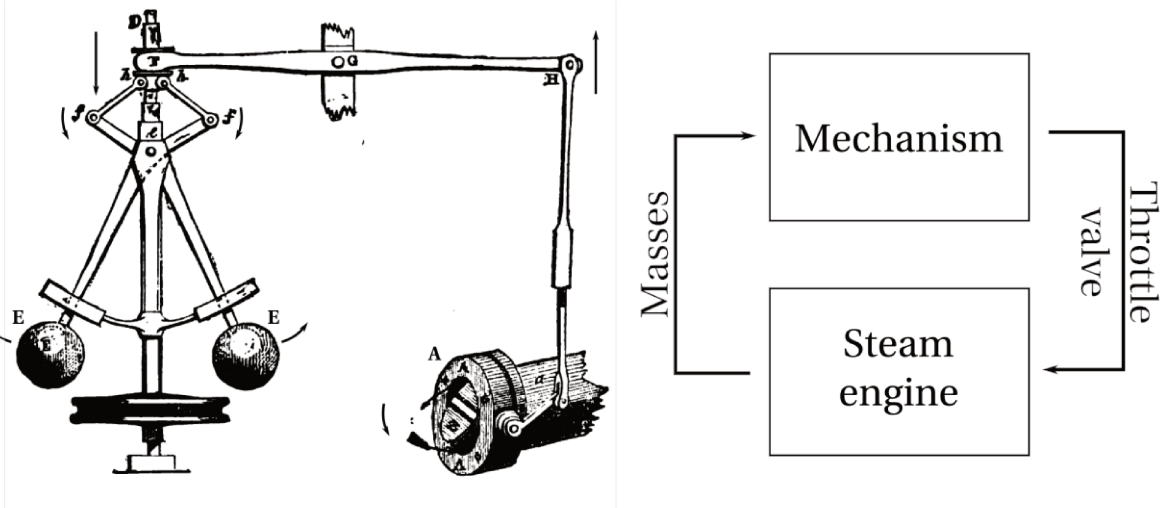

Figure 1-2 Adapted from (Routledge 1881). Watt & Boulton's centrifugal governor - The two masses (E) spread outward when the rotational speed applied to the main shaft (D) increases, thus actioning a mechanism that will reduce the aperture of the steam throttle-valve (A) and finally reduce the rotational speed. The centrifugal governal was the central object of study of early control theory in the 19th century. It is difficult to distinguish the sensor from the controller and from the actuator in those early implementations, but the "plant" is the steam engine, or the rotational speed out of a steam engine.

The centrifugal governor played a major role in the industrial revolution. The possibility of not only harnessing steam, but also regulating it, led to unprecedented sustained power and left its mark in every contemporary scientific mind. Perhaps it should not be too surprising to us then that, in an analogy perfectly tailored to the purpose of this thesis, Charles Darwin and Alfred Russel Wallace wrote about the principle of evolution in 1858 (Darwin & Wallace 1858):

*"The action of this principle is exactly like that of the centrifugal gover*nor of the steam engine, which checks and corrects any irregularities almost before they become evident; and in like manner no unbalanced deficiency in the animal kingdom can ever reach any conspicuous magnitude, because it would make itself felt at the very first step, by rendering *existence difficult and extinction almost sure soon to follow."*

The importance of centrifugal governors in 19th century technology, from trains to steamships to hydraulic pumps, drew scientific attention to their improvement. Because of the lack of any integral error compensation, errors in the choice of the masses would almost always cause offset error. Nonlinear effects generated limit cycles. And also the steady-state speed could not be modified on-the-fly. These problems could be circumvented with differential gears,

springs, and a variety of ingenious mechanisms, but soon engineers ran into stability issues. Throughout 19th century Europe interest in the mathematical analysis of the stability of those systems increased, laying the first theoretical framework for control theory.

From then on, theoretical analysis would foster the development of control devices even further. New needs for precise positioning, steering engines, and various stabilizers on one hand, and the advent of powerful combustion engines and electrical sensors, actuators, and signal processors on the other, also promoted widespread deployment of automatic feedback mechanisms, but it wasn't until after World War II that the various analytical approaches would be unified under a common theory and convention. The Cold-War era rocket frenzy, the exponential growth of computing power, and the third industrial revolution would make control theory into the established and ubiquitous science it is today.

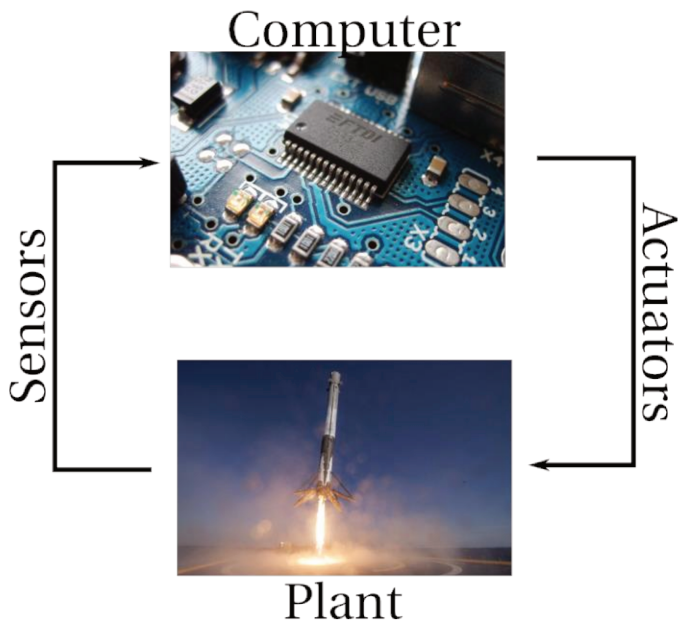

Figure 1-3 Modern implementation of control feedback loops. The "plant" can be a chemical reaction, a temperature to regulate or a modern rocket. Various electronic or electromechanical sensors, like pH meters, thermometers or accelerometers, will feed data about the process to the computer, or at least the part of the electronic circuit that implements the control algorithm. Finally, the state of the electronic or electromechanical actuators, like valves, heaters, or motors, is modified in real time to steer the process in the right direction.

1.2 Control engineering and biology

In 1948, the book Cybernetics: or Control and Communication in the Animal and the Machine was published. The result of a collaboration between control theory pioneer Norbert Wiener at the Massachussetts Institute of Technology and Harvard Medical School physiologist Arturo Rosenblueth, the book became a reference in the field and beyond, defining the term of cybernetics as a transdisciplinary approach for exploring regulatory systems, their structures, constraints, and possibilities. The book exposes the possible contribution of the discipline to a variety of scientific domains. Of particular interest to us, it laid bare the connection between early control theory and a number of physiological regulations, like body homeostasis or biomechanical feedback control.

Indeed, since the 1950s the study of prehension, locomotion, and motor coordination in animals and human beings has relied heavily on a control theory approach(Todorov & Jordan

2002): It is generally accepted that the tensions in the many muscles implied in any movement, to grasp a pencil for example, are updated throughout the entire process according to sensory inputs (e.g. vision, touch, and proprioception) and advanced control algorithms taking place at all levels of the nervous system. Similarly, in physiology, following the seminal paper by Grodins (Grodins et al. 1954) on the subject, the study of body homeostasis including thermoregulation, blood pressure, blood pH, and the levels of calcium, potassium, sodium and glucose in the blood were all studied from the angle of control theory(Milhorn 1966).

The discovery of DNA, and more importantly, the identification of genetic regulation mechanisms in 1961(Jacob & Monod 1961) opened the way to the study of the inner workings of cells themselves. Refinements in protein sensing technology and in quantitative biology over the years unveiled the staggering complexity of the genetic-proteomic circuitry. Gene networks have since become one of the main subjects of study of systems biology, and the seminal works of Alon, Barkai and Leibler(Barkai & Leibler 1997; Alon et al. 1999) in the late 1990s first suggested and demonstrated that adaptation to constant stimuli by the bacterial chemotaxis system is a robust property of the network and not a result of fine-tuning of biochemical parameters. In the following years, control theoreticians looked into the interplay between network topologies and the robustness of their responses. A review on the subject can be found in (Stelling et al. 2004).

The early 2000s also saw the appearance of another related field: synthetic biology. Synthetic biology stems from the progress made in genetic manipulation tools in the late 1980s and 1990s (most notably in cloning techniques, PCR and DNA sequencing) and systems biology. It aims at integrating an exogenous genetic network in the cell that not only can provide the genetic information for synthesizing new proteins, as is routinely done in traditional genetic manipulations, but also incorporates a function or program encoded into the genetic material to respond to certain external or internal stimuli or to increase the robustness of the desired process. Potential applications include drug production and smart drug delivery, biofuel synthesis, bioremediation, biosensing, and waste processing (Collins & Khalil 2010; Cameron et al. 2014). Because synthetic biology draws a significant part of its goals from computer science and electrical engineering, control theory was rapidly incorporated into the fundamental concepts that drove its development. The difficulty of designing synthetic genetic modules to assemble into more elaborate systems became rapidly apparent, and soon engineered circuit modules involved feedback loops to increase robustness and reliability. For a detailed description of the co-evolution of synthetic biology and control theory see (Del Vecchio et al. 2016).

Unfortunately the processing subtleties that can be involved in modern control algorithms can hardly ever be incorporated into genetic circuitry, and a new theoretical framework for in vivo internal stochastic control algorithms must be formulated to reach the long-term objectives of synthetic biology. But looking at the history of control theory, the trend over the post-war era has been to externalize the control process out of the electromechanical realm into the more computation-indulgent digital one. Key technological advances that made this shift possible were the development of reliable, precise, digital sensors on one hand, and efficient and

powerful electromechanical actuators on the other, with of course the exponentially increasing computational power of modern microprocessors. In the years to come, the expansion of in vivo external real-time control to new applications will be fostered by: a) renewed interest in fluorescent probe design following the recent breakthrough of super-resolution imaging; b) the actuation capacity provided by the sophistication of optogenetics over the last decade; and c) continued innovation in microfluidics both for observation and actuation through chemicals delivery.

1.3 Externalizing Control

Inducible promoters are now routinely used in systems and synthetic biology to understand and construct genetic regulatory networks. Because of the stochasticity of gene expression and its limited measurability in cells, such uses have long remained limited to short term, full induction of populations of cells. Fine-tuning of in vivo protein levels in bacteria remains a challenge.

Before using drug-responsive promoters like the lac or tet systems(Lutz & Bujard 1997) to induce gene expression in living cells, the first technique to study the action of a gene in an organism was complete gene knockout. Not only is it a demanding task, but the process is not reversible, not quantitative, and one cannot observe the effect of the gene's removal on the rest of the genetic network dynamically. A step towards quantifiable and dynamic studies of gene expression was made with the introduction of phage promoters in the cells, such as P_L of phage lambda(Elvin et al. 1990) or the RNA polymerase and promoter of T7 phages(Studier & Moffatt 1986). Again, those techniques are not convenient and hardly quantitative: it required heavy cloning procedures where all the phage transcription machinery had to be expressed by the cell, and the induction level of the promoters could not be fine-tuned over time, as the P_L promoter is activated by temperature shifts and T7 promoter activation necessitates the introduction of the T7 RNA polymerase in the cell via phage infection. Finally, during the 1990s, a lot of effort was put into the development and characterization of inducible promoters that could be quantitatively tuned, induced over wide ranges of expression, and would be orthogonal to most of the cells' natural repressors(Guzman et al. 1995; Skerra 1994). The most widely used systems are the lac and tet promoter-repressor couples(Lutz & Bujard 1997). These inducible systems opened the way for quantitative studies of gene regulatory networks and later, synthetic biology. However, although these newer inducible promoters allow for a more finely graded induction of gene expression levels than the original on/off phage-based systems, and refinements based on self-regulating feedback loops make this resolution even finer(Nevozhay et al. 2009; Rosenfeld et al. 2002), induced gene expression is still unpredictable, heterogeneous, and shows significant fluctuations in time.

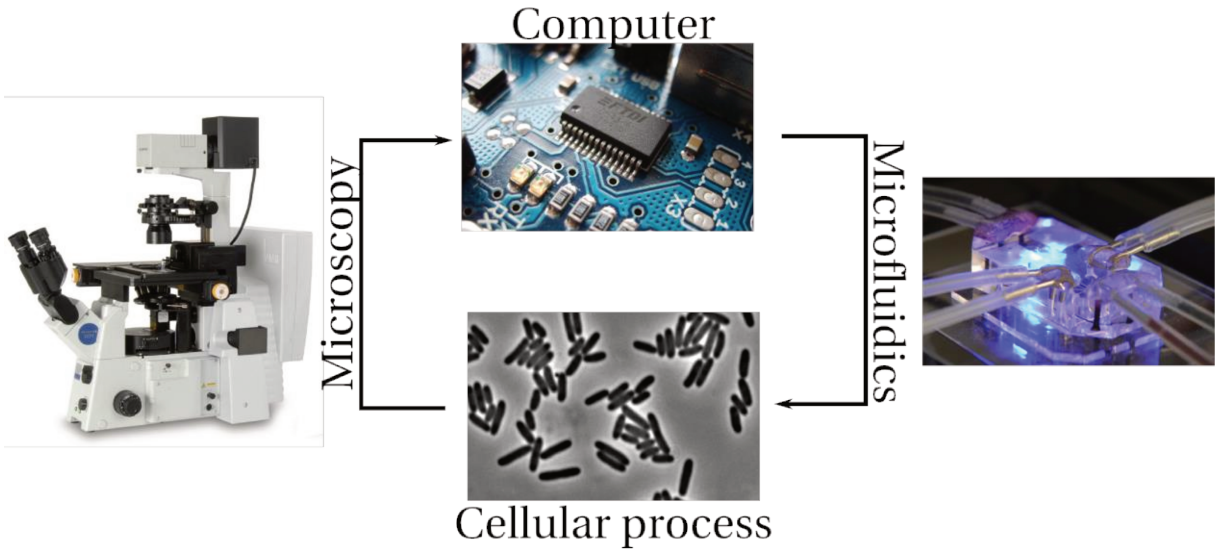

Figure 1-4 General description of our platform for external control of intracellular processes. The "plant" to control is the expression level of a gene of interest. Time-lapse, epifluorescence movies of the cells are acquired and the data is sent to a computer for evaluating the control strategy. The environment of the cells is then modified via microfluidics accordingly to steer gene expression in the appropriate direction.

In the State of the Art section of this thesis I will detail recent results that show that gene expression levels can be precisely manipulated over extended periods of time using control theory, with in silico feedback loop control platforms. Externalizing the control system allows for more sophisticated control strategies and a broader range of possible dynamic perturbation, or programming, of the system of interest. With precise control of gene expression, the signal processing properties of a genetic circuit can be analyzed by measuring its dynamic input-output function. The circuit can then be used as a brick in a larger assembly and used to drive chemical processes of interest or as a precise input knob on other cellular processes.

Multistable systems play a central role in both synthetic and natural gene regulatory networks, and it is important for the applicability of control theory to demonstrate that such systems can be controlled in a large portion of their state space. A particularly good point for external control would be to demonstrate that cells can be maintained for extended periods of time in states that cannot be maintained by constant induction of inducible promoters. Such a result would extend the domain of possible study of cellular phenomena, since it would allow the experimenter to bring the system into states that could not be reached or maintained with previous methods, and without extensively modifying the underlying genetic network.

The development of external control of gene expression platforms in conjunction with other developments in the fields of microfluidics, optogenetics, and fluorescent probes should lead to the routine use of external control in the years to come. This is the central problem of my PhD work. I built an experimental platform for the real time control of gene expression and put significant efforts into making sure that it can be used by others.

With this single-cell control platform for bacteria, I investigated the controllability of a simple bistable system, the genetic toggle switch.

2 Scientific problem: Long-term and real-time control of a bistable genetic system in single-cell bacteria

The main objective of my PhD was to develop a single cell control platform to study the feasibility of control in single cell Escherichia Coli bacteria, and specifically control of a bistable gene circuit, the toggle switch. It implied studying and integrating together the different domains that are necessary for a functional control platform: actuation through synthetic biology and microfluidics, sensing through microscopy and image analysis, and control theory.

The genetic toggle switch consists in two genes mutually repressing each other, and features two stable states in which either of the two genes is expressed and completely represses the other. A third, unstable, equilibrium point in which the two genes expression levels are such that their equivalent repression strengths are balanced is predicted by deterministic models of the system. However in practice the cells are always in either of the two stable states and can sometimes switch between the two due to random fluctuations. My objective was to maintain Escherichia coli cells harboring this circuit in their unstable area by dynamically changing their environment.

This genetic toggle switch control problem bears several similarities with the benchmark control problem of the inverted pendulum and allowed me to investigate the potential offered by external control theory on single-cell control. The genetic toggle switch is also a fundamental topology in core natural gene regulatory networks as well as one of the foundational results of synthetic biology and as such is frequently used in complex synthetic circuitry when bistability, memory or binary signal processing is desired. Hence, studying the dynamics of the toggle switch, especially in its unstable area, where it is rarely observed, was also one of the goals of this thesis. Combining a hands-on approach made possible by external cell control with a theoretical study of its dynamics made it possible to study the behavior of a multistable gene regulatory network out of equilibrium, as it is perturbed by time-varying perturbations that modify its phase portrait, thus allowing us to further the study of multistable genetic systems and raise new questions about their implications in stochastic cell fate determination, commitment and its reversibility, and the extent of dynamic or periodic stimulations in natural gene networks.

3 Approach

To make single-cell control of a multistable genetic system possible, I started by designing a control platform for single-cell bacteria. I developed several microfluidic chips to contain Escherichia coli cells and be able to both observe single cells for extended durations, and programmatically and rapidly change their environment. I also worked on image analysis to extract long-term single cell fluorescence levels, and experimented with state-of-the-art segmentation algorithms before developing my own image segmentation algorithm. I also developed a modular set of programs to integrate the different software parts of the control platform together and completely automate all hardware elements.

In parallel I developed Escherichia coli strains to minimize interferences between the host cell and the lac and tet systems I used to control gene expression. I iterated on the chassis strains while trying to control simple transcription units. I developed a library of 36 dual-reporter LacI-TetR toggle switches, and picked the best of them to conduct characterization experiments to identify the parameters of a model inspired by other works in the literature. I used both deterministic and stochastic simulations based on this model to assess its controllability and study different control strategies and implementations.

Once platform development, biological implementation, and theoretical study reached maturity, I performed control experiments under various control strategies. The knowledge garnered from these successful control experiments on the behavior of toggle switch in its unstable area allowed me to identify a family of open-loop dynamic stimulations that would invert the stability map of the system.

4 Contributions

The first, practical, contribution of this thesis is the development of a platform combining hardware and software elements from different disciplines, to achieve long-term tracking, real-time actuation, and external control on single-cell bacteria. This platform combines microfluidics, microscopy automation, image analysis, and control. The platform offers the possibility to control the expression level of a specific gene with minor genetic manipulations and minimal knowledge of the underlying process. It is also highly modular: the implementation of the control platform consists of different processes that communicate with each other, so that parts of the control loop, like image analysis or control strategy computation, can be modified independently and adapted to different situations, and also be distributed over several machines if more computational power is needed. We believe this platform, in combination with other works in the field, will open the way to a new level of precision in the study of gene regulatory networks dynamics.

Because the platform allows for single-cell measurements, we were able to use it on an in vivo stochastic bistable system: the genetic toggle switch. The genetic toggle switch can be viewed as the genetic equivalent to the benchmark problem in control theory that is the inverted pendulum problem. Multistable systems are widespread in all dynamical systems and their control and controllability has always been an important subject of study. Multistability can be undesirable: it can force discretization of responses, with a limited number of accessible states linked to the attractors of the system, when one might want to investigate its intermediate, graded responses. But it can also be advantageous in systems where mutually-exclusive tasks are attached to specific states of the systems, like in differentiation. In all cases, controlling multistable systems is a generally more difficult problem than controlling linear or non-linear monostable systems. It is an important milestone in any control implementation, and demonstrates the potential of our control framework.

Beyond the simple demonstration of the potential of control theory, multistability plays a central role in the genetic circuitry of all organisms, and is a core element of any advanced

synthetic biology circuit. The toggle switch in particular is present in many gene regulatory networks involved in cellular decision making, and has been used in a number of synthetic circuits where strong bistable filtering of genetic information was necessary (see State of the Art for a detailed analysis). The ability to control multistable systems opens the way to high precision setting of cellular state, even allowing for the study of intermediate states out of equilibrium before the system is fully committed. We also demonstrate the possibility of stabilizing the system in areas that are inaccessible with traditional induction techniques, thus expanding the area of investigation on gene regulatory networks. In particular, it allowed us to investigate the dynamics of the toggle switch in the vicinity of its unstable steady-state.

Another contribution of this thesis is multiple inputs-multiple outputs (MIMO) gene control. MIMO systems are also a common subject of study in control theory, since most systems consist in intricate, multivariate interactions, where more than one input knob can help better steer the system, and one might want to control more than one of its outputs. The genetic toggle switch is a perfect example of a non-streamlined genetic circuit where MIMO control is necessary. In a more general sense, MIMO control of gene expression should improve the performance of GRNs (Gene Regulatory Network) control.

This work is also the first theoretical and experimental study of dynamic stabilization in an externally-driven gene regulatory network. The study of cellular response to dynamic stimulations is still in its infancy, with studies of time-varying concentrations of morphogens inputs on embryo development and patterning booming in the late 2000s and early 2010s (see Kutejova et al. 2009 and Sorre et al. 2014 for discussions of those results). The results presented here illustrate the importance of out-of-equilibrium study of gene regulatory networks dynamics, which can lead to unexpected results, even from well-documented network topologies. The model developed to study the dynamic response of the toggle switch to external stimuli gave valuable insights into how multistable systems react to dynamic stimulations and as such can be used to further study the behavior of this fundamental network.

One of the tools I developed, Stack-based image analysis, can be used beyond the scope of single-cell bacteria segmentation, and can be transparently applied to other organisms without any software modifications. This method proposes the use of machine learning methods on z-stacks of images above and below focus to identify regions of the images for cell segmentation. Cell segmentation is one of the core difficulties of single-cell control, and although recent developments in the field of Convolutional Neural Networks (CNN) are promising, no cell segmentation algorithm has given satisfactory results for online, long-term cell tracking yet. The use of high-dimension information hidden in Z-stacks combined with traditional segmentation algorithms and newer deep learning techniques can lead to fast advances in online image analysis of microscopy images, with possible further developments in 3D segmentation.

Finally, all the tools I developed, especially the control platform, will be released in open-source and open-hardware formats, and my data will be made available, following a

movement in modern science of transparency and exchange that I believe will be beneficial to science in general and to the field of external control in particular.

5 Outline

My thesis document is organized as follows:

Chapter 1 will present the state of the Art, detailing both the recent development of external real-time control of gene expression, and the central role of double negative feedback loops, or toggle switches, in both natural and synthetic gene regulatory networks.

In chapter 2, I present the main techniques and implementation choices for controlling the genetic toggle switch. I discuss the development of Escherichia coli chassis strains to circumvent problems caused by cellular processes interfering with the control inputs. I report the various constructs I developed for obtaining a bistable switch and the changes in implementation that occurred throughout my work. I also present the implementation of the control platform, with various obstacles and solutions for microfluidic chip fabrication, inputs delivery, image analysis, cell containment and experiment automation. I close this chapter by discussing the modeling of the toggle switch and its simulation, both deterministic and stochastic, as well as the control implementation choices and the parameters identification approach.

In chapter 3, I present the main results of this thesis, both numerical and experimental, and present their implications on our understanding of the toggle switch dynamics and on genetic multistability in general.

In chapter 4, I present a novel image analysis and cell segmentation algorithm inspired by hyperspectral imaging. The results extend well beyond bacteria cell segmentation and holds great promises for online cell segmentation, and can be combined with other techniques in cell segmentation, especially recently developed deep neural network for cell segmentation.

Chapter 5 concludes this thesis.

The Appendix present additional data and development made during this thesis.

Alon, U. et al., 1999. Robustness in bacterial chemotaxis. Nature, 397(6715), pp.168–71.

Barkai, N. & Leibler, S., 1997. Robustness in simple biochemical networks. Nature, 387, pp.913–917.

Bennett, S., 1979. A history of contro engineering, 1800-1930, Institution of Electrical Engineers Stevenage, UK.

Cameron, D.E., Bashor, C.J. & Collins, J.J., 2014. A brief history of synthetic biology. Nature reviews Microbiology, 12(5), pp.381–90.

Collins, J.J. & Khalil, A.S., 2010. Synthetic biology: applications come of age. Nature reviews. Genetics, 11(5), pp.367–379.

Darwin, C. & Wallace, A.R., 1858. On the Tendency of Species to form Varieties; and on the Perpetuation of Varieties and Species by Natural Means of Selection.

Elvin, C.M. et al., 1990. Modified bacteriophage lambda promoter vectors for overproduction of proteins in Escherichia coli. Gene, 87(1), pp.123–126.

Grodins, F.S. et al., 1954. Respiratory Responses to CO2 Inhalation. A Theoretical Study of a Nonlinear Biological Regulator. Journal of Applied Physiology, 7(3), pp.283–308.

Guzman, L.M. et al., 1995. Tight regulation, modulation, and high-level expression by vectors containing the arabinose PBAD promoter. Journal of bacteriology, 177(14), pp.4121–30.

Jacob, F. & Monod, J., 1961. Genetic regulatory mechanisms in the Synthesis of proteins. Journal of molecular biology, 3, pp.318–356.

Kutejova, E., Briscoe, J. & Kicheva, A., 2009. Temporal dynamics of patterning by morphogen gradients. Current Opinion in Genetics and Development, 19(4), pp.315–322.

Lutz, R. & Bujard, H., 1997. Independent and tight regulation of transcriptional units in Escherichia coli via the LacR/O, the TetR/O and AraC/I1-I2 regulatory elements. Nucleic acids research, 25(6), pp.1203–10.

Mead, T., 1787. A regulator for wind and other mills.

Milhorn, H.T., 1966. Application of Control Theory to Physiological Systems, Saunders (W.B.) Co Ltd.

Nevozhay, D. et al., 2009. Negative autoregulation linearizes the dose-response and suppresses the heterogeneity of gene expression. Proceedings of the National Academy of Sciences of the United States of America, 106(13), pp.5123–8.

Rosenfeld, N., Elowitz, M.B. & Alon, U., 2002. Negative Autoregulation Speeds the Response Times of Transcription Networks. Journal of Molecular Biology, 323(5), pp.785–793.

Routledge, R., 1881. Discoveries and inventions of the nineteenth century, London ; New York : G. Routledge.

Skerra, A., 1994. Use of the tetracycline promoter for the tightly regulated production of a murine antibody fragment in Escherichia coli. Gene, 151(1–2), pp.131–135.

Sorre, B. et al., 2014. Encoding of Temporal Signals by the TGF-β Pathway and Implications for Embryonic Patterning. Developmental Cell, 30(3), pp.334–342.

Stelling, J. et al., 2004. Robustness of Cellular Functions. Cell, 118(6), pp.675–685.

Studier, F.W. & Moffatt, B.A., 1986. Use of bacteriophage T7 RNA polymerase to direct selective high-level expression of cloned genes. Journal of molecular biology, 189(1), pp.113–30.

Todorov, E. & Jordan, M.I., 2002. Optimal feedback control as a theory of motor coordination. Nature neuroscience, 5(11), pp.1226–35.

Del Vecchio, D., Dy, A.J. & Qian, Y., 2016. Control theory meets synthetic biology. Journal of The Royal Society Interface, 13(120).

Chapter I
State of the art

Table of Contents

1 Genetic networks as dynamical systems. 16
2 External control 17
2.1 External control of gene expression – first results 2011-2012 17
2.2 External control of gene expression – 2014-2016 21
3 The genetic toggle switch 25
3.1 Double repressive loops in natural gene regulatory networks 26
3.2 The genetic toggle switch in synthetic biology 27
3.3 Theoretical studies of the genetic toggle switch 29
3.4 Theoretical control of the toggle switch 31

1 Genetic networks as dynamical systems.

As the study of biological systems becomes increasingly quantitative, ecosystems, organisms and genetic circuits are studied more and more with tools from dynamical systems theory. This evolution was the result of improved measurement techniques over several decades. In the dynamic study of gene regulation networks, the shift occurred throughout the 1990s and led first to the establishment of the field of systems biology (Alon 2007; Kitano 2002), and then to the field of synthetic biology (Benner and Sismour 2005; Endy 2005). Although the two fields are connected and mutually benefit from one another, the former favors a top-down approach to the question and focuses on large-view studies of gene networks made of smaller, finer detailed subunits (or systems), while the latter, synthetic biology, favors a bottom-up approach in which hands-on studies of minimal synthetic genetic networks are used to gain understanding on gene expression by mimicking nature (Di Ventura et al. 2006). The quality of the dynamical data acquired on the evolution of genetic circuits increased thanks to developments in several fields I the 1980s and 1990s. First of all, as discussed in the introduction of this thesis, the development of new genetic manipulation techniques (Sambrook and Russell 2003), inducible genetic systems (Lutz and Bujard 1997) and intracellular fluorescent probes (Shaner, Steinbach, and Tsien 2005) in this period made the assembly, steering and observation of such systems possible. But the study of the dynamics of those systems was still rudimentary, and biological microfluidics, in combination with automated timelapse microscopy, was developed subsequently to facilitate the study of gene regulation networks in time and at the single cell level. Microfluidic chips are microfabricated devices in which low volumes of fluids are moved, mixed, separated or otherwise processed to miniaturize procedures with applications in various domains of physics, chemistry and biology. Microfluidic rapidly became a key element in the study of the dynamics of synthetic gene regulation networks because it could be used both to trap cells and hence monitor their fluorescence over many generations, possibly at the single cell level, but also because it could be used to change the environment of the cells dynamically (Bennett and Hasty 2009). With this type of setup, inducible synthetic systems could be steered and observed in real time, and the study of their dynamic response to time-varying perturbations was used to gain a deeper understanding of the temporal mechanics of genetic systems. (Bennett et al. 2008; Hersen et al. 2008; Mettetal et al. 2008) are seminal studies that used microfluidic devices to create temporal changes in the growth medium to study dynamic biological phenomena. In (Bennett et al. 2008; Hersen et al. 2008) the authors used periodic osmotic shocks to study the frequency response of the High Osmolarity Glycerol pathway in yeast and used those results to identify the mechanisms of the cellular response to shocks at different timescales. In (Bennett et al. 2008) the authors also studied the frequency response in yeast to periodic changes in the external carbon source by using a microfluidic platform. The galactose utilization network is shown to filter out fast fluctuations in the nutrients source, but to slowly adapt to long-term evolutions. In this paper the interplay between mathematical modeling, dynamic experiments and biology is elegantly illustrated by the discovery of a previously-unknown post-transcriptional regulation loop in the network because of a discrepancy between the model-predicted dynamics of the system and experimental data.

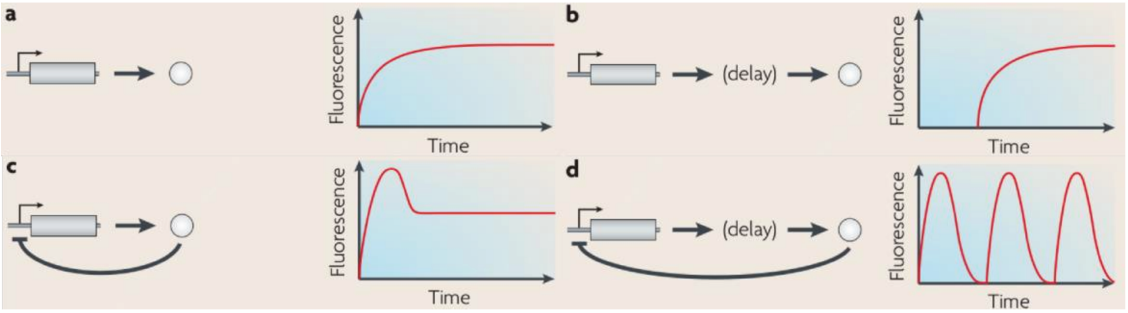

Figure 1-1 From (Bennett and Hasty 2009) – The study of the dynamics of gene expression can reveal hidden processes in the system. A simple one-gene system can produce various types of temporal profiles for its level of expression, and the study of its dynamics can help discriminate between the different possible models for its expression.

The response of biological systems to fluctuations of their growth environment has brought deeper understanding of the dynamic properties and structure of fundamental genetic network implied in the response to environmental change. But precise manipulation of elements of the networks themselves to study the effects on the rest of the system has the potential to be both more informative and have a larger scope of study (Del Vecchio, Dy, and Qian 2016). With microfluidics, automated time-lapse microscopy, and progresses in automated image analysis, closing the loop for performing real-time control of intracellular processes became possible. In the following section I present recent results made in this direction.

2 External control

The field of external control of intracellular processes, or in silico cybergenetics[1], is born from the desire to bring the dynamical study of genetic network to a higher level of scrutiny. The underlying idea is to dissect the dynamic response of gene regulation networks by shifting the stimuli from external changes to the environment to arbitrary fluctuations within the networks themselves. The field is still in its infancy, and the first experimental results date back to 2011. Since then a number of theoretical and experimental studies were published, in a variety of organisms, using different control approaches, different methods of actuation, and different cell cultures. In this section I will analyze and compare the work done so far in external feedback control of gene expression.

2.1 External control of gene expression – first results 2011-2012

Three studies can be considered as seminal in the field (Milias-Argeitis et al. 2011; Toettcher et al. 2011a; Uhlendorf et al. 2012). Although all three studies have different objectives and use different approaches to implement the control framework, they all follow the same general approach. First of all, the cells are minimally modified to actuate and/or observe the genetic system of interest. They all achieve complex manipulations of intracellular processes with simple genetic manipulations of the controlled system, where tedious circuit development would have been required to create an equivalent single-usage, intra-cellular

[1] The term was coined by Prof. Mustafa Khammash during a plenary session at the IFAC'14 conference, in a reference to the term cybernetics which describes the study of regulatory systems, in domains like mechanics, engineering, physics, biology, cognitive sciences, social sciences or economics (see Wiener 1948 for the original definition of cybernetics).

function. Secondly, even though the different studies differ in their cell culture apparatus, measurement method and actuation, the feedback loop is implemented in silico and the level of refinement of the feedback function is much higher than any possible internal implementation of genetic regulation. Finally, those different studies achieve levels of control of their respective systems that open-loop stimulation based on models and predictions could not have achieved.

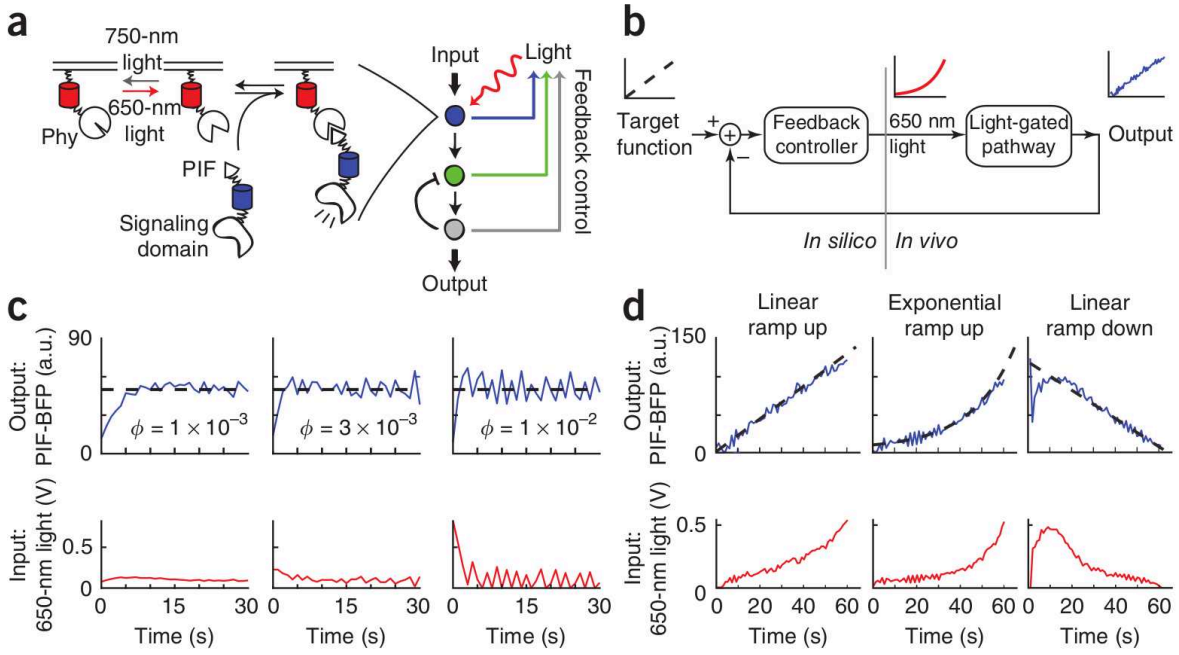

Figure 2-1 From (Toettcher et al. 2011a). a) Schematic of the Phy-PIF recruitment optogenetic system. Under different light conditions, the Phy domain switches between a closed state and an open state that triggers recruitment of the PIF-tagged proteins. The feedback control loop senses membrane recruitment by image analysis, and applies the appropriate amount of light to control the amount of recruitment at the membrane. b) The schematic of the feedback control loop. The user specifies a target function, and the controller compares it to the observed cell level and corrects the light input accordingly. c) Setpoint control results for different feedback parameters. The blue curves represent the binding of the PIF-tagged protein to the membrane. The red curve is the amount of light input sent to the cells by the controller. The black dashed line represents the objective. d) Control experiments of time varying profiles.

The first one (Toettcher et al. 2011b) to be published focused on controlling a signaling pathway by first driving fluorescence recruitment at the membrane of HeLa cells, and then using the method to recruit the PI 3-kinase responsible for the synthesis of phosphatidylinositol-3-phosphate (PIP3) at the membrane and thus control the location of those lipids. The PI 3-kinase is attached to the Phy-PIF optogenetic[2] system (Toettcher et al. 2011a) for actuation and the production level and location of the lipids are monitored through an engineered fluorescent proxy: PHAkt-Cerulean, which binds to the produced lipids (see Figure 2-4b). Recruitment and fluorescence levels are monitored via epifluorescence microscopy. Optogenetic membrane recruitment is activated by shining light on cells observed. Because of the dynamics of the different systems involved, the time resolution of their acquisitions is in the order of seconds, and their experiments typically last about 10 minutes. Because of this short timescale and the optogenetic actuation, the study did not

[2] Optogenetics is a biological technique which involves the use of light-sensitive proteins to control intracellular processes, such as gene expression, protein recruitment, or enzymatic activity.

require a complex cell culture and input delivery platform and the experiments were performed in liquid cultures on a microscope. A proportional-integral controller connects the observation and actuation in silico. Proportional-integral (PI) controllers are controllers that correct an error between a measured output of the system of interest and a target level by changing the input to the system both proportionally, i.e. by a quantity that is proportional to the current error, and integrally, i.e. by a quantity that is proportional to the integral of the error over time. Several procedures exist to fine-tune PI controllers to minimize oscillations and overshooting (Åström and Hägglund 2006), but they all apply to linear systems. For the class of nonlinear systems we are interested in here, those techniques can be used as a starting point but their results usually need to be refined empirically. The authors show that not only are they able to control membrane recruitment and the synthesis of PIP3 with a PI controller, at setpoint levels and in time-varying profiles, they also reduce cell-to-cell variability in recruitment resulting from non-uniform expression of the optogenetic system.

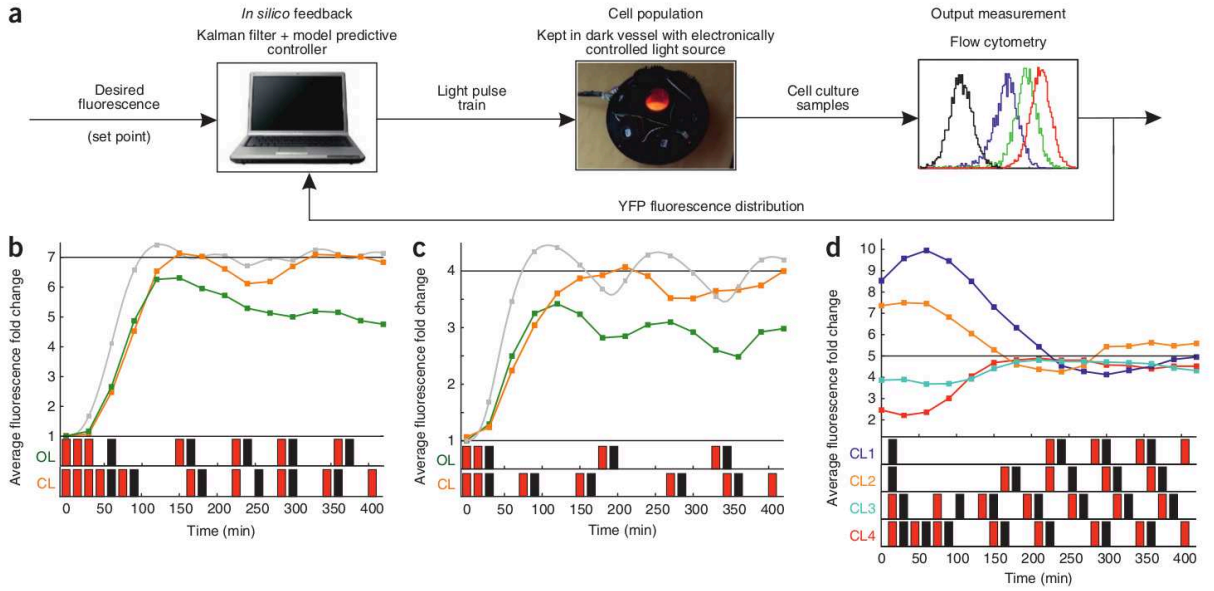

Figure 2-2 From (Milias-Argeitis et al. 2011). a) The in silico feedback control loop. A batch of cells are illuminated in red and far-red light to drive a synthetic genetic system. b-c) Regulation of the fluorescence level around the setpoint (black line). The controlled system, in orange, is compared to a numeric simulation of the cells, in gray, and to an open-loop dynamic stimulation. The red and black bars at the bottom represent the light pulses in red and far-red, respectively. d) Setpoint control from different random initial points.

In the second study (Milias-Argeitis et al. 2011), the authors use a control platform to drive the expression of a synthetic system (see Figure 2-4a) in a population of Saccharomyces cerevisiae cells. The expression of a fluorescent protein downstream of an another Phy-PiF-based optogenetic system (Tyszkiewicz and Muir 2008) is controlled around a setpoint by using pulses of light in a batch culture of yeast cells. In liquid batch apparatuses, cells grow in suspension in a liquid media, under constant shaking and temperature control. However since the cells are growing in large volumes in a flask, the sensing must be done on a sample of the cell culture because the traditional instruments used for fluorescence measurement are not adapted for in situ use in large volumes of cell culture. Here the state of the cells is measured periodically by flow cytometry and used by the control algorithm. A model-predictive control approach is chosen to implement the feedback loop between measurement and actuation. In

model predictive control, or MPC, a dynamic model of the system is used to predict how the process will react to series of possible inputs, and the best course of action is applied until the next acquisition. This type of approach thus requires a priori knowledge about the system to control, but typically yields better results than PI controllers. A state estimation algorithm is also used in combination with the model-predictive controller to estimate the state of the non-observable variables that are used by the model-predictive controller. The authors demonstrate the possibility to control the gene expression level at various levels of expression for several hours, and also the possibility to drive the system after different random perturbations to the state of the cells.

Finally, in (Uhlendorf et al. 2012) the authors use the natural high osmolarity glycerol (HOG) pathway in Saccharomyces cerevisiae to control gene expression downstream of it (see Figure 2-4c). The HOG pathway is a phosphorylation cascade, and one of the genes that are activated downstream of it is STL1 (Rep et al. 2000). The authors used its native promoter to drive the expression of a fluorescent protein by submitting the cells to pulses of high-osmolarity media. The level of the fluorescent protein is observed in real time and at the single-cell level by time-lapse microscopy and automated image analysis, and the cells are grown in a microfluidic chip in which the osmolarity of the growth media flown through the chip can be changed in real time. In microfluidic chips the problem of cell sampling and measurement is greatly simplified, and the same population of cells, or even single cells, can be followed over extended periods of time. Again a model-predictive control approach is chosen to numerically close the loop. The authors use their platform to perform setpoint control experiments and also to force the cells fluorescence to follow time-varying profiles for up to 17 hours. They demonstrate that pre-computed open-loop dynamic stimulation cannot control the system as accurately. The microfluidic and time-lapse components of the platform allowed for both population and single-cell control. The individual cells were segmented through a circular Hough transformation. The authors use their platform to demonstrate the feasibility of control at the single cell level, but they also use the single-cell data acquired study the effect of population and single-cell control on noise levels.

Together, these studies demonstrate that real-time control can be used to robustly drive intracellular processes in real-time, dynamically limit the effects of gene expression stochasticity, or counter the effects of endogenous feedback loops. Different types of intracellular processes are controlled, in different organisms and with different actuation and sensing method, demonstrating the versatility of external control of gene expression.

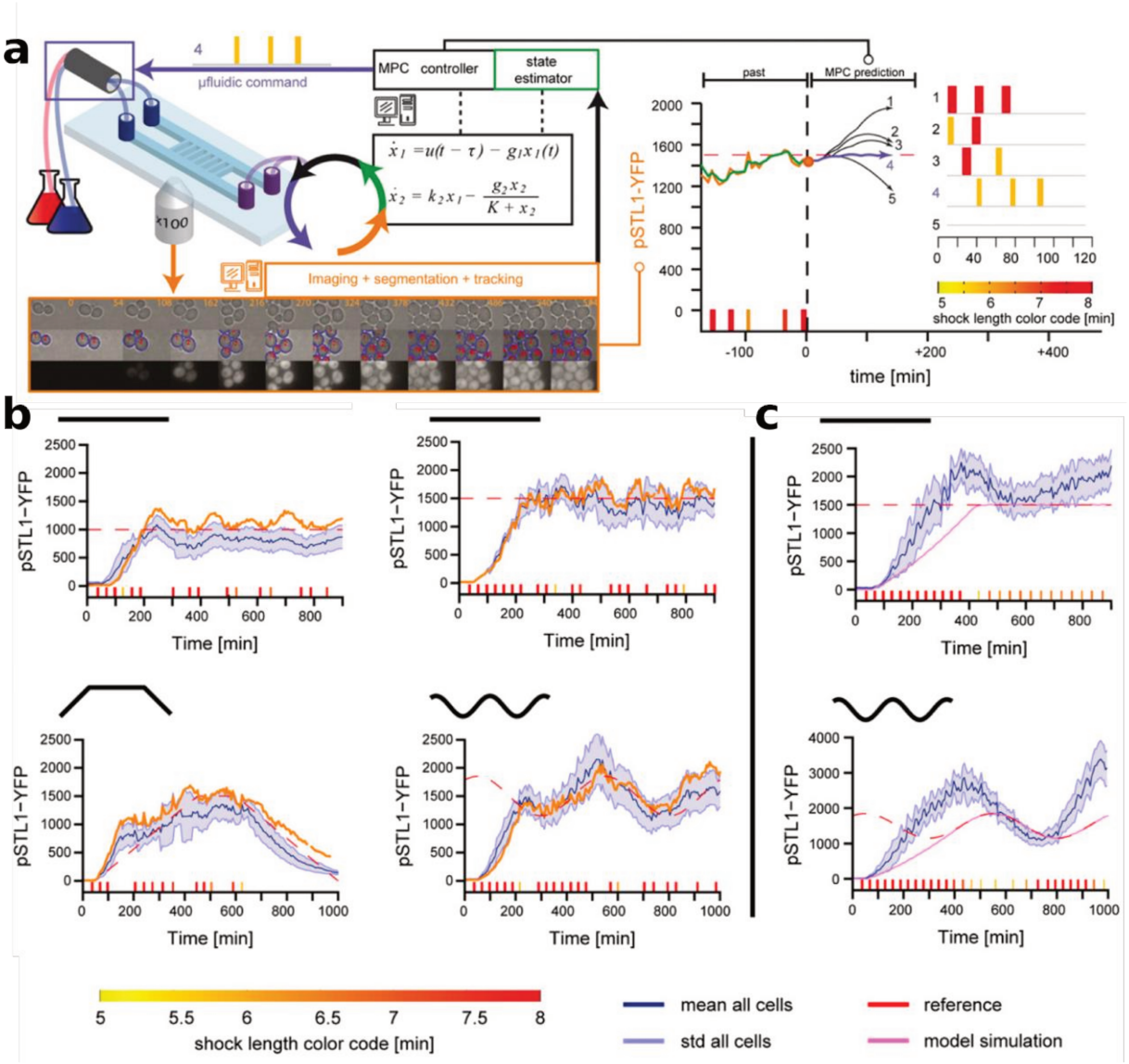

Figure 2-3 From (Uhlendorf et al. 2012). a) Yeast cells are cultured in microfluidic chambers and observed via epifluorescence microscopy. Individual cells are segmented and tracked over time. State estimation followed by model predictive control are applied to the extracted single-cell fluorescence levels, and the environment of the cells is modified in real time in the microfluidic device to steer the process of interest accordingly. b) Setpoint and time-varying control experiments. The dashed red line represents the objective, and the orange line represent the fluorescence of the single controlled cell. The dark blue line represents the population average and the light-blue enveloppe is the standard deviation of the population. c) Open-loop stimulation of gene expression. The light inputs to apply to the cells was computed prior to the experiment to follow a certain trajectory (dashed red line). The average population levels and the standard deviation are represented in blue.

2.2 External control of gene expression – 2014-2016

Since then a number of refinements to the implementation of external feedback loops have been developed, and the approach has been extended to new organisms and processes.

In (Menolascina et al. 2014) the authors use a platform based on a microfluidic apparatus to observe in real time via epifluorescence microscopy and actuate on their system via chemical inducers delivery. The authors control the expression of a fluorescent protein downstream of a synthetic network in yeast named IRMA (Cantone et al. 2009, see also Figure 2-5e). The same team proposed one year later on the same platform, but on a simpler genetic system, a range of control approaches and compared their performance (Fiore et al. 2015). The authors

use galactose and glucose to turn on or off the expression of the GAL1 promoter to control the fluorescence level of GFP. On top of the previously-used control algorithms, the authors propose zero average dynamics control. Zero average dynamics (ZAD) is a type of sliding-mode control algorithm. It produced results comparable to model-predictive control. In (Fracassi et al. 2016) the same team developed a microfluidic platform for long term control of gene expression in mammalian cells. A fluorescent protein under a synthetic tetracycline-inducible system is controlled over extended periods of time (up to 58 hours).

In (Melendez et al. 2014) the authors use an approach similar to (Milias-Argeitis et al. 2011) but improve the automation of the platform and add a turbidostat to the cell culture apparatus, i.e. they control the density of cells in the flask. They also use a bang-bang control approach: It is similar to the PI control approach, but does not apply intermediary inputs to the system it tries to control. Instead the system is fully activated or inhibited when it is respectively below or above its objective. With this setup the authors are able to drive protein expression for up to 45 hours.

Finally, in (Milias-Argeitis et al. 2016) the authors expand on the approach described in (Milias-Argeitis et al. 2011) and develop a similar platform for bacteria. Additional effort is put on the automation of the platform and the authors can precisely control gene expression but are also able to control the growth rate of the cells by actuating on the vital synthesis of methionine (see Figure 2-4d). The sensing of the population density is done by optical density directly in the flask and the growth rate is derived from it. This result demonstrates the possibility offered by in silico control of gene expression to manipulate fundamental functions with minimal modifications to the cell.

All of these improvements on the principle of in silico control platforms show on one hand how much potential the field has. Control theory is a broad discipline and all the knowledge gathered over the decades of existence of the field can be used to drive the development of those control platforms. On the other hand, it also shows how recent this field is. So far, all applications of the approach remain limited to proofs of concepts on the feasibility of the method on new toy systems or organisms or on the comparison of performance between different control strategies. Although the ultimate goal is to use control to induce precise intracellular perturbations to study the response of the rest of the cell, the field is still being explored. The knowledge necessary to apply this methodology to biological question still needs to be acquired and open questions on the potential of the field remain. During this PhD I worked on the question of external control of multistability in bacteria, by studying stabilization of a bistable genetic system in its unstable area.

Closely related to the dynamical study of gene expression, recent advances in iterative experiment design (Ruess et al. 2015) could be combined with control platforms to further automate the study of intracellular processes: Instead of having the experimenter run control experiments and then decide on what perturbations to run next, the data could be analyzed on-the-fly by iterative experiment design algorithms which would, as a higher-level layer, dictate control profiles to the control platform to maximize the quantity of information acquired, thus completely automating the process.

Apart from the dynamic study of genetic systems, external control of gene expression could also be applied to other domains of study. Being able to drive intracellular processes of interest can also of course find applications in biotechnology. In this domain, external feedback control has already been suggested to improve productivity, robustness and batch-to-batch reproducibility (Polizzi and Kontoravdi 2015). In (Milias-Argeitis et al. 2016) the authors already discuss the scalability of their approach to industrial bioreactors. Although some of the studies presented here were performed in batch cultures (Melendez et al. 2014; Milias-Argeitis et al. 2011, 2016) of yeast or bacteria and are therefore closer in their implementation to bioreactors, scaling up the size of the batch culture raises a number of issues, one of them being the difficulty to stimulate cells with optogenetic tools in the dense, opaque molasses that can typically be found this kind of apparatus.

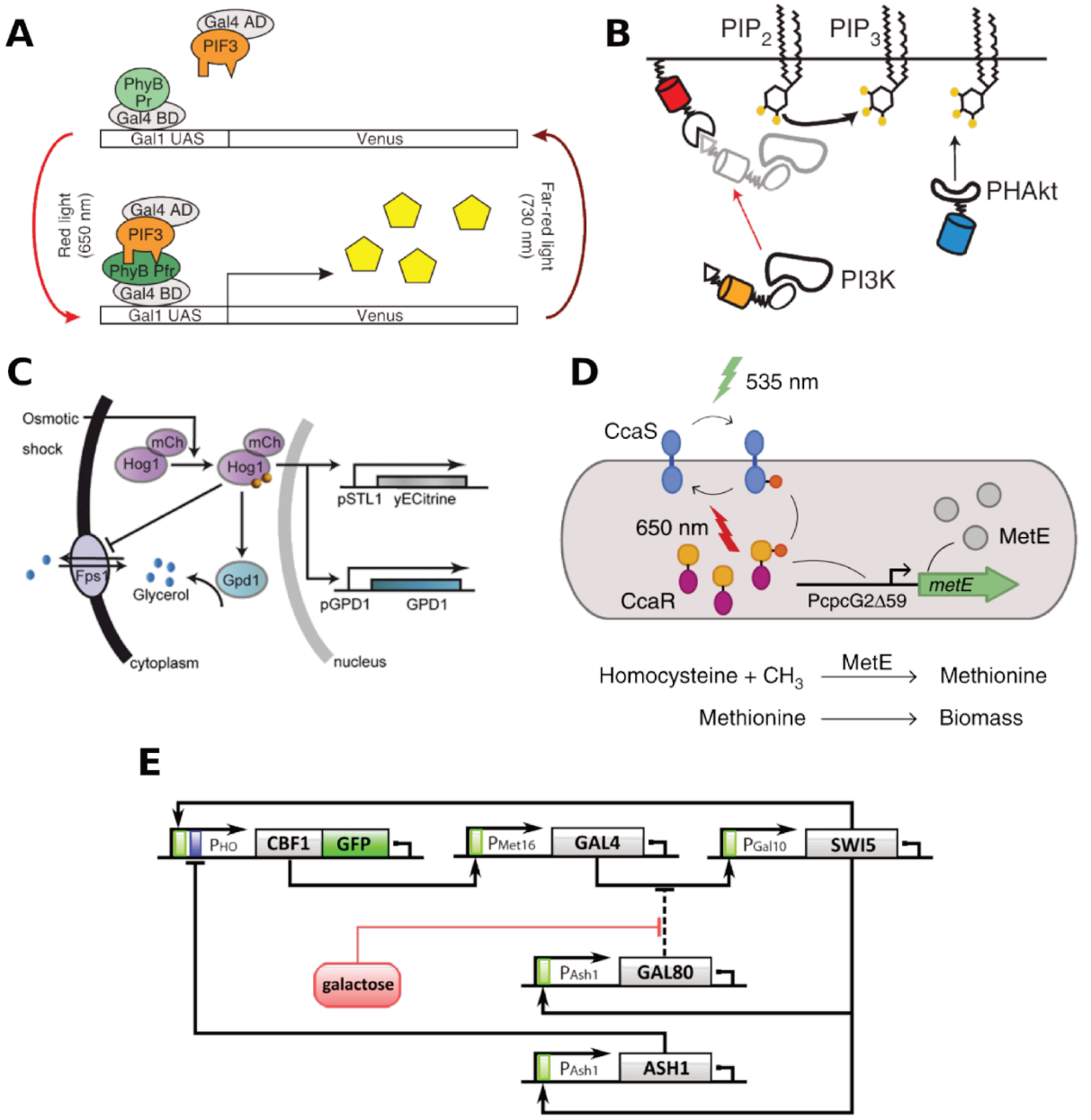

Figure 2-4 Examples of controlled circuits. A) From (Milias-Argeitis et al. 2011). The authors control the fluorescence level of a light-switchable gene system based on PhyB-PIF3 interaction in Saccharomyces cerevisiae. The transcription activation domain of Gal4 is recruited to the promoter under red light and induces transcription of the fluorescent reporter, Venus. Far-red light causes dissociation of the activator. B) From (Toettcher et al. 2011b). PI3K is recruited to a membrane target under 650nm light input and catalyzes the production of 3′ phosphoinositide (PIP3) lipids at the membrane of mammalian cells. The authors control the PI3K lipid products level by observing the membrane translocation of a fluorescent Akt-PH domain that binds to PIP3 lipids. C) From (Uhlendorf et al. 2012). The authors induce osmotic shocks by injecting glycerol in the environment of Saccharomyces cerevisiae cells to control the expression of a fluorescent reporter under a shock-responsive promoter downstream of the HOG cascade. D) From (Milias-Argeitis et al. 2016). The authors use the CCaS-CCaR optogenetic system to activate or repress expression of the MetE enzyme with green or red light inputs. MetE catalyses synthesis of methionine which is necessary for cell growth and division. The authors use this optogenetic system to control cellular growth rate in Escherichia coli via turbidity measurements. E) From (Menolascina et al. 2014). The IRMA network, implemented in Saccharomyces cerevisiae, is composed of 5 genes encoding for transcription factors modulating the expression of each other. The authors use galactose injection via microfluidics to drive the expression of the GFP reporter downstream of network.

3 The genetic toggle switch

So-called genetic toggle switches consist of two genes repressing each other expression. As a result of this double negative feedback, the genetic circuit features two stable states in which one of the two genes is expressed at a high level and completely represses the expression of the other.

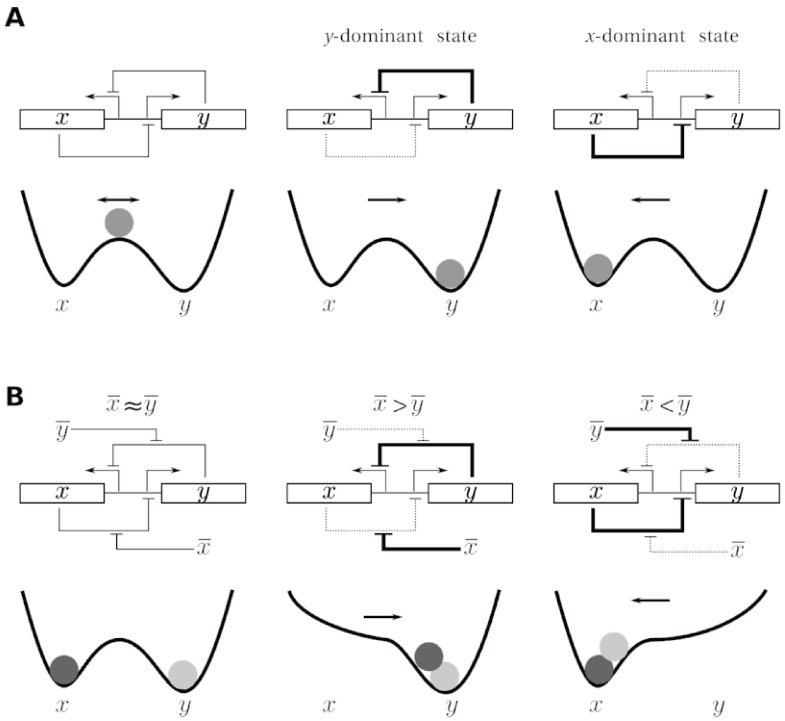

Figure 3-1 Principle of the genetic toggle switch topology. A) Two genes x and y repress each other's expression. This genetic circuit is bistable: either x or y can take over and be expressed at high concentrations. The repression of the opposite gene "locks" the system in the x-dominant or y-dominant state. The system can fall in either basin of attraction. It remains in a basin of attraction unless noise or perturbations to system make it switch basins. B) External inducers can be used to switch the system between the two states. The inducers can modify the repression strength of one of the two genes and make the system monostable.

Double negative regulation topology has been known to play a central role in many natural gene regulatory networks involved in cellular memory, stability or cellular decision-making. The construction of a genetic toggle switch was first published in 2000 and is largely viewed as one of the foundational results of synthetic biology. It has been employed since then in various synthetic biology projects. Toggle switches have been investigated from a systems biology perspective due to the frequent presence of simple bistable topologies in natural and synthetic genetic circuits, and studies on the toggle switch, both as a toy example for benchmarking algorithms or as a central theme of study, is abundant. Recently, with the advent of techniques for external control of gene expression, several studies focused on the interplay between a genetic toggle switch and a number of control frameworks. In this section I present the scientific literature investigating the toggle switch. I will first present examples of natural toggle switches in gene regulatory networks in a variety of organisms, and then describe its crucial role in synthetic biology. Finally I will present theoretical studies of its dynamics, with a section on external control of the toggle switch.

3.1 Double repressive loops in natural gene regulatory networks

Simple bistable networks can be found in numerous organisms, at the heart of important internal cellular processing functions. Examples of complex cellular processes driven by simple bistable circuits include cell fate determination, cellular memory, bet-hedging, nutrient selection, and tissue patterning. The double-negative regulation of the toggle switch is not the only simple gene network topology to exhibit bistability (Siegal-Gaskins et al. 2015) and single-positive, self-activating loops have been shown to be at the heart of many bistable gene regulatory networks. See (Dubnau and Losick 2006; Ferrell 2002, 2012) for reviews on the subject of positive feedback bistability in prokaryotes and eukaryotes.

Nevertheless, the genetic toggle switch can be found in a wide range of organisms and at different levels of genetic cellular organization. Although not really an organism, the most studied and first-discovered example of a genetic toggle switch was found in the bacteriophage lambda lysis-lysogeny decision making process (see Oppenheim et al. 2005 for an extensive review). In this network the CI and Cro proteins, driving the lysogenic and lytic responses respectively, mutually repress their expression. In higher eukaryotes, several double-repressive toggle switches have been demonstrated to drive cellular decision making and cell differentiation (Johnston et al. 2005; Mikeladze-Dvali et al. 2005), tissue patterning (Collier et al. 1996), as well as fundamental cellular processes like the cell cycle (Yao et al. 2008), and cellular proliferation (Lu et al. 2013).

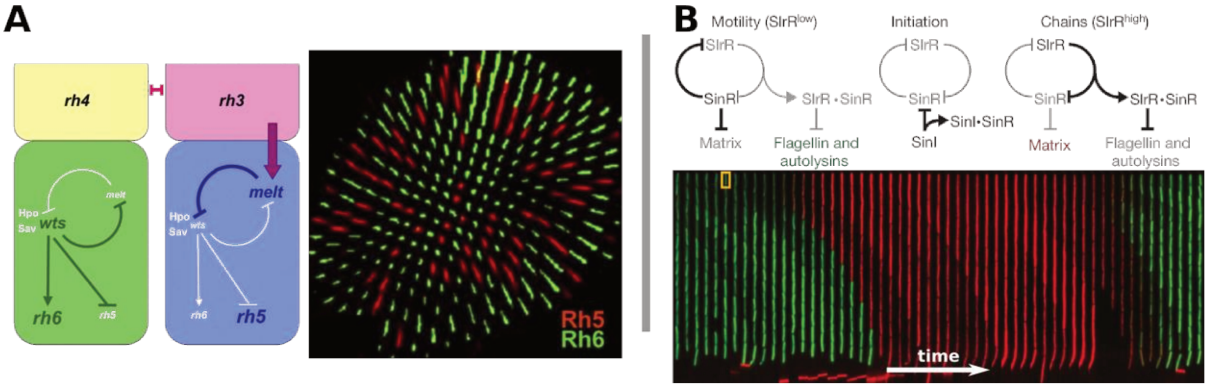

Figure 3-2 Examples of toggle switch topologies in natural genetic circuits. A) Adapted from (Mikeladze-Dvali et al. 2005). A double negative feedback loop between the wts and melt genes is responsible for cell fate determination of retinal cells in Drosophila. B) Adapted from (Norman et al. 2013). The SlrR – SinR double negative feedback loop stabilizes Bacillus subtilis cells in either a motile or a chained state. The cells switch periodically between the two states in what is believed to be a bet-hedging strategy.

More closely related to my project, prokaryotes also exhibit bistable behaviors caused by genetic toggle switches. An excellent example can be found in a Bacillus subtilis cellular decision mechanism, where a double-repressive loop is at the center of a fundamental behavior (Norman et al. 2013). The cells alternate between a motile state in which they swim independently and a so-called “chained” state where they grow and divide as usual but daughter cells do not separate completely and remain attached to each other. This second state allows the rapid formation of biofilms if the environment becomes hostile to the cells. Stochasticity of gene expression switches cells at random times into the chained state. On the other hand, chained cells resume the motile state after about eight generations, so that chaining does not condemn the cells to lower nutrient sources forever, and that at all times a

portion of chains of optimal length exist in the colony. This constitutes a prime example of cellular decision making and bet-hedging[3] in bacteria driven by a toggle switch.

3.2 The genetic toggle switch in synthetic biology

Gardner and Collins published the first synthetic genetic toggle switch in bacteria (Gardner, Cantor, and Collins 2000). They implemented two versions, a temperature-IPTG sensitive one, and the aTC-IPTG sensitive toggle switch I worked on in my thesis. They show that after the toggle has been induced into one state or another, it remains in that state until the opposite inducer is added to the growth medium. It is worth noting that only one of the two branches is monitored directly with a fluorescent reporter. Later implementations will use a reporter on both branches to monitor the state of the system, making model fitting and parameters identification more difficult. The system was soon adapted to mammalian cells with an antibiotic-regulated switch that showed bistability even after encapsulation and implantation in mice (Kramer et al. 2004). The toggle switch was then given its pivotal role in synthetic biology as a building block by being interfaced with other synthetic and endogenous elements to drive biosensing, biofilm formation, cell-density-dependent gene activation (Kobayashi et al. 2004), improvement of metabolic biosynthetic productivity of biofuels (Anesiadis, Cluett, and Mahadevan 2008), or to monitor changes in the mammalian gut environment (Kotula et al. 2014). Advances in modeling and prediction of synthetic circuits behavior eventually led to automated design of toggle switches in silico and accurate prediction of their stability (Chen et al. 2012; Ellis, Wang, and Collins 2009). The study by Chen and colleagues in 2012 constitutes, to my knowledge, the only example of a dual-reporter toggle switch in the literature, however the authors did not use it to fit a dynamic model of the toggle switch.

The toggle switch has since gone through major implementation modifications. A major change in bistability implementation in synthetic circuits occurred in the 2000s when (Ham et al. 2006) designed a recombinase switch which, while not exactly equivalent to a toggle switch in its topology, leads to strong bistability and low leakage. It does not rely on transcription factors to inhibit the expression of a gene, but instead uses invertases that will flip a promoter's orientation to stably turn two gene's expression fully on or off. The principle can even be generalized to an n-states switch by using orthogonal invertases since the number of possible states evolves exponentially with the number of recombinases (Ham et al. 2008). This type of system has been used in synthetic biology since then as an advanced, robust and reversible memory module (Bonnet, Subsoontorn, and Endy 2012) and to perform computational tasks within the cells (Moon et al. 2011; Yang et al. 2014). However such systems have an all-or-none nature to them and are not well-suited to the control problem we are interested in. Another exotic implementation of the genetic toggle switch is based on RNA interference in mammalian cells (Deans, Cantor, and Collins 2007), which allows for specific gene targeting and is used to control various biological processes.

But the most important change in gene regulatory network design and construction in the last few years was brought by the recent revolutions in transcriptional regulator design with the

[3] Bet-hedging is the process by which, in some species, a variety of different phenotypes exist within a population of to ensure that a subpopulation is well prepared for environmental changes. The phenomenon is well documented in antibiotic resistance and biofilms formation in bacteria (Lewis 2007).

CRISPR/Cas9, zinc fingers, and TALE systems. The possibility to design custom, orthogonal transcription factors calls for the reconstruction of custom versions of pioneering synthetic biology circuits. However none of those techniques has so far been engineered for cooperative binding of DNA, as is seen in the naturally occurring transcriptional regulators used in traditional toggle switches. The bistability of the toggle switch and the functioning of many other synthetic circuits rely on the nonlinearities introduced in their dynamics by the cooperative binding of multimeric transcription factors such as TetR or LacI. However, in (Lebar et al. 2014), the authors design and implement a toggle switch based on the TALE system and solve the non-cooperative binding problem by introducing nonlinearities through the competitive binding of 2x2 TALE assemblies fused to both activator (VP16) and repressor domains (KRAB). They argue that their approach can be adapted to CRISPR/Cas and zinc fingers to obtain nonlinearities in custom orthogonal synthetic circuits. Another approach for the possible use of custom monomeric DNA-binding domains derived from the CRISPR/Cas in multistable gene regulatory networks can be found in (Cuba Samaniego et al. 2016) which relies on molecular titration to introduce strong nonlinearities. The prospect of designing

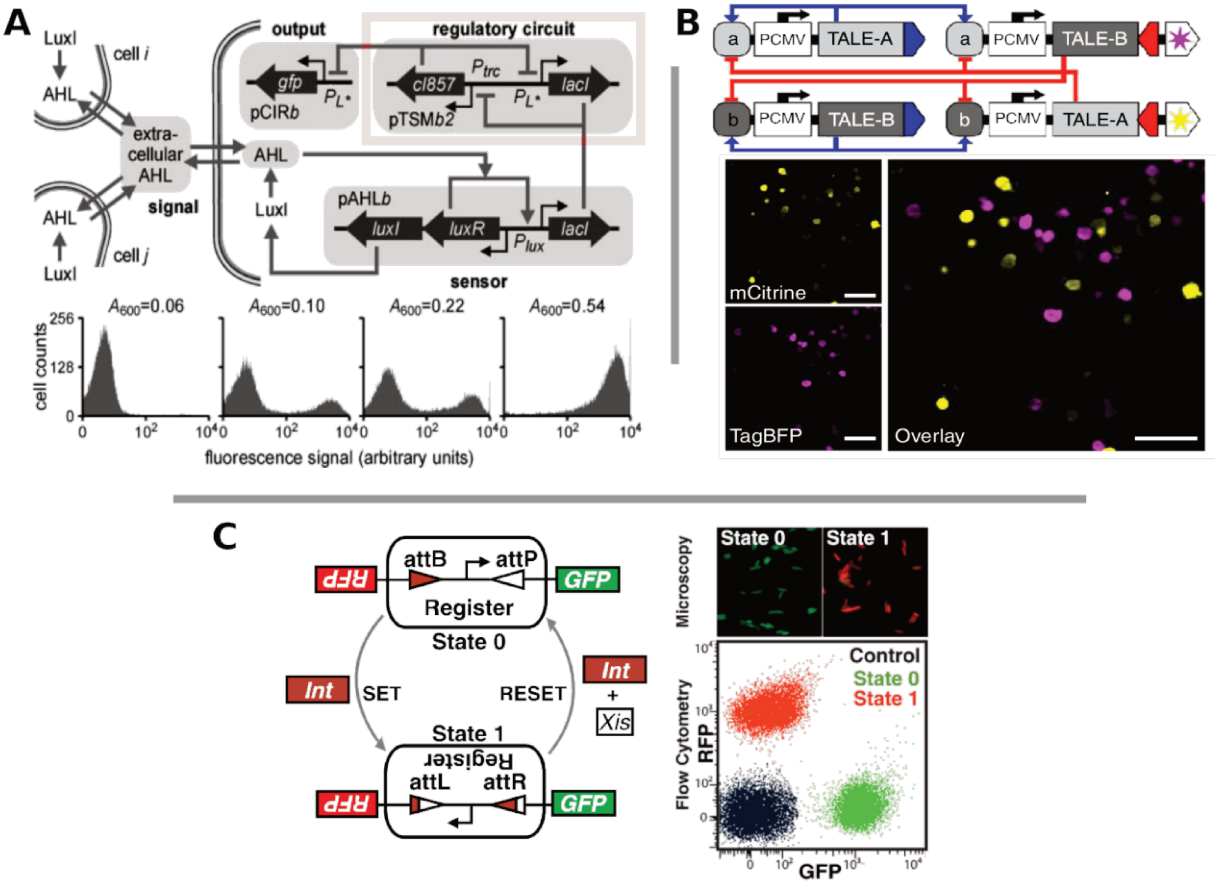

Figure 3-3 Examples synthetic toggle switches. A) Adapted from (Kobayashi et al. 2004). A genetic toggle switch (red rectangle) was integrated in a synthetic quorum-sensing-based circuit to binarize the expression of a reporter as a function of population level. A variation on this circuit is proposed in (Anesiadis et al. 2008) to control population levels. B) Adapted from (Lebar et al. 2014). The authors use competitive binding to introduce nonlinearities in a double-negative loop based on TALE technology in mammalian cells. They succeed in obtaining a bistable switch, thus circumventing the problem of non-cooperative binding of the recently developed TALE system, and propose this topology for other custom DNA-binding domains technologies. C) Adapted from (Bonnet et al. 2012). The authors propose a robust, rewritable bistable circuit based on an invertase + excisionase implementation to switch between RFP and GFP expression.

arbitrary regulatory interactions with the recent breakthrough of the CRISPR/Cas and TALE systems coupled with the possibility of introducing strong nonlinearities in the networks would open a new wide range of possibilities in synthetic biology.

Synthetic implementations of the traditional genetic toggle switch have also been used to study the mechanics of cell fate determination (Sekine et al. 2011; Wu et al. 2013). In spite of using a reduced, isolated prototype of a cellular regulatory circuit, the authors garner insights on the dynamic responses of cellular decisions and use it to predict the evolution of the two stable phenotypes in the cell population. They also demonstrate how the autonomous, stochastic repartition of cells between the two stable states of the system can be influenced by external stimuli, like manually added inducers or quorum sensing signals. Such circuits can also be used in synthetic biology to obtain two sub-populations of cells in a population of genetically identical individuals.

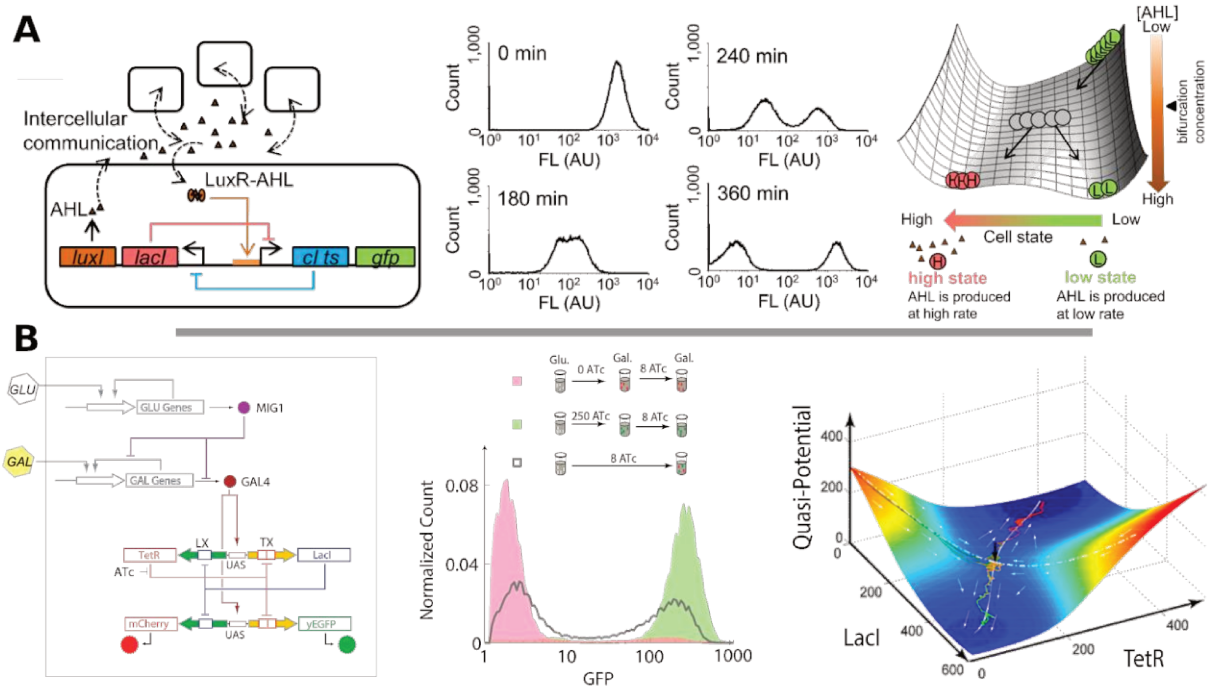

Figure 3-4 Synthetic toggle switches used to study cellular decision making. A) Adapted from (Sekine et al. 2011). A LacI-Clts toggle switch is coupled to a quorum sensing module. The system becomes bistable when the population level reaches a threshold value, and two phenotypes arise. B) Adapted from (Wu et al. 2013). The authors use a synthetic toggle switch to investigate cell fate determination in yeast cells. The toggle switch is repressed on both branches when the cells are grown in glucose, and the toggle switch circuit is activated under galactose. Cell fate can be influenced by the presence of aTc in the media. The authors use their experimental results to infer a model of their circuit, of which the quasi-potential landscape on the right is deduced. On this landscape the two basins of attraction LacI-high and TetR-high are visible. We use the fitted model from this study as a starting point for our model of the genetic toggle switch.

3.3 Theoretical studies of the genetic toggle switch

In the years leading up to the construction of the genetic toggle switch by Gardner and Collins in 2000, a number of theoretical studies led to a better understanding of the cooperative binding mechanisms that are necessary for nonlinearity and bistability to arise in the double-negative feedback loop (Arkin, Ross, and McAdams 1998; Cherry and Adler 2000; Thomas, Thieffry, and Kaufman 1995). The coupled ODEs with Hill function design (see Figure 3-5) was introduced and it was shown that higher Hill coefficients improved the stability of the

system. Despite the apparent importance of the magnitude of the Hill coefficients, it was later shown that cooperative binding was not the only possible source of nonlinearity in the toggle switch and that other mechanisms such as competition for operator sites can also generate it (Warren and Ten Wolde 2004).

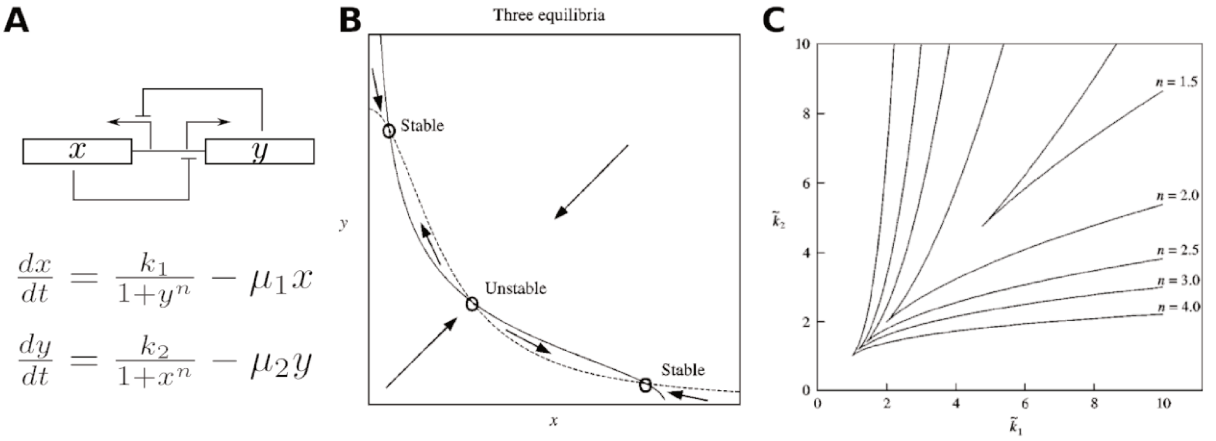

Figure 3-5 Theoretical study of the toggle switch. Adapted from (Cherry and Adler 2000). A) The standard ODE model of the toggle switch. The two species of the toggle switch, x and y repress each other's expression. This repressive effect is modeled by coupled ODEs: a Hill function describes the decrease in expression of one gene product because of the increase in the opposite gene product. B) The nullclines of the model are represented in the state space. The nullclines are the solutions to the functions $\frac{dx}{dt}(y) = 0$ and $\frac{dy}{dt}(x) = 0$. The arrows represent the direction of the derivatives of the x and y variables, the stable and unstable equilibrium points are identified at the intersection between the nullclines. C) Values of the parameters $\tilde{k}_1 = k_1/\mu_1$ and $\tilde{k}_2 = k_2/\mu_2$ for which the system is bistable for different values of the hill coefficient n. Values inside the cusps works for each of the particular values of n. Higher hill coefficients, and hence stronger nonlinearities, increase the domain of bistability of the network.

Because the toggle switch can play a central role in decision making, biosensing, and patterning, its stochastic behavior and response to noise was also extensively studied. The stochastic nature of gene expression and burst transcription can play a major role on stability and on levels of inducers necessary for state-transition (Bokes et al. 2013; Mackey and Tyrankami 2015; Tian and Burrage 2006; Walczak, Sasai, and Wolynes 2005) . Stochasticity of gene expression can also compensate for the linearity of non-cooperative binding and produce stable states on a monomeric double-repression loop (Andrecut et al. 2011; Lipshtat et al. 2006; Loinger et al. 2007). There is also growing evidence in the literature of the presence of a third or even a fourth "stable state" that can be introduced by the stochastic behavior of gene expression, small number effects, or transcription-translation delays in the toggle switch topology (Loinger et al. 2007; Ma et al. 2012; Strasser, Theis, and Marr 2012). A study of patterning in toggle-switch-controlled differentiation of cellular tissues also shows that stochasticity can alter the pattern from the deterministic prediction (Perez-Carrasco et al. 2016). An elegant analysis of stochastic toggle switch behavior can be found in (Kim and Wang 2007), where the authors use tools traditionally used in quantum mechanics to derive the potential energy landscape in the toggle switch, which can be of particular interest for the probabilistic examination of responses to external stimuli.

It is worth noting though, that to my knowledge, none of the models of the toggle switch reported so far in the literature have been fitted to single-cell data, and they were at best roughly compared to flow cytometry curves. The only exception being the paper by (Wu et al. 2013), where the authors construct a genetic toggle switch in yeast and use it to study

stochastic cell fate determination in naïve cells. They develop a deterministic coupled ODE model with nested Hill functions to account for the double repression in the inducer-TF-promoter interaction. They fit experimental results and use it to predict the stability of their designs. They also then run stochastic simulations of their model to study stochastic cell fate determination. This study is of particular interest for this work since it is the only one that has been fitted to real data. I use the model and parameters in this study as a starting point for my own model (See Chapter 2).

3.4 Theoretical control of the toggle switch

External control of genetic systems is still in its infancy, and although no experimental attempts have been made so far to control a genetic toggle switch externally, there are a few theoretical studies on the subject. I the first theoretical studies on controllability and control of the genetic toggle switch (Farcot and Gouzé 2007a, 2007b, 2008) the authors report a piecewise affine differential model of the toggle switch and make it follow a transition graph between the different regions of their model by inferring piecewise constant feedback control laws. Another study on the subject of toggle switch control can be found in (Wang et al. 2016). A control framework for nonlinear differential models is developed for controlling multistable networks, including an "enhanced" version of the genetic toggle switch. Their control framework is based on bifurcation analysis and focuses on control and controllability of the system from one attractor, or stable state, to another. A small discussion of the possibly beneficial effects of noise on controllability is also included, where the authors argue that an appropriate amount of noise can help destabilize attractors that would otherwise be too strong for the control strategy to successfully drive the system around.

We can also cite the series of papers by Sootla and colleagues (Sootla et al. 2013; Sootla, Oyarzún, et al. 2016; Sootla, Mauroy, and Goncalves 2016) where the authors focus on robust switching pulse strategies between the two stable states of a LacI-TetR toggle switch. Their proposed solutions are based on either reinforcement learning, which they conclude can be impractical due to the large amount of measurement data it necessitates, or on monotone systems theory, which relies on a model but is robust to parametric uncertainty. They go on to identify the so-called "switching separatrix" in the space of the two parameters of the input pulses (duration and intensity) and, interestingly, use their framework to investigate forced oscillations in a generalized eight species repressilator. They do not however investigate periodic switching between the states of the toggle switch or the out-of-equilibrium behavior of their eight species model.

Two focus on the specific question of controlling the genetic toggle switch towards its unstable equilibrium point or towards the unstable area. In the first one the authors use a piecewise affine model to describe the toggle switch and use the piecewise constant control framework previously described to drive the system towards the unstable steady state (Chaves and Gouzé 2011). They suggest approximate solutions to drive the system in the neighborhood of the state, or by passing through an unstable state cyclically. The second paper (Mohajerin Esfahani, Milias-argeitis, and Chatterjee 2013) uses the toggle switch system as an example to demonstrate the possibilities offered by the recently developed stochastic motion planning framework (Mohajerin Esfahani, Chatterjee, and Lygeros 2015) or

external cell control. The authors develop a stochastic differential equation model of the toggle switch by deriving the Langevin equation of the system, and then infer optimal input policies that maximize the probability of staying in the neighborhood of the deterministic unstable equilibrium point. They also investigate the possibility of controlling the toggle switch with only one input inducer and observe a dramatic decrease in performance.

In this thesis we will explore the possibility to control externally the bistable genetic toggle switch around its unstable equilibrium point. This thesis provides the first control results on bacteria at the single-cell level. New tools are created for this particular task, and others are improved, thus expanding the domains of applicability of in silico cybergenetics. The toggle switch is a fundamental topology encountered in both synthetic and natural gene regulation networks, and being able to control toggle switches and to maintain them in their unstable area opens the door to the study of intricate multi-stable networks, but it also opens a new range of possible usage for this topology in synthetic genetic circuits. In this thesis we demonstrate for example that it is possible to “reset” a population of toggle-switch-bearing bacteria by stabilizing them in the unstable area and then releasing the stabilization to let them separate in the two basins of attraction. It is also a difficult control problem, which would have required tedious trial-and-errors to implement in vivo, and solving it demonstrates the possibility offered by single-cell external control approaches. Finally, observing the dynamics of the genetic toggle switch in the vicinity of its unstable equilibrium point helped us expand the knowledge on its dynamics.

Alon, Uri. 2007. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Chapman & Hall/CRC.

Andrecut, Mircea, Julianne D. Halley, David A. Winkler, and Sui Huang. 2011. "A General Model for Binary Cell Fate Decision Gene Circuits with Degeneracy: Indeterminacy and Switch Behavior in the Absence of Cooperativity" edited by Nick Monk. *PLoS ONE* 6(5):e19358.

Anesiadis, Nikolaos, William R. Cluett, and Radhakrishnan Mahadevan. 2008. "Dynamic Metabolic Engineering for Increasing Bioprocess Productivity." *Metabolic Engineering* 10(5):255–66.

Arkin, A., J. Ross, and H. H. McAdams. 1998. "Stochastic Kinetic Analysis of Developmental Pathway Bifurcation in Phage Lambda-Infected Escherichia Coli Cells." *Annual Review of Genetics* 149(4):1633–48.

Åström, Karl Johan, and Tore Hägglund. 2006. *Advanced PID Control*. Research Triangle Park, NC: The Instrumentation, Systems and Automation Society.

Benner, Steven A., and A.Michael Sismour. 2005. "Synthetic Biology." *Nature Reviews Genetics* 6(7):533–43.

Bennett, Matthew R. et al. 2008. "Metabolic Gene Regulation in a Dynamically Changing Environment." *Nature* 454(August):1119–22.

Bennett, Matthew R., and Jeff Hasty. 2009. "Microfluidic Devices for Measuring Gene Network Dynamics in Single Cells." *Nature reviews Genetics* 10(9):628–38.

Bokes, Pavol, John R. King, Andrew T. A. Wood, and Matthew Loose. 2013. "Transcriptional Bursting Diversifies the Behaviour of a Toggle Switch: Hybrid Simulation of Stochastic Gene Expression." *Bulletin of Mathematical Biology* 75(2):351–71.

Bonnet, J., P. Subsoontorn, and D. Endy. 2012. "Rewritable Digital Data Storage in Live Cells via Engineered Control of Recombination Directionality." *Proceedings of the National Academy of Sciences* 109(23):8884–89.

Cantone, Irene et al. 2009. "A Yeast Synthetic Network for In Vivo Assessment of Reverse-Engineering and Modeling Approaches." *Cell* 137(1):172–81.

Chaves, Madalena, and Jean-Luc Gouzé. 2011. "Exact Control of Genetic Networks in a Qualitative Framework: The Bistable Switch Example." *Automatica* 47(6):1105–12.

Chen, Shuobing et al. 2012. "Automated Design of Genetic Toggle Switches with Predetermined Bistability." *ACS Synthetic Biology* 1(7):284–90.

Cherry, Joshua L., and Frederick R. Adler. 2000. "How to Make a Biological Switch." *Journal of Theoretical Biology* 203(2):117–33.

Collier, J. R., N. a Monk, P. K. Maini, and J. H. Lewis. 1996. "Pattern Formation by Lateral Inhibition with Feedback: A Mathematical Model of Delta-Notch Intercellular Signalling." *Journal of theoretical biology* 183(4):429–46.

Cuba Samaniego, Christian, Giulia Giordano, Jongmin Kim, Franco Blanchini, and Elisa Franco. 2016. “Molecular Titration Promotes Oscillations and Bistability in Minimal Network Models with Monomeric Regulators.” *ACS synthetic biology*.

Deans, Tara L., Charles R. Cantor, and James J. Collins. 2007. “A Tunable Genetic Switch Based on RNAi and Repressor Proteins for Regulating Gene Expression in Mammalian Cells.” *Cell* 130(2):363–72.

Dubnau, David, and Richard Losick. 2006. “Bistability in Bacteria.” *Molecular Microbiology* 61(3):564–72.

Ellis, Tom, Xiao Wang, and James J. Collins. 2009. “Diversity-Based, Model-Guided Construction of Synthetic Gene Networks with Predicted Functions.” *Nature biotechnology* 27(5):465–71.

Endy, Drew. 2005. “Foundations for Engineering Biology.” *Nature* 438(7067):449–53.

Farcot, Etienne, and Jean-luc Gouzé. 2007a. *A Mathematical Framework for the Control of Piecewise Affine Models of Gene Networks*. Sophia-Antipolis.

Farcot, Etienne, and Jean-luc Gouzé. 2007b. “Piecewise Constant Feedback Control of Piecewise Affine Gene Network Models.” Pp. 688–92 in *Hybrid Systems: Computation and Control, HSCC*, edited by A. Bemporad, A. Bicchi, and G. Buttazzo. Pisa, Italy: Springer.

Farcot, Etienne, and Jean-luc Gouzé. 2008. “A Mathematical Framework for the Control of Piecewise-Affine Models of Gene Networks.” *Automatica* 44(9):2326–32.

Ferrell, James E. 2002. “Self-Perpetuating States in Signal Transduction: Positive Feedback, Double-Negative Feedback and Bistability.” *Current Opinion in Cell Biology* 14(2):140–48.

Ferrell, James E. 2012. “Bistability, Bifurcations, and Waddington’s Epigenetic Landscape.” *Current Biology* 22(11):R458–66.

Fiore, Gianfranco, Giansimone Perrino, Mario di Bernardo, and Diego di Bernardo. 2015. “In Vivo Real-Time Control of Gene Expression: A Comparative Analysis of Feedback Control Strategies in Yeast.” *ACS synthetic biology* 5(2):154–62.

Fracassi, Chiara, Lorena Postiglione, Gianfranco Fiore, and Diego di Bernardo. 2016. “Automatic Control of Gene Expression in Mammalian Cells.” *ACS synthetic biology* 5(4):296–302.

Gardner, Timothy S., Charles R. Cantor, and James J. Collins. 2000. “Construction of a Genetic Toggle Switch in Escherichia Coli.” *Nature* 403(6767):339–42.

Ham, Timothy S., Sung K. Lee, Jay D. Keasling, and Adam P. Arkin. 2008. “Design and Construction of a Double Inversion Recombination Switch for Heritable Sequential Genetic Memory” edited by Rosemary Jeanne Redfield. *PLoS ONE* 3(7):e2815.

Ham, Timothy S., Sung Kuk Lee, Jay D. Keasling, and Adam P. Arkin. 2006. “A Tightly Regulated Inducible Expression System Utilizing Thefim Inversion Recombination

Switch." *Biotechnology and Bioengineering* 94(1):1–4.

Hersen, Pascal, Megan N. McClean, L. Mahadevan, and Sharad Ramanathan. 2008. "Signal Processing by the HOG MAP Kinase Pathway." *Proceedings of the National Academy of Sciences* 105(20):7165–70.

Johnston, Robert J., Sarah Chang, John F. Etchberger, Christopher O. Ortiz, and Oliver Hobert. 2005. "MicroRNAs Acting in a Double-Negative Feedback Loop to Control a Neuronal Cell Fate Decision." *Proceedings of the National Academy of Sciences of the United States of America* 102(35):12449–54.

Kim, Keun-Young, and Jin Wang. 2007. "Potential Energy Landscape and Robustness of a Gene Regulatory Network: Toggle Switch." *PLoS Computational Biology* 3(3):e60.

Kitano, H. 2002. "Systems Biology: A Brief Overview." *Science* 295(5560):1662–64.

Kobayashi, Hideki et al. 2004. "Programmable Cells: Interfacing Natural and Engineered Gene Networks." *Proceedings of the National Academy of Sciences of the United States of America* 101(22):8414–19.

Kotula, Jonathan W. et al. 2014. "Programmable Bacteria Detect and Record an Environmental Signal in the Mammalian Gut." *Proceedings of the National Academy of Sciences of the United States of America* 111(13):4838–43.

Kramer, Beat P. et al. 2004. "An Engineered Epigenetic Transgene Switch in Mammalian Cells." *Nature biotechnology* 22(7):867–70.

Lebar, Tina et al. 2014. "A Bistable Genetic Switch Based on Designable DNA-Binding Domains." *Nature communications* 5:5007.

Lewis, Kim. 2007. "Persister Cells, Dormancy and Infectious Disease." *Nature reviews Microbiology* 5(1):48–56.

Lipshtat, Azi, Adiel Loinger, Nathalie Q. Balaban, and Ofer Biham. 2006. "Genetic Toggle Switch without Cooperative Binding." *Physical Review Letters* 96(18):1–4.

Loinger, Adiel, Azi Lipshtat, Nathalie Q. Balaban, and Ofer Biham. 2007. "Stochastic Simulations of Genetic Switch Systems." *Physical Review E* 75(2):21904.

Lu, Mingyang et al. 2013. "Tristability in Cancer-Associated MicroRNA-TF Chimera Toggle Switch." *The Journal of Physical Chemistry B* 117(42):13164–74.

Lutz, R., and H. Bujard. 1997. "Independent and Tight Regulation of Transcriptional Units in Escherichia Coli via the LacR/O, the TetR/O and AraC/I1-I2 Regulatory Elements." *Nucleic acids research* 25(6):1203–10.

Ma, Rui, Jichao Wang, Zhonghuai Hou, and Haiyan Liu. 2012. "Small-Number Effects: A Third Stable State in a Genetic Bistable Toggle Switch." *Physical review letters* 109(24):248107.

Mackey, Michael C., and Marta Tyran-kami. 2015. "The Limiting Dynamics of a Bistable Molecular Switch With and Without Noise." *Journal of mathematical biology* 1–20.

Melendez, Justin et al. 2014. “Real-Time Optogenetic Control of Intracellular Protein Concentration in Microbial Cell Cultures.” *Integrative biology* 6(3):366–72.

Menolascina, Filippo et al. 2014. “In-Vivo Real-Time Control of Protein Expression from Endogenous and Synthetic Gene Networks.” *PLoS computational biology* 10(5):e1003625.

Mettetal, Jerome T., Dale Muzzey, C. Gomez-Uribe, and Alexander van Oudenaarden. 2008. “The Frequency Dependence of Osmo-Adaptation in Saccharomyces Cerevisiae.” *Science* 319(5862):482–84.

Mikeladze-Dvali, Tamara et al. 2005. “The Growth Regulators Warts/lats and Melted Interact in a Bistable Loop to Specify Opposite Fates in Drosophila R8 Photoreceptors.” *Cell* 122(5):775–87.

Milias-Argeitis, Andreas et al. 2011. “In Silico Feedback for in Vivo Regulation of a Gene Expression Circuit.” *Nature biotechnology* 29(12):1114–16.

Milias-Argeitis, Andreas, Marc Rullan, Stephanie K. Aoki, Peter Buchmann, and Mustafa Khammash. 2016. “Automated Optogenetic Feedback Control for Precise and Robust Regulation of Gene Expression and Cell Growth.” *Nature communications* 7(May):12546.

Mohajerin Esfahani, P., D. Chatterjee, and J. Lygeros. 2015. “Motion Planning for Continuous Time Stochastic Processes: A Dynamic Programming Approach.” *IEEE Transactions on Automatic Control* PP(99):1.

Mohajerin Esfahani, Peyman, Andreas Milias-argeitis, and Debasish Chatterjee. 2013. “Analysis of Controlled Biological Switches via Stochastic Motion Planning.” Pp. 9–14 in *European Control Conference*.

Moon, Tae Seok et al. 2011. “Construction of a Genetic Multiplexer to Toggle between Chemosensory Pathways in Escherichia Coli.” *Journal of Molecular Biology* 406(2):215–27.

Norman, Thomas M., Nathan D. Lord, Johan Paulsson, and Richard Losick. 2013. “Memory and Modularity in Cell-Fate Decision Making.” *Nature* 503(7477):481–86.

Oppenheim, Amos B., Oren Kobiler, Joel Stavans, Donald L. Court, and Sankar Adhya. 2005. “Switches in Bacteriophage Lambda Development.” *Annual review of genetics* 39:409–29.

Perez-Carrasco, Ruben, Pilar Guerrero, James Briscoe, and Karen Page. 2016. “Intrinsic Noise Profoundly Alters the Dynamics and Steady State of Morphogen-Controlled Bistable Genetic Switches.” *ArXiv* 1–22.

Polizzi, Karen M., and Cleo Kontoravdi. 2015. “Genetically-Encoded Biosensors for Monitoring Cellular Stress in Bioprocessing.” *Current Opinion in Biotechnology* 31:50–56.

Rep, Martijn, Marcus Krantz, Johan M. Thevelein, and Stefan Hohmann. 2000. “The

Transcriptional Response of Saccharomyces Cerevisiae to Osmotic Shock." Pp. 8290–8300 in *Journal of Biological Chemistry*, vol. 275.

Ruess, Jakob, Francesca Parise, Andreas Milias-Argeitis, Mustafa Khammash, and John Lygeros. 2015. "Iterative Experiment Design Guides the Characterization of a Light-Inducible Gene Expression Circuit." *Proceedings of the National Academy of Sciences* 112(26):8148–53.

Sambrook, Joseph, and David W. Russell. 2003. *Molecular Cloning : A Laboratory Manual*. 2nd ed. Cold Spring Harbor Laboratory.

Sekine, R. et al. 2011. "Tunable Synthetic Phenotypic Diversification on Waddington's Landscape through Autonomous Signaling." *Proceedings of the National Academy of Sciences* 108(44):17969–73.

Shaner, Nathan C., Paul A. Steinbach, and Roger Y. Tsien. 2005. "A Guide to Choosing Fluorescent Proteins." *Nature methods* 2(12):905–9.

Siegal-Gaskins, Dan, Elisa Franco, Tiffany Zhou, and Richard M. Murray. 2015. "An Analytical Approach to Bistable Biological Circuit Discrimination Using Real Algebraic Geometry." *Journal of The Royal Society Interface* 12(108):20150288.

Sootla, Aivar, Alexandre Mauroy, and Jorge Goncalves. 2016. "Shaping Pulses to Control Bistable Monotone Systems Using Koopman Operator." *ArXiv* (1).

Sootla, Aivar, Diego Oyarzún, David Angeli, and Guy Bart Stan. 2016. "Shaping Pulses to Control Bistable Systems: Analysis, Computation and Counterexamples." *Automatica* 63:254–64.

Sootla, Aivar, Natalja Strelkowa, Damien Ernst, Mauricio Barahona, and Guy-bart Stan. 2013. "Toggling a Genetic Switch Using Reinforcement Learning." *ArXiv* 1–7.

Strasser, Michael, Fabian J. Theis, and Carsten Marr. 2012. "Stability and Multiattractor Dynamics of a Toggle Switch Based on a Two-Stage Model of Stochastic Gene Expression." *Biophysical Journal* 102(1):19–29.

Thomas, René, Denis Thieffry, and Marcelle Kaufman. 1995. "Dynamical Behaviour of Biological Regulatory networks—I. Biological Role of Feedback Loops and Practical Use of the Concept of the Loop-Characteristic State." *Bulletin of Mathematical Biology* 57(2):247–76.

Tian, Tianhai, and Kevin Burrage. 2006. "Stochastic Models for Regulatory Networks of the Genetic Toggle Switch." *Proceedings of the National Academy of Sciences of the United States of America* 103(22):8372–77.

Toettcher, Jared E., Delquin Gong, Wendell a Lim, and Orion D. Weiner. 2011a. *Light Control of Plasma Membrane Recruitment Using the Phy–PIF System*.

Toettcher, Jared E., Delquin Gong, Wendell a Lim, and Orion D. Weiner. 2011b. "Light-Based Feedback for Controlling Intracellular Signaling Dynamics." *Nature methods* 8(10):837–39.

Tyszkiewicz, Amy B., and Tom W. Muir. 2008. “Activation of Protein Splicing with Light in Yeast.” *Nature methods* 5(4):303–5.

Uhlendorf, Jannis et al. 2012. “Long-Term Model Predictive Control of Gene Expression at the Population and Single-Cell Levels.” *Proceedings of the National Academy of Sciences of the United States of America* 109(35):14271–76.

Del Vecchio, Domitilla, Aaron J. Dy, and Yili Qian. 2016. “Control Theory Meets Synthetic Biology.” *Journal of the Royal Society, Interface / the Royal Society* 13(120).

Di Ventura, Barbara, Caroline Lemerle, Konstantinos Michalodimitrakis, and Luis Serrano. 2006. “From in Vivo to in Silico Biology and Back.” *Nature* 443(7111):527–33.

Walczak, Aleksandra M., Masaki Sasai, and Peter G. Wolynes. 2005. “Self-Consistent Proteomic Field Theory of Stochastic Gene Switches.” *Biophysical journal* 88(2):828–50.

Wang, Le-Zhi et al. 2016. “A Geometrical Approach to Control and Controllability of Nonlinear Dynamical Networks.” *Nature Communications* 7(7):11323.

Warren, Patrick B., and Pieter Rein Ten Wolde. 2004. “Enhancement of the Stability of Genetic Switches by Overlapping Upstream Regulatory Domains.” *Physical Review Letters* 92(12):128101–1.

Wiener, Norbert. 1948. *Cybernetics: Or Control and Communication in the Animal and the Machine*. MIT Press.

Wu, Min et al. 2013. “Engineering of Regulated Stochastic Cell Fate Determination.” *Proceedings of the National Academy of Sciences* 110(26):10610–10615.

Yang, Lei et al. 2014. “Permanent Genetic Memory with > 1-Byte Capacity.” *Nature Methods* 11(12):1261–66.

Yao, Guang, Tae Jun Lee, Seiichi Mori, Joseph R. Nevins, and Lingchong You. 2008. “A Bistable Rb–E2F Switch Underlies the Restriction Point.” *Nature Cell Biology* 10(4):476–82.

Chapter II

Observing and actuating on trapped bacterial cells

Table of Contents

1 Introduction 43

2 Synthetic biology 45

2.1 Chassis development 46

2.2 LacI-TetR toggle switch 49

2.3 Modular Cloning (MoClo) and toxicity issue 49

2.4 Operon branches design 54

2.5 Fusion protein branches design 55

2.6 The toggle switch library 57

3 Microfluidics & automation 59

3.1 Microfluidics 60

3.1.1 Molded PDMS devices 60

3.1.2 Microfluidic wafers fabrication methods 60

3.1.3 Main device – Mother Machine 63

3.1.4 Early device designs: Population chambers 64

3.1.5 Surface passivation, clogging, and flushing 65

3.1.6 Input delivery 66

3.1.7 Reference crosses 68
3.2 Image analysis 69
3.2.1 Robust algorithm for online measurement of fluorescence levels 70
3.2.2 Image gradient and ellipsis fitting on mother machine images 71
3.2.3 Mathematical morphology attempts on single layers of cells 72
3.3 Experiments automation 74
4 Modeling 77
4.1 The reaction network 77
4.1.1 LacI-DNA and LacI-IPTG interactions 77
4.1.2 TetR-DNA and TetR-aTC interactions 78
4.1.3 Transcription and translation 78
4.1.4 Dilution and degradation 79
4.2 ODE model 79
4.3 Gillespie's stochastic simulation algorithm 80
4.4 Parameter identification 82
4.5 Controllers 83
4.5.1 Bang-bang control 83
4.5.2 Proportional-integral control 83
5 Conclusion 84

1 Introduction

The development of an external single-cell control platform involves a number of different methods and scientific fields: genetic engineering, signal processing, microfabrication, microscopy, electrical engineering, control theory, and programming were all required to achieve real-time control of gene expression in bacteria. Specifically, the possibility of controlling gene expression through an in silico feedback loop at the single cell level was made possible by recent developments in time-lapse microscopy, cell segmentation, and microfluidics, but a number of open problems still remain to be solved by the experimenter for each different application. The necessity of automated and on-line imaging, data analysis and actuation required by real-time control increased the difficulty of creating such a platform. In this chapter, I discuss the obstacles I had to cope with and the implementation choices I made.

The first section focuses on the biological aspects of my work. I start by describing how I modified the host cell strains to prevent endogenous cellular processes from interfering with the synthetic toggle switch. Then, I review the different iterations for the design of the toggle switch itself. The design and construction methods were chosen to maximize the chances that at least one designed circuit would be bistable. I finish the section by describing the characteristics of the selected circuit.

In the second section, I discuss the hardware and software choices that were made to develop the platform. In order to control a multistable system and be able to accurately measure its state, I needed to be able to track single cells for extended periods of time and to extract their level of fluorescence periodically, which necessitated a microfluidic-based platform rather than a batch-culture type of platform. The platform described in this section is comprised of: a custom-made microfluidic chip for long-term tracking of bacteria as well as automated chemical actuation; an entirely automated epifluorescence microscope; various pieces of software to link the different interfaces together, extract data, and run the control algorithms; and finally a custom-made electronic valve actuator for precise, real-time modification of the chemical environment of the cells.

Finally, the last section of this chapter details the mathematical model derived from a simplified chemical description of our toggle switch, and the simulation algorithms that were used to study the feasibility of real-time control of the toggle switch. The control algorithms used to control toggle switches in real-time (namely the PI and bang-bang strategies) are also discussed. The last part of the section describes the procedure for parameters identification that was applied to characterization data.

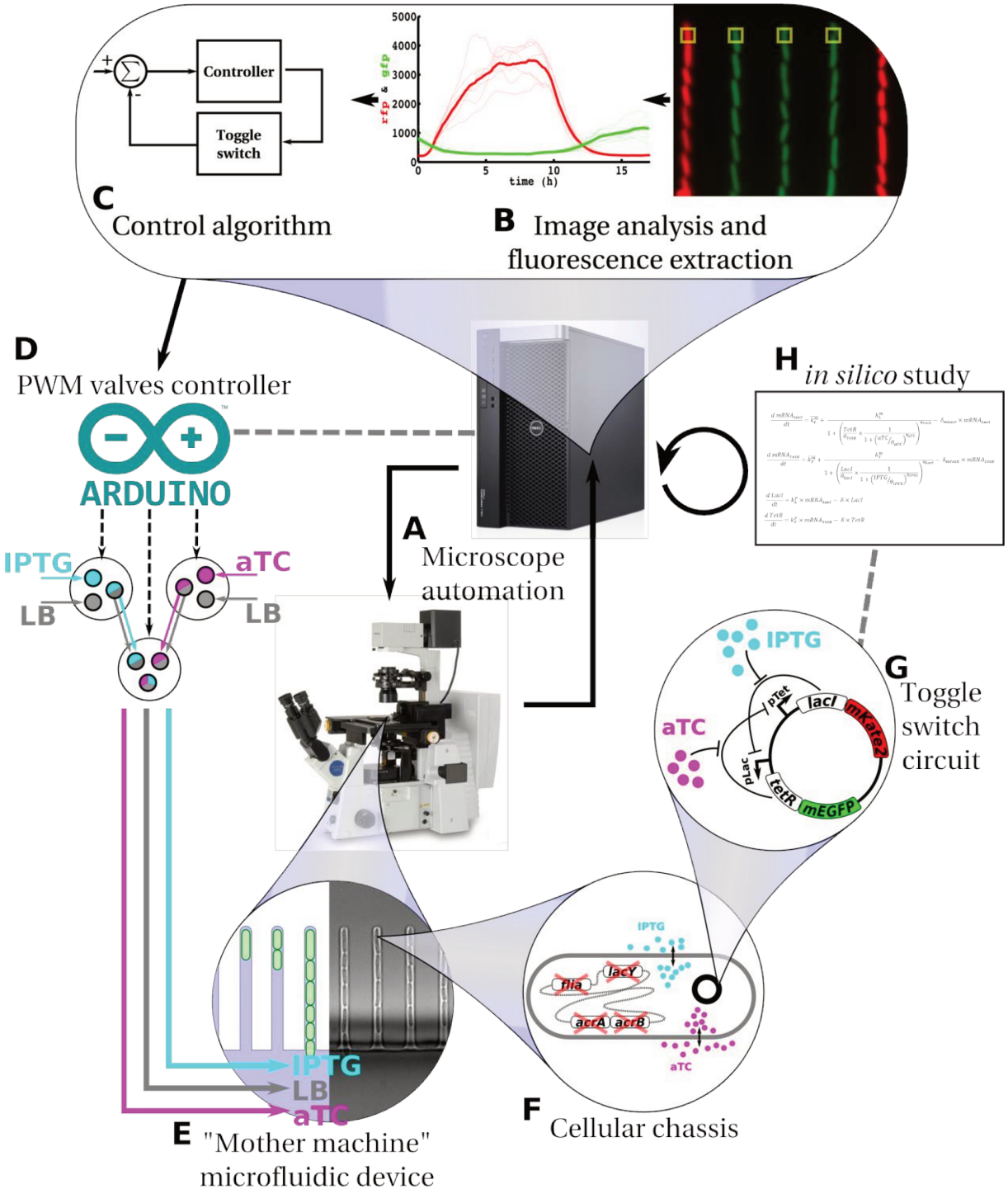

Figure 1-1 General architecture of the control platform. A) Microscope automation (section 0) makes it possible to image cells in both red and green channels over extended periods of time with a precise focus. B) Image analysis (section 3.2) are run on-the-fly to observe cell state in real time. C) The fluorescence state of the cells is used by the control algorithms (section 4.5) to decide on the concentrations of inducers to apply to the cells to maintain them around the control objective. D) The decision is sent to a custom-made arduino-based electronic circuit connected to the computer to mix the inducers to the right concentrations with solenoid valves and pulse-width modulation (section 3.1.6). E) The mix of inducers and nutrients is flown through a “mother machine” microfluidic device in which the cells are grown in isolated chambers (section 3.1.3). The microfluidic device is placed on the microscope for automated time-lapse observation and control of the cells. F) To circumvent serious interferences between the host cell and the controlled circuit, genes were removed from the genome of our Escherichia coli strain to obtain a usable “chassis” strain (section 2.1). G) A library of toggle switches was developed for control and transformed into the chassis strain (sections 2.2 to 2.6). H) A mathematical model of the toggle switch was developed and fitted to the data to study toggle switch dynamics and controllability (section 4).

2 Synthetic biology

At the heart of the platform, the cells and their synthetic genetic circuit went through many iterations throughout this PhD. In this section I describe the main design choices as well as the protocols that were developed to obtain both host cell strains adapted to our control needs, and a bistable toggle switch circuit to control. All circuits and strains are listed in appendix A.

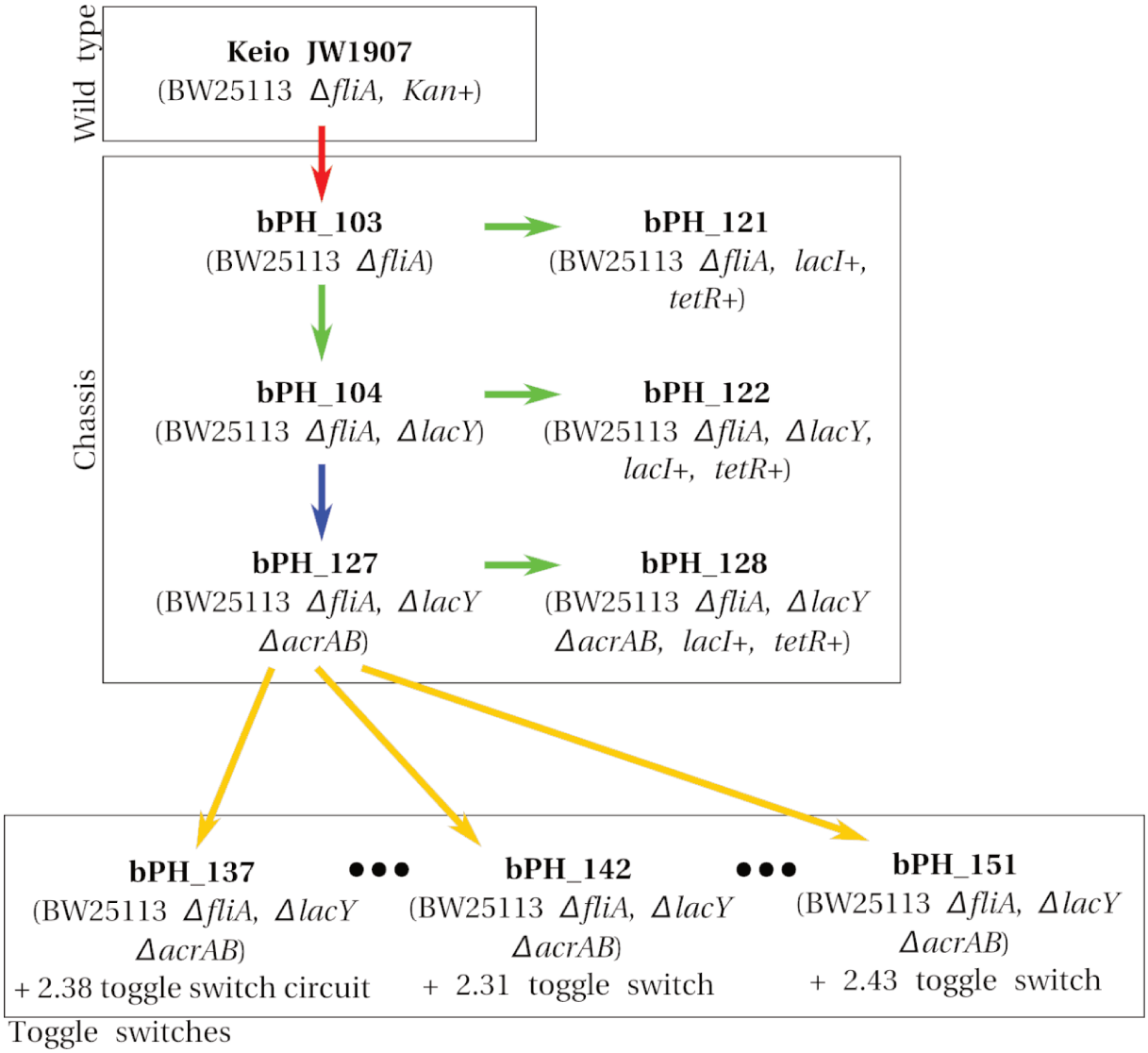

Figure 2-1 Hierarchy of the strains presented in this study. The initial strain was taken from the Keio collection, and after several iterations the bPH_127 chassis strain was used as the host for toggle switch control experiments. The red arrow represents flipase excision to remove the Kan cassette for kanamycin resistance. Green arrows represent P1 phage transduction. Blue arrows represent Datsenko-Wanner knockouts. Yellow arrows represent plasmid transformations. The bPH_121, bPH_122, and bPH_128 strains were transformed with plasmids to run exploratory experiments on the developed chassis for control. See appendix A for an exhaustive list of strains.

2.1 Chassis development

The cellular chassis[1] is an integral part of the platform, as the genome of the organism needs to be modified in such a way that the toggle switch can be implemented, observed, and controlled. In the case of this toggle switch based on the action of the tet and lac repression systems, some cellular mechanisms, described in the following paragraphs, can interfere with the action of the inducible promoters and impede the controllability of the system, and had to be removed. Furthermore, E. coli possess flagella and can usually swim around in their growth media, which presents a problem for long-term observation and tracking in the microfluidic device. In Figure 2-1, a hierarchical representation of the main strains presented in this study is given. After several rounds of experimental characterization, the final chassis strain, bPH_127, is selected as the best host for synthetic circuits to control.

As a starting point for my chassis, I used the JW1907 strain from the Keio collection (Baba et al. 2006). The Keio mutants collection features all possible E. coli K-12 BW25113 strains with all non-lethal single-gene knockouts in the genome. In JW1907 the fliA gene is knocked out. fliA encodes for the specialized flagellar sigma factor σ28 (McCarter 2006; Wilkinson et al. 2011; Ikebe et al. 1999; Dailey & Berg 1993), and is traditionally removed to prevent the cells from swimming around in microfluidic chips. Keio strains feature a resistance cassette, which I removed to obtain the first chassis strain bPH_103.

From this strain, I developed the strain bPH_104 by deleting the lacY gene. Gene lacY expresses a lactose permease that is responsible for bi-stability in the lac operon (Santillán et al. 2007; Ozbudak et al. 2004). The expression of LacY is activated when lactose (or analogues such as Isopropyl β-D-1-thiogalactopyranoside[2]) is present in the growth media, and this increase in expression of the permease causes more lactose to be transported into the cytoplasm, further increasing permease expression (see strain BW25113 in Figure 2-2). This type of self-activating network makes the expression of genes downstream of the promoter bistable: LacY would interfere with our constructions based on the lac promoter and with lactose/IPTG induction. I removed the lacY gene using standard P1 phage transductions (Miller 1992). Interestingly, once the LacY permease is removed, lactose or its equivalent IPTG can still diffuse through the membrane, although at much lower rates (Marbach & Bettenbrock 2012). Thus we could drive the internal concentration of inducer but without the bistable effects of the lac operon.

Finally, I implemented a variant of this strain by integrating the regulator genes tetR and lacI in tandem to the chromosomes of bPH_104 to produce two more strains, bPH_121 and bPH_122, respectively. The constitutive expression of TetR and LacI allowed me to control the expression of any gene placed under a lac or tet promoter. The cassette containing lacI and tetR was transferred from the E.coli strain DH5αZ1(Lutz & Bujard 1997), again by standard P1 phage transduction (Miller 1992). I did not remove the antibiotic resistance

[1] The engineered Escherichia coli strains in which the genetic circuits to control will be transformed are referred to as "chassis" strains. We refer to the original BW25113 strain as "wild-type".

[2] Isopropyl β-D-1-thiogalactopyranoside, or IPTG, is a molecular mimic of allolactose that cannot be metabolized by the cell. Allolactose is an isomer of lactose and is the actual inhibitor of the LacI protein. Lactose is converted to allolactose by the β-galactosidase enzyme of the lac operon.

cassette used for selection, therefore the bPH_121 and bPH_122 strains are spectinomycin resistant. These two strains are the chassis strains I used to conduct my first series of control experiments and gather knowledge on the different systems I was working with.

Early in my exploratory experiments I observed adaptation to anhydrotetracycline (aTC) induction of the tet system, i.e. the response of the tet system to successive inductions would weaken over time. aTC is part of the tetracylines family of antibiotics and it is the inducer of choice for the tet induction system in bacteria because it is less toxic to cells than tetracycline itself or its readily-available variants (Rasmussen et al. 1991). The tet induction system is derived from the Tn10 operon system, which is activated in presence of tetracycline and expresses an efflux pump for evacuating tetracyclines out of the cytoplasm. The original wild-type BW25113 strain does not have the tetA gene responsible for the expression of the tetracyclines-specialized pump. However, it is suspected that Escherichia coli uses another efflux pump for adapting to high concentrations of tetracycline: the acriflavine pump encoded by the acrA and acrB genes (Ma et al. 1995; Zgurskaya & Nikaido 1999; Le et al. 2006), which are present in the genome of BW25113. No clear connection has been identified between the tetracyclines and the two genes, but it would not be surprising that the expression of AcrA and AcrB is directly or indirectly activated by tetracyclines levels, thus leading to slowly decreasing activation of the tet promoter's expression by aTC. See strain BW25113 in Figure 2-2 for a visual representation of the mechanism.

In order to make control of gene expression with the tet system possible for extended periods of time, I designed another strain in which the acrA and acrB genes were deleted. To do this, I removed the two genes, which are next to each other, in a one-step deletion from the "wild-type" BW25113 genome. Then I transferred the double deletion through P1 phage transduction into strain bPH_104 to obtain strain bPH_127, as is the classical procedure to avoid off-target mutations. Double acrAB deletion was performed by a modified Wanner chromosomal deletion/integration protocol (Datsenko & Wanner 2000). This protocol uses lambda-red homologous recombination to replace the target genes in the chromosome with a selection cassette. Because the technique requires short homology regions flanking the acrAB genes, I could easily obtain them via oligo annealing. I transferred the regions into standard modular cloning (MoClo) plasmids[3] to rapidly construct the replacement cassettes and to offer the possibility of fast chromosomal integration of MoClo circuits. Once the acrA and acrB genes were removed, we observed a higher sensitivity to light, especially during fluorescence imaging, and antibiotics, as well as a lower growth rate in presence of antibiotics. It is believed that the acriflavine pump that those two genes encode for is actually a multi-drug efflux pump transporting a wide range of toxic elements out of the cell (Zgurskaya & Nikaido 1999). However, the absence of this efflux pump may lead to the accumulation of other toxic chemicals, leading to the hypersusceptible phenotype observed in the new bPH_127 strain.

[3] Modular cloning, or MoClo, is a standardized cloning method based on the Golden gate protocol. This methodology was used extensively during my PhD because it offers a number of advantages for synthetic circuits construction. The method and its usage are described in section 2.2.
The MoClo plasmids bearing the homology regions are described in appendix A as level 1 plasmids 1.25 to 1.27.

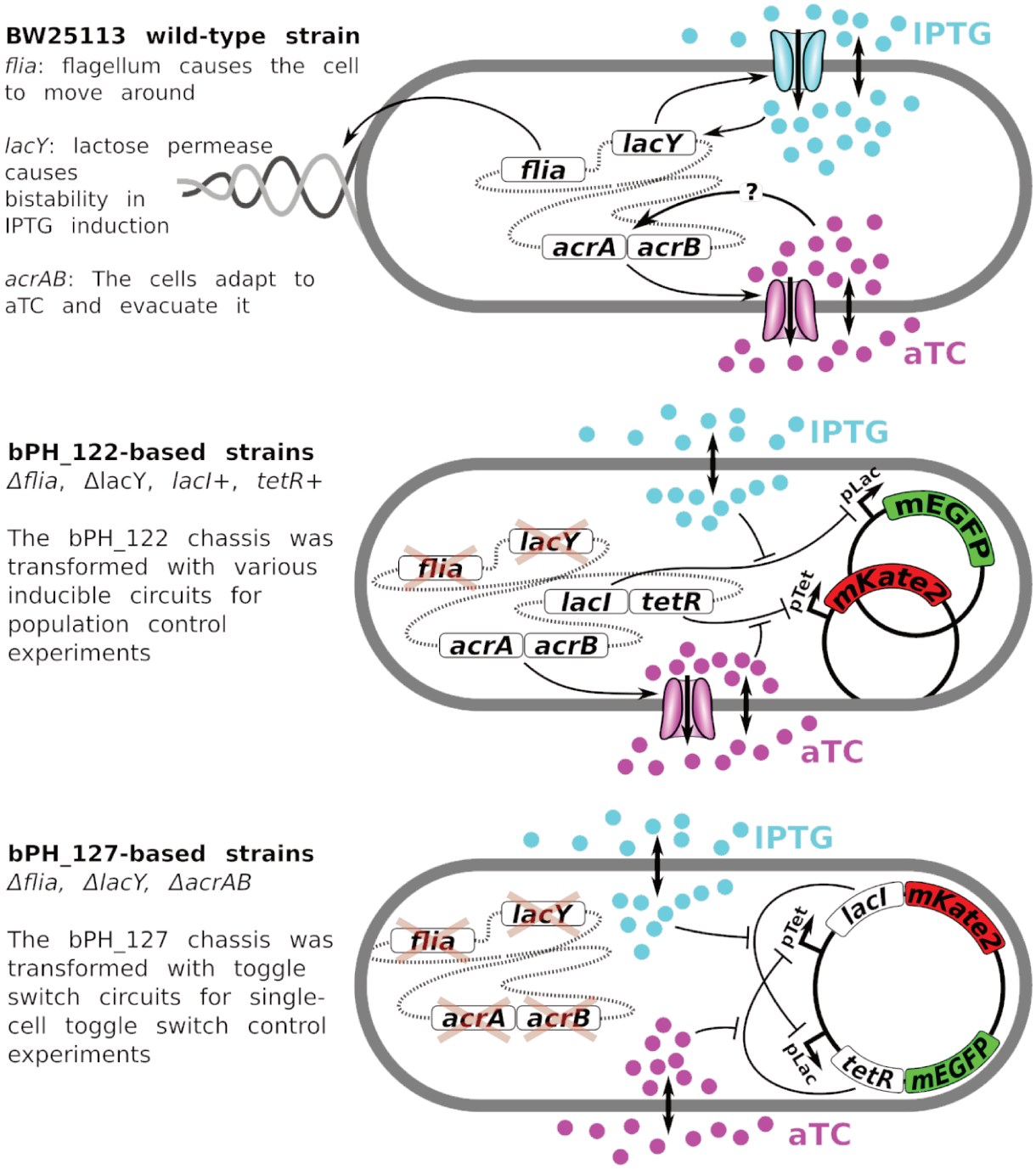

Figure 2-2 - Chassis strains used in this thesis with the type of circuits they were transformed with. The wild type strain from the Keio library (BW25113) features several elements that hinder the quality of control or make long-term observation difficult: the flagellum makes the cell move around the field of view, and the lactose permease (lacY) and the acriflavine pump (acrAB) interfere with cytoplasmic levels of inducers. In the bPH_122 chassis strain I removed the flagellum (fliA) and inserted a cassette into the chromosome to express lacI and tetR constitutively. Characterization experiments with simple transformed plasmids showed that the response to IPTG induction was bistable, so I also removed the lacY gene responsible for the permease, and bistable induction by IPTG was no longer observed. I ran control experiments on simple circuits transformed into this chassis strain and observed long-term adaptation of the cells to anhydrotetracycline. In the last chassis strain, bPH_127, I removed the acrA and acrB genes responsible for the production of the efflux pump that would cause adaptation to aTC. LacI and TetR were no longer expressed constitutively, and I transformed the strain with several different toggle switch plasmids to work on the final goal of this thesis: real-time control of a genetic toggle switch.

Because this strain was going to be transformed with toggle switch plasmids in order to control them, I constructed it from the bPH_104 strain instead of bPH_122. This way the strain did not express LacI and TetR constitutively, which would interfere with the functioning of the toggle switch.

To summarize, the final chassis strain bPH_127 is a quadruple mutant of Escherichia coli K12 BW25113 strain with genes fliA, lacY, acrA and acrB knocked out. The cells do not self-propel, their response to IPTG or other lactose derivatives is not bistable and they do not flush out tetracyclines.

Now that it has been developed, the chassis strains I developed and present here can be used for controlling other systems as they are adapted to the control framework. A list of the strains I developed during my PhD can be found in appendix A.

2.2 LacI-TetR toggle switch

I chose to develop a toggle switch based on the lac and tet systems, both because these systems have been extensively used and studied in synthetic biology, and because inducer chemicals can be delivered into the growth media to the expression downstream of each system. To be able to observe the state of the toggle switch in real time, I also ensured that two different fluorescent reporter proteins would be co-expressed with the toggle switch. I chose to combine LacI with the mKate2 fluorescent protein and TetR with the mEGFP protein. The two opposite parts of the toggle switch consisting of, on one hand, the tet promoter (pTet), the lacI gene and the mKate2 fluorescent protein, and, on the other hand, the lac promoter (pLac), the tetR gene and the mEGFP fluorescent protein, are referred to as "branches" here. Also, in order to maximize the chances of having at least one bistable toggle switch, I constructed a library of circuits with different expression strengths for the two branches. I eventually picked the circuit in the library that seemed the most promising, circuit 2.31 (see appendix A).

The final toggle switch design is illustrated in Figure 2-3. This 2.31 plasmid was eventually transformed into bPH_142 strain and used to perform the control experiments described in the next chapter. In the rest of the section I will present the design choices that were made to construct this circuit, the protocols I used and developed to construct the library and some of the problems I ran into.

2.3 Modular Cloning (MoClo) and toxicity issue

To build a bistable toggle switch it is important that both TetR and LacI are expressed in proportions such that their mutual repressive strengths are comparable. Also, tagging both genes with a different fluorescent protein allows for the monitoring of the state of the toggle switch by microscopy. Thus, developed a library of toggle switch circuits to ensure I would obtain at least one bistable toggle switch with appropriate properties (dynamics, bistability, symmetry). I took advantage of recent technical progresses in the field of molecular cloning and developed a few techniques of my own to accelerate the process.

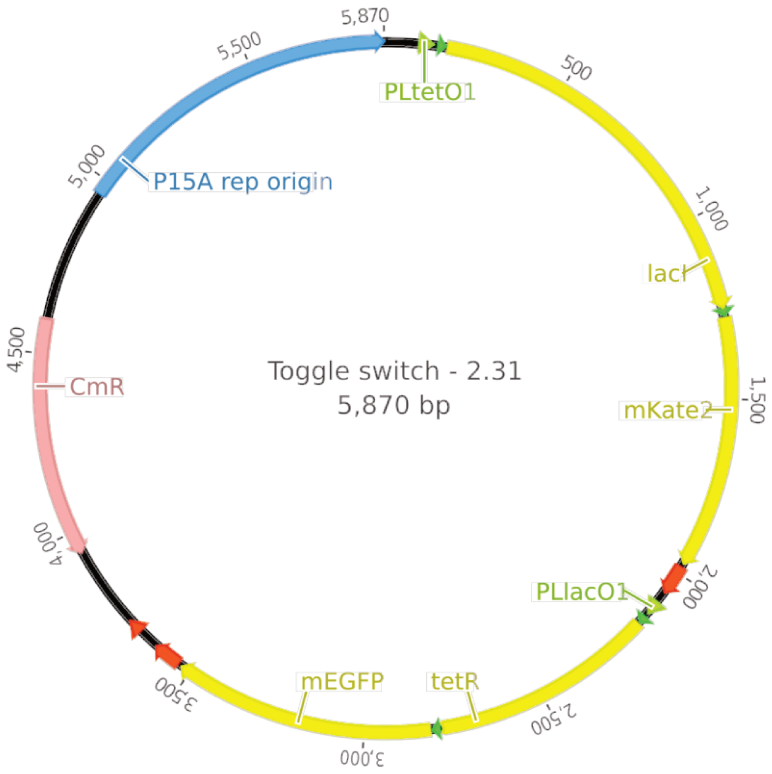

Figure 2-3 Plasmid map of the 2.31 toggle switch plasmid. The yellow arrows represent the coding sequences of transcription factors and fluorescent proteins. The light green arrows represent the pLac and pTet promoters to which the transcription factors will bind to repress each other's expression. The sharper green arrows represent ribosome binding sites. The red arrows represent terminators. The pink arrow represents the chloramphenicol resistance cassette, and the blue arrow represents the origin of replication of the plasmid.

Instead of traditional cloning methods, I used a recently-developed cloning protocol called the Modular Cloning method(Weber et al. 2011). MoClo is based on the Golden Gate cloning protocol (Engler et al. 2008; Engler et al. 2009) which uses type IIS enzymes. Those enzymes have the particularity of cutting outside of their recognition site. These recognition sites can be inserted into the DNA sequences of interest to minimize the cloning scar and possibly avoid it completely. The Golden Gate cloning procedure consists of cycles of restriction and ligation that make the cloning more robust and efficient, thus requiring lower quantities of DNA at each step. Building on the reliability of the Golden Gate cloning method and on previous DNA parts standardization initiatives (Rebatchouk et al. 1996; Knight 2003; Anderson et al. 2010), the MoClo procedure is a standardized version of the Golden Gate DNA assembly method. In synthetic biology circuits, construction mostly consists of assembling the same type of elements in the same order. In MoClo, the genetic circuits consist of assemblies of transcriptional units, which themselves consist of an assembly of the following elements: a promoter (P) – a 5' untranslated region (U) – an optional signaling peptide (S) – a coding sequence (C) – and finally a terminator (T). The authors designed a series of plasmids with ligation overhangs such that elements of each category would always assemble in the same order (P>U>S>C>T). One of the great advantages inherited from Golden Gate is that it all the parts can be assembled together during a single one-pot assembly, meaning that an entire transcriptional unit consisting of up to 5 elements can be assembled overnight with MoClo, when traditional cloning methods would require at least three steps.

It also allows an easy combinatorial assembly: because parts are standardized, several assembly pots can be easily set up with a single variation between them (e.g a different 5'UTR, various promoters…), and each would produce a different transcription unit. Golden Gate / MoClo even offers the possibility of assembling families of construct variants in one pot (Engler et al. 2009), although the task of screening for the better constructs then becomes the bottleneck (Engler et al. 2009; Engler & Marillonnet 2011). See Figure 2-4 for a description of the MoClo protocol as well as illustrations and appendix A for a list of the MoClo vectors used and developed in this thesis.

To summarize, the Modular Cloning technique offers a number of valuable advantages for circuit library assembly, and I used this method to assemble my library of toggle switches. To maximize the chances of obtaining a bistable switch, a simple solution was to design and implement circuits with a wide range of protein translation rates. These different circuits would only differ in their ribosome binding sites, so the only part that would change in my assemblies would be the 5'UTR parts of each branch ("U" part of MoClo assemblies). Once I transferred the ribosome binding site (RBS) libraries to the MoClo Level 0 plasmids (pL0-U), the possibility to rapidly produce all the variants of each branch used in the circuits greatly sped up the process. It would also permit rapid reconstruction of the branches if necessary, since I could re-use the same parts. With the knowledge gathered on the branches during this work on the toggle switch to model circuit behavior, I can also construct new circuits with those same branches without much effort or time.

I had to cope with several complications during the assembly of the toggle switches. At the transcriptional unit level, the promoter or RBS region of the level 1 transcriptional units were mutated. The fact that these specific parts of the branches were mutated suggested that the proteins expressed from the plasmids caused significant burden to cells. This intuition was also supported by the fact that the pUC replicon (Yanisch-Perron et al. 1985) used in the backbone plasmids of the MoClo system create a high number of plasmid copies, from 500 to 700 copies per cell (Lin-Chao et al. 1992).

Todecrease the burden caused to cells, I developed a library of MoClo backbone plasmids based on low copy origins of replication (see Figure 2-5A). I mainly used the ACYC family of plasmids, which are based on a p15A replicon (Chang & Cohen 1978) producing about 10-12 plasmids per cell (Sambrook & Russell 2003), and a chloramphenicol resistance gene for selection. I also developed MoClo variants from the CDF plasmids family, based on the CloDF13 replicon (Veltkamp et al. 1979) which produces about 20 to 40 copies per cell (Sambrook & Russell 2003), and with either Kanamycin or Spectinomycin resistance genes (see appendix A for a list of all the low-copy MoClo plasmids). The use of these novel backbone plasmids does not change the procedure for Modular Cloning, except for differing antibiotics used at the selection step.

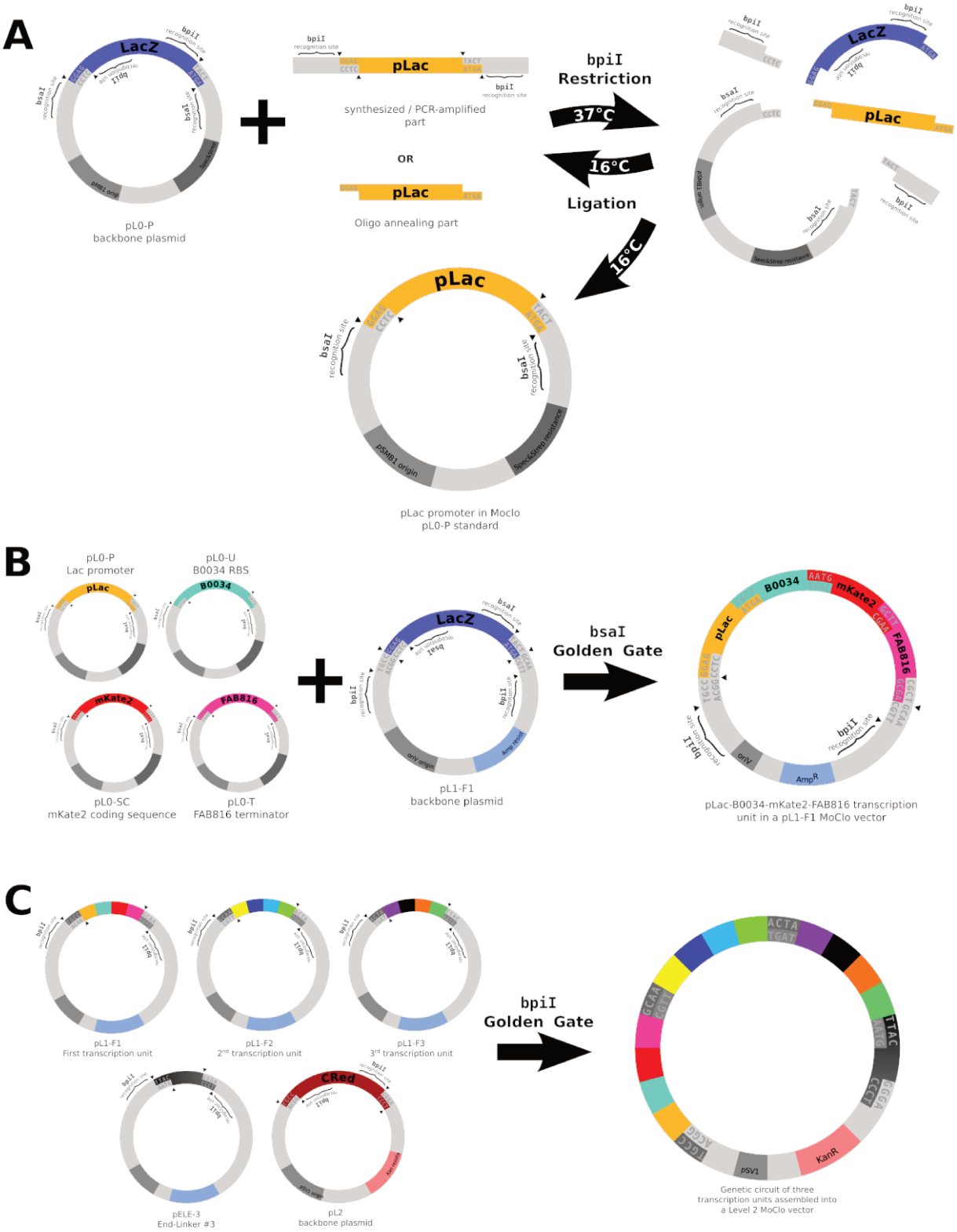

Figure 2-4 The Modular Cloning procedure. A) Level 0 golden gate assembly. The golden gate cloning uses cycles of restriction and ligation to assemble DNA strands robustly and efficiently. When activated at 37°C, the type-IIS restriction enzymes will cut out a portion of the target DNA strands, leaving 4 base pairs DNA overhangs for subsequent ligation. The ligase, when activated at16°C, will assemble some of the DNA strands back into their original configuration, before restriction. But, if the overhangs are chosen properly, the ligase can also assemble the parts of interest together. Although very few will be assembled together in one cycle, the use of type IIS enzymes favors the assembly of the product of interest over time. Since type-IIS enzymes cut outside of their recognition site, the product of interest does not feature the enzyme's recognition site after assembly. On the other hand, strands that re-assemble into the intial DNA parts will still feature the recognition site and can again be restricted during the next restriction cycle. Over enough cycles (typically ~30), most of DNA parts have assembled into the product of interest. This procedure does not require a lot of initial DNA and can assemble

more than two parts at once. In the MoClo version of this Golden gate technique, the overhangs are standardized so that they always ligate in a specific order. The MoClo procedure illustrated in this panel is level 0 promoter integration into the library, i.e. the introduction of a new promoter into a level 0 pL0-P backbone. The bpiI restriction enzyme is used for level 0 assemblies. B) Level 1 assembly. In level 1 assembly a transcription unit is created from level 0 parts on level 0 plasmids (pL0). In this study the level 0 plasmids that will be used are promoters (pL0-P), RBSs (pL0-U), coding sequences (pL0-SC), and terminators (pL0-T). These 4 elements constitute the basic parts of transcription units in bacteria. In this illustration 4 of those parts are assembled together and into a level 1 backbone plasmid (pL1) to form a transcription unit. The bsaI restriction enzyme is used for level 1 assembly. C) Level 2 assembly. In level 2 assembly, transcription units from previous level 1 assemblies are ligated together and into a level 2 backbone to form a genetic circuit. Level 2 assembly is performed with bpiI restriction enzymes.

This change in the copy number of plasmids per cell solved the mutation problem for almost all variants of the toggle switch branches. An interesting outcome is that although the copy numbers of plasmids in each cell with this new library were between 10 and 70 times lower than the original MoClo plasmids, the plasmid yields out of the plasmid preparation steps that were performed at the end of each MoClo step were only 2 to 5 times lower than the typical yields with pUC origins. This result illustrates a tradeoff between yield and the burden on the amplifying cells. Higher copy plasmids may not always be the best choice for MoClo, and for plasmid preparation in general, especially in the case of Golden Gate cloning, which does not require high concentrations of plasmids. A remarkable result of the original MoClo paper is that the authors are able to seemingly easily construct 11 different transcription units and assemble them into a 33 kb plasmid, without burden problems. It is important to note however that the circuits developed are to be expressed in plants, with plant promoters and plant virus RBSs used in the transcription units. So, even though the plasmid preparations are performed in bacteria at each MoClo step, the encoded proteins are not expressed, and the burden is therefore much lower.

Unfortunately, in a few cases I still had problems assembling the plasmids, even with low copy plasmids. I did not know whether expression burden during plasmid preparation was the problem or if it was happening earlier, during DNA assembly. To circumvent the issue without having to pinpoint the source of the problem, I incorporated a new part into my library of RBSs in MoClo level 0 format (see Figure 2-5B and parts 32 and 33 in appendix A). This part consists of type IIS identification sites instead of an actual RBS sequence: this way, non-functional transcription units can be constructed and assembled into circuits, and the real RBSs can be incorporated into the circuit at the last moment (see Figure 2-5C). This solved several problems: firstly, there was no protein expression burden from the level 1 plasmids during plasmid preparation of the toggle switch branches, in which the lac and tet promoters are completely unrepressed. Secondly, the assembly of multiple parts at once, although an advantage of the MoClo technique, is of course less efficient than a one part-one backbone assembly with the same procedure. Therefore, the fact that RBS integration was done at the end of the whole assembly, with only one part inserted at a time, would compensate the ligation difficulties that could be associated to that specific part. Finally, even though this method required one extra step to start from single parts to obtain a final working switch, once I got the version of the toggle switch circuit with the “wildcard” restriction-site parts inserted where the RBSs should be, it took only one extra step to produce a new, different toggle switch. With this circuit now, inserting completely new RBSs only requires one robust and efficient golden gate step. Or two in the case of a “double-wildcard” toggle switch.

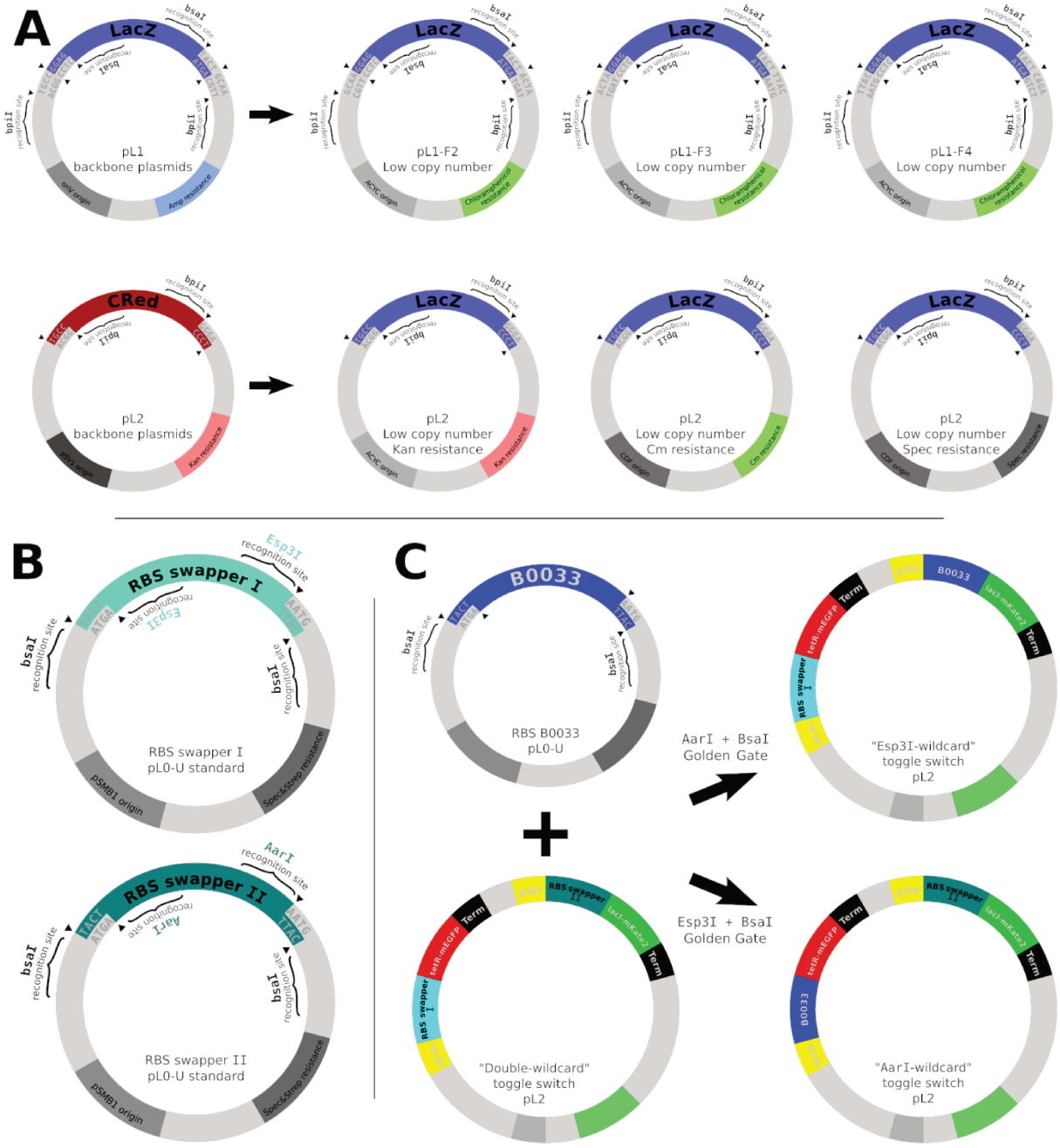

Figure 2-5 New MoClo tools for bacterial constructions. A) Several different MoClo level 1 and level 2 vectors were made on low copy plasmids, to reduce the burden problems caused by over-expression of the transcription units assembled into the vectors. B) MoClo "RBS swapper" or "wildcard": These two short constructs have been synthesized via oligo-annealing and integrated into pL0-U MoClo plasmids in lieu of real RBSs. They feature opposing type IIS restriction enzyme recognition sites that can be later swapped for a normal DNA part in the MoClo standard using the appropriate enzyme and Golden Gate assembly. The same principle can be applied to all MoClo level 0 parts. C) Once one of the two RBS swapper was assembled into each branch of a toggle circuit, robust creation of new toggle switches was made possible in two overnight Golden Gate assemblies of one vector + one RBS part in the MoClo standard.

2.4 Operon branches design

In order to co-express the LacI and TetR transcription factors with fluorescent proteins to measure the state of the toggle switch in real time, I developed operons in which the lacI and tetR genes were transcriptionally coupled respectively to the genes encoding for the mKate2 and mEGFP fluorescent proteins.

Operons are omnipresent in bacteria and are its main method of transcriptional coupling in cases where different proteins must be expressed at comparable levels. RNA polymerases transcribe the coding sequences of all genes in an operon on a single strand of messenger RNA, and ribosomes translate it into proteins from the same strand. Although coupling can be even stronger with fusion protein designs, an operon induces a strong correlation between fluorescence intensity and transcription factor concentration in the cell since most of intrinsic noise seems to happen at the transcription level (Ozbudak et al. 2002).

In an operon, the ribosomes read each coding sequence and release the unfolded protein strand at the end of each CDS when they reach a stop codon. Normal ribosome behavior when they reach a stop codon is to unbind from the RNA strand. However, two mechanisms exist in operons to ensure that the ribosome continues translating after each gene. The first, more common one is the presence of another ribosome binding site in the intercistronic region (Maizels 1974; Schaefer et al. 1989). This allows for different translation rates between the different CDS in the case of operons where proteins need to be expressed in non-equal stoichiometry but still have linear co-expression levels. In those cases ribosomes that just unbound the previous coding sequence immediately rebinds the next ribosomal binding site and continues translation, but other ribosomes can also bind to the second coding sequence directly. A second, more complex mechanism induces translational coupling and co-expression in a 1:1 stoichiometry. In this mechanism, the stop codon of the first CDS is next to, or sometimes overlaps with, the start codon of the next coding sequence, and the end of the CDS of the first gene features a statistically weak Shine-Delgarno sequence[4] (Oppenheim & Yanofsky 1980; Torgov et al. 1998; Govantes et al. 1998).

I developed both types of operon by PCR amplification and re-assembly into a MoClo pL0-SC plasmid similar to that of fusion proteins described in section 2.5. I only developed the versions where the fluorescent reporter genes were placed after the tetR and lacI genes, so that I could infer whether co-expression of both proteins in the operon was happening from simple fluorescence measurements. I then assembled these new CDS level 0 parts into level 1 transcription units for the branches of my toggle switch (see appendix A, level 1 TUs 1.40 to 1.51). I could not get the overlapping stop-start codon strategy to work, and continued working with only the multiple RBS operon designs.

2.5 Fusion protein branches design

Before designing operon-based toggle switches, I tried to design fusion proteins to observe the state of the toggle switch. I designed circuits with the LacI and TetR transcription factors fused to the monomeric fluorescent proteins mKate2 (red fluorescent protein) and mEGFP (green fluorescent protein). I initially went through the trouble of making fusion proteins because measuring their associated fluorescence would give me the state of the toggle switch directly, whereas other methods such as co-expression, in an operon or another transcription

[4] The Shine-Dalgarno (SD) sequence is a part of ribosomal binding sites in prokaryotic messenger RNA. The RNA sequence helps recruit the ribosome to the messenger RNA to initiate protein synthesis by aligning the ribosome with the start codon. The six-base consensus sequence is AGGAGG. Variants of this sequence have more or less chances of recruiting the ribosome.

unit under the same promoter, would not have a 1:1 transcription factor to fluorescent report ratio, as the co-expressed protein levels would be subject to independent noise and independent degradation times.

There are examples in the literature of fusions between the TetR and LacI proteins with fluorescent reporters (Rosenfeld et al. 2002; Webb et al. 1997; Dewar et al. 2004; Marshall et al. 1997; Kato & Lam 2003). For TetR-GFP fusions it has been demonstrated that the TetR transcription factor keeps its DNA-binding property, represses expression downstream of the tet promoter, and is still inhibited by tetracyclines (Rosenfeld et al. 2002). The LacI-FP fusions, on the other hand, were proven to bind the DNA domains, but the expression downstream of the lac promoter as well as the effect of lactose equivalents have not been reported in the literature. Due to the tetrameric nature of both LacI and TetR binding to DNA, it was unclear whether DNA binding of fusion proteins could take place, but the aforementioned studies proved otherwise. The question of IPTG-LacI binding was still open.

I developed the fusion proteins, each with the fluorescent reporter on either the N-terminus or the C-terminus of the transcription factor protein, to obtain the following assemblies: LacI::mKate2, mKate2::LacI, TetR::mEGFP and mEGFP::TetR. I amplified the coding sequences by PCR and inserted type IIS enzymes restriction site at the 5' and 3' tail ends of the amplicons to introduce a standard flexible Serine-Glycine protein linker(Chen et al. 2013) between the two coding sequences (see appendix A). I then transferred them back into standard pL0-SC MoClo vectors to be able to use them in my assemblies. Unfortunately Level 1 cloning was problematic for most of the branches, and I would sometimes get mutations in the linker region in addition to the more usual mutations in the RBS and promoter regions. I did not assemble them into toggle switches right away, but instead assembled them into simpler one-gene-one-feedback circuits (see circuits 2.4 and 2.5) to rapidly find out whether the circuits would respond to inducer inputs and to see if I was able to measure the fluorescence levels.

Although the cells were fluorescent, large foci would appear in the cells indicating toxicity, as well as a remarkable increase in photosensitivity resulting in cell death during the first few hours of time-lapse experiments, and those cells that did not die right away would often elongate or stop growing altogether. This could be a sign of misfolding of the transcription factors, leading to toxic aggregates. It seems however that not all of the transcription factors would misfold, since, in survivor cells, the levels of each fluorescence level could be swayed with IPTG or aTC induction. Still, it became rapidly evident that the catastrophic cellular death rate during observation would make it almost impossible to try to control individual cells on any practical time horizon.

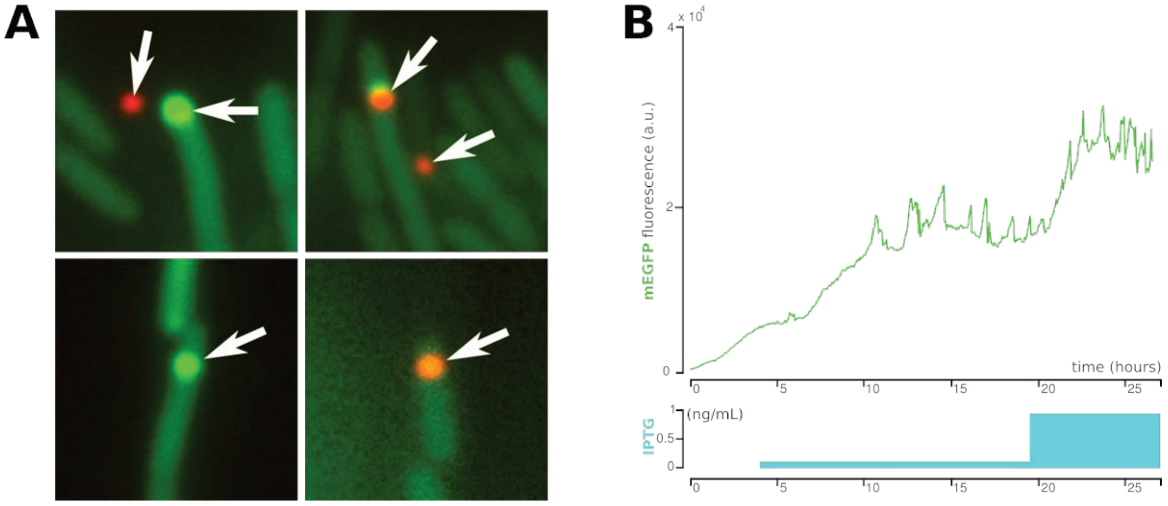

Figure 2-6 Fusion proteins: A) Cells present fluorescent loci in their cytoplasms and die after a few hours of exposition. B) A population of bPH_122 cells transformed with the pConst-LacI::mKate2-pLac-mEGFP circuit is submited to various levels of IPTG. The fluorescence level is influenced by the inducer.

2.6 The toggle switch library

Having developed my strategy to obtain healthy, fluorescent cells, I could proceed with the generation of a full library of putative toggle switches. To maximize the chances of obtaining at least one bistable toggle switch with which to do control experiments, I developed a library of 2x6 toggle switch branches with different expression strengths that I combined together into 36 independent toggle switches (see appendix A, circuits 2.10 to 2.45). I used a common library of RBSs in bacteria and calculated the respective expression strength on each branch with the Salis RBS calculator(Espah Borujeni et al. 2014). The RBS family used spans a wide range of expression level values, although they are not spaced perfectly linearly (see Table 2.6-1). This at least gave me an idea of the range of expression levels at which bistability would occur. I used the MoClo technique and the variants I developed (see section 2.2) to obtain a family of toggle switches.

I obtained a first batch of toggle switches which I transformed into strain bPH_127 to obtain strains bPH_137 to bPH_151. I performed switching experiments in overnight culture to discriminate potentially bistable circuits (see Table 2.6-2 and Figure 2-7). I grew the cells transformed with the different variants of the toggle switch in overnight cultures with high concentrations of IPTG (1mM) or aTC (100 ng/mL), as well as a control tube without inducers, to observe any switching between the two states. A number of circuits would only express one of the two states, demonstrating either unlikely mutations in the corresponding branch that might have happened after sequencing, or a complete imbalance of the circuit in favor of one of the two states. I then washed the cultures twice and resuspended them in growth media with either inducer or without any inducer at all and observed whether they would retain their state or switch again over time.

RBS reference	pTetO-lacI-mKate2 branch Translation Initiation Rate (au)	pLacO-tetR-mEGFP branch Translation Initiation Rate (au)
B0030	3440	7734
B0031	53	458
B0032	12	8788
B0033	27	301
B0034	1035	1624
B0035	251	520

Table 2.6-1 RBS strengths as calculated by the Salis RBS Calculator v2.0. Note the importance of the context of the RBS.

	pTet-mKate2-LacI						
pLac-mEGFP-TetR	RBS B003x	30 (3400)	34 (1000)	35 (251)	31 (53)	33 (27)	32 (12)
	32 (8800)	Switches Not bistable	Not tested	GFP locked	Not tested	Not tested	GFP locked
	30 (7700)	Switches Not bistable	Not tested	GFP locked	Not tested	GFP locked	Not tested
	34 (1600)	Not tested	GFP locked	Not tested	Not tested	Switches Not bistable	Not tested
	35 (520)	Not tested	Not tested	Switches Not bistable	Not tested	Switches + bistable	Not tested
	31 (458)	RFP locked	Not tested	Not tested	GFP locked	Not tested	GFP locked
	33 (301)	GFP locked	Not tested	Not tested	Not tested	Switches + bistable	Not tested

Table 2.6-2 Summary of early characterization results. A corresponding table listing the circuit numbers can be found in appendix A.

Out of all tested circuits only two were capable of switching and were bistable. I proceeded with circuit 2.31 (strain bPH_142) because it produced the most well-defined states (the expression level in either states was high and the expression of the opposite protein was almost entirely repressed) and would remain in either state for over 8 hours without inducers. Figure 2-7 shows the switching behavior of several circuits as well as the stability of strain bPH_142 over time.

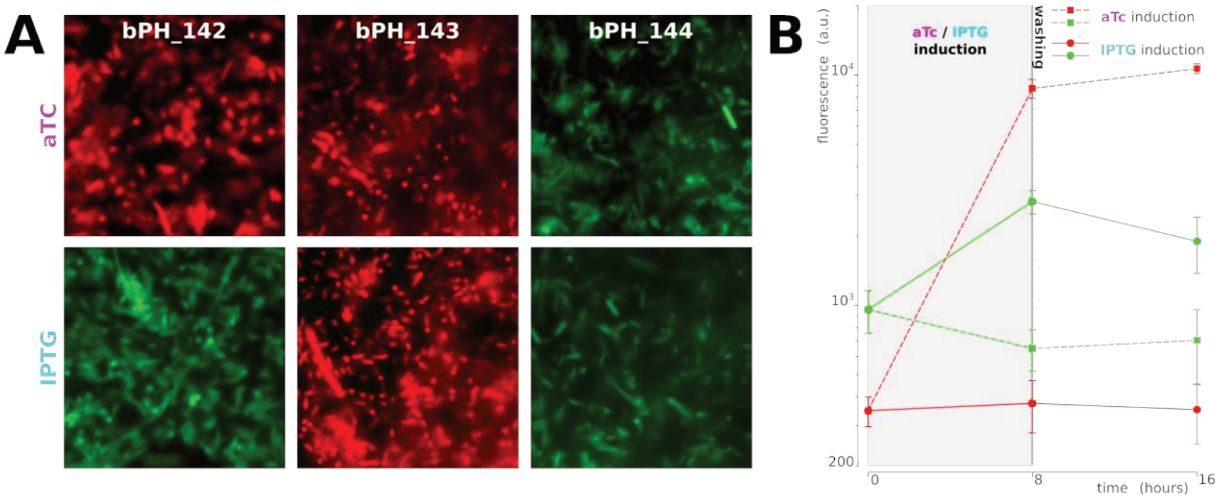

Figure 2-7 Behaviors of different toggle switch circuits. A) Different circuits may or may not switch when subjected to inducers. B) Strain bPH_142 is stable over long periods of time. The cells were grown overnight at 37C in LB without inducer. They were re-suspended for ~4 hours in fresh media. Their fluorescence was measured using epifluorescence microscopy and then the cells were re-suspended for 8 hours in fresh media with inducers (either 100 ng/mL of aTC or 1mM of IPTG). After 8 hours their fluorescence was measured again, and the cells were washed twice and then re-suspended in fresh media without inducer. Finally, fluorescence was measured after 8 more hours. The red curves shows the mKate2 level and the green curves show the mEGFP levels. Dashed curves represent the fluorescence levels of the cells grown in aTC for the first 8 hours, and the solid curves represent the fluorescence levels of the cells grown in IPTG for the first 8 hours. The 2.31 toggle switch in strain bPH_142 showed impressive stability in batch culture, even after more than one washing.

3 Microfluidics & automation

Containing and observing single bacteria over extended periods of time is a difficult task. Time-lapse microscopy and microfluidics have been around for some time now, but the automation of acquisition, especially when several experiments are run in parallel, is still not completely straightforward and requires a non-negligible amount of work before experiments can be run routinely. Another crucial aspect is concurrent cell containment and culture: cells should be observed in their normal state of growth, but should also be contained so that they do not leave the field of view of the experiment. This difficulty is exacerbated in the case of single-cell observations. Image analysis is also a complex problem, in particular in bacteria. This difficulty is also exacerbated in the case of external control, which introduces requirements of robustness and reliability. Finally, in the case of external control or dynamic stimulations in general, input delivery has to be automated and interfaced with the control algorithm.

In this section I present how I tackled these different problems. I developed several microfluidics devices to contain and observe the cells, and make rapid changes to their chemical environment possible. I developed software and hardware to control solenoid valves

for input delivery. Finally, I wrote software for microscope control, basic image analysis, control and communication with the valves controller, thus closing the loop and allowing for cellular control.

3.1 Microfluidics

I had the opportunity to access a clean room facility at Paris-Diderot University where I could experiment with lithography and microfabrication techniques, and design my own wafers for microfluidic devices. Our team has a history of developing its own microfluidic wafers for S. cerevisiae (Uhlendorf et al. 2012; Vulin 2014; Llamosi et al. 2016) and I took advantage of the expertise in the team to develop my own devices for bacteria.

3.1.1 Molded PDMS devices

The dominant approach for making microfluidic devices is based on soft lithography and PDMS molding. A silicium wafer is covered in a layer of epoxy-based resin of known thickness. The resin is insulated to obtain a desired pattern on the wafer. The insulation techniques used in this thesis are described in the following two sections. Once the desired pattern has been drawn, the so-called master wafer is used as a reusable negative for replica-molding of the pattern with Polydimethylsiloxane.

Polydimethylsiloxane (PDMS) is a transparent silicon that is a viscous fluid in its monomeric form, and an elastic solid when homopolymerized. Liquid PDMS supplemented with a curing agent to catalyze polymerization is poured onto the wafer mold. The liquid PDMS will perfectly follow the mold's shape and then cure into its solid polymerized form. To form a working microfluidic device, the solidified silicon is peeled off the master wafer and stuck onto a glass slide. The glass slide floors the circuit and the device can then be used to load cells, flow media and various chemical, and the chambers and channels can be observed through the glass slide on a microscope. Transmitted light can be shone through the device since PDMS is transparent.

3.1.2 Microfluidic wafers fabrication methods

Various methods exist for developing microfluidics devices for in vivo cell cultures and observation, such as wafer engraving, 3D printing, or CNC milling. The most common technique is so-called 'soft' lithography, in which layers of epoxy-based resins are insulated locally to create patterns on a silicium wafer. This method is described in this section and was used to develop the microfluidic devices presented here.

3.1.2.1 Photolithography

The most common way of designing microfluidic devices is by coating a wafer with an epoxy-based resin, the SU-8 resin developed in the 1980s by IBM, and then illuminating this resin with high intensity UV light. The light goes through a custom made negative mask and only the desired parts are exposed to the high-energy photons. The resin is photosensitive and will solidify when exposed to high-energy particles. A solvent bath will dissolve all parts that were not exposed to the UV light, and only the exposed part will remain. Coating of the wafer is performed with a spin coater, and rotational speed is adjusted to obtain precise layer thickness. Several variants of the resin exist, of different viscosities, to allow for a wide range

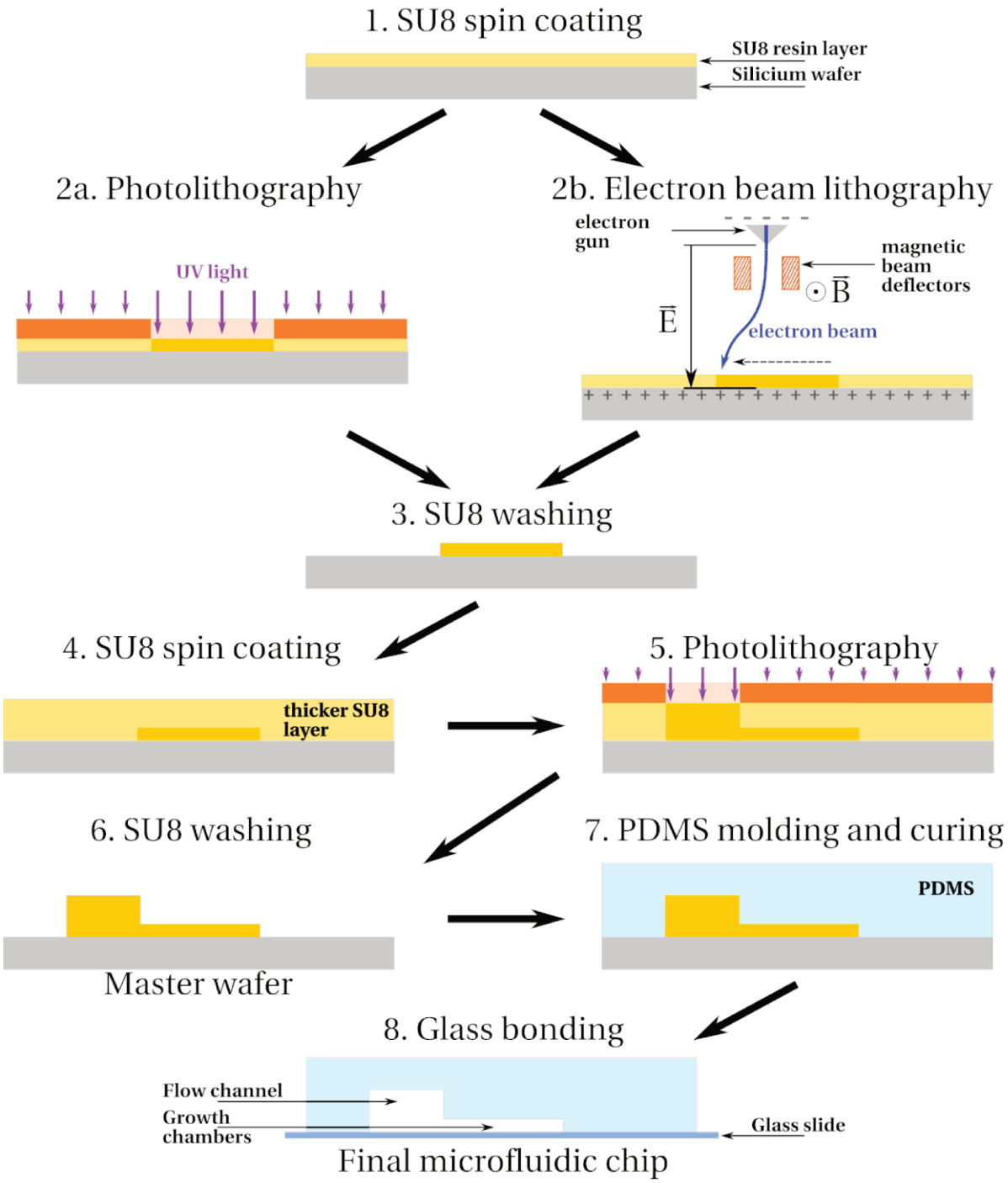

Figure 3-1 Microfluidic devices fabrication protocol. Steps 1 to 6 are microfabrication steps to develop the master wafer and are performed in the clean room. Steps 7 and 8 are the actual microfluidic chip assembly steps and are performed in the lab, every time a new microfluidic chip is needed. 1) Epoxy-based SU8 resin is spread over a silicium wafer. The wafer and the resin are spun at a high speed (typically 2000 to 8000 rpm) to obtain desired layer thickness. This process is known as spin coating. 2) The resin is solidified by exposition to high-energy particles: 2a) UV light is shone through a patterned chrome mask to insulate parts of the SU8 resin. This process is known as photolithograpy. 2b) An electron beam is scanned through the SU8 layer following a pre-programmed pattern. Magnetic deflectors are used to insulate the pattern at nanometric resolutions. 3) Non-insulated SU8 resin is dissolved and only the insulated pattern remains. 4) A new SU8 layer of higher thickness is spun onto the wafer for producing objects of different height. 5) The higher objects are insulated via photolithography. In our case, the main flow channels are produced via photolithography because they are too large for EBL. 6) non-insulated resin is washed away. This final product of microfabrication is the master wafer, i.e. the wafer that will be used as a re-usable negative mold to produce microfluidic chips: 7) Polydimethylsiloxane (PDMS) is poured on the master wafer and cured for about 8 hours at 65C. After curing, the once fluid PDMS solidifies. 8) The solidified PDMS countermold is stuck onto a microscopy glass slide by plasma bonding. This final microfluidic chip will be used to grown cells and deliver nutrients and chemicals. In our design, the smaller regions are the growth chambers, and the bigger and higher regions are the main flow channels. Chips are used for a single experiment and then a new chip is produced from the master wafer.

of possible layer thickness (from about 500nm to 100µm). Several objects of different heights can be obtained on the same device by sequentially applying different layers of SU-8 and insulating them with UV light one by one through patterned chrome masks (see step 2a in Figure 3-1). A review of the type of devices that can be obtained with this procedure is given in (Ng et al. 2002).

Although photolithography is a relatively simple and fast process, it is limited in its resolution by the diffraction of light on the insulation mask. Details under 5 µm start getting difficult to obtain and require the use of vacuum chambers and optical filters. The closer the dimensions get to the wavelength of UV light, the more imprecise the details get.

3.1.2.2 Electron-Beam lithography

For smaller details, typically around 1 micron or less, I had to use electron-beam lithography, or EBL: Since using a lithography mask causes diffraction that limits the resolution of the insulation, the solution is to direct the high-energy particles beam directly towards the wafer and scan it through the parts to be polymerized. However, mechanical orientation of the beam is very imprecise, and a light beam cannot be bent easily towards a specific location on the wafer. Electron trajectories, on the other hand, can be very precisely oriented with an electromagnetic field, as is routinely done in a scanning electron microscope. In practice, electron-beam lithography actually uses slightly modified scanning electron microscopes to guide an electron beam over a resin layer, following an insulation path predetermined numerically by the experimenter (see step 2b in Figure 3-1). Similarly to photolithography, the high-energy electrons create Lewis acids in the resin and catalyze the homo-polymerization reaction.

EBL allows for nanometer-precise details, and since no physical mask is used, the variety of shapes, sizes, details and structures of the microfluidic circuits obtained is not limited by the price and delays linked to the manufacturing of a photolithography mask. It cannot be used on thick layers of resin however, since electrons scatter in matter as they lose energy penetrating into SU-8. Increasing the energy of the electrons allows for deeper penetration, but the penetration depth increases only logarithmically with the electrons' energy (Suñé 2008). The working surface on our apparatus is also limited to, at best, 1mm. One can insulate several working surfaces, but the details may not be perfectly aligned from one working area to the next since the displacement between scanning surfaces is performed mechanically.

Although this method has been used extensively in other domains of micro- and nano-fabrication (Tseng et al. 2003; Suñé 2008), and has been demonstrated to be a versatile micro- and nano-fabrication method in SU8 resin (Kudryashov et al. 2003), its usage in microfluidic devices fabrication is still anecdotal (Mali et al. 2006). I started working on this technique after discussing my light diffraction problems with quantum physicists at the clean room who used this method routinely on other resins. Although the production of a complete device with EBL and photolithography is more tedious than with photolithography only, iterations on the EBL part of the designs can occur at a much faster time scale because the technique does not require the edition of a new insulation mask after each modification to the desired pattern. I

could test several different designs in a day. I tested the limits of the method and observed that I could produce objects that were about 100nm in width without much difficulty.

3.1.3 Main device – Mother Machine

The main microfluidic device used in this thesis is a so-called "Mother machine" (P. Wang et al. 2010) device that traps individual cells in narrow growth chambers. A team of collaborators (Lindner team, U1001 INSERM) uses such a so-called mother-machine microfluidic device that consists of one single nutrient channel flanked by 0.6-to-0.8µm-high chambers that are slightly wider than an E. coli cell (about 1 µm wide). In this setup the cells do not grow next to each other and instead form one single line of mother-daughter pairs. In addition to greatly simplifying segmentation (as cells are contained in a predefined space), the cell that is at the end of the chamber is never washed away and that cell can be observed for extended periods of time. Our collaborators use this setup to study the evolution and aging of one cell over weeks.

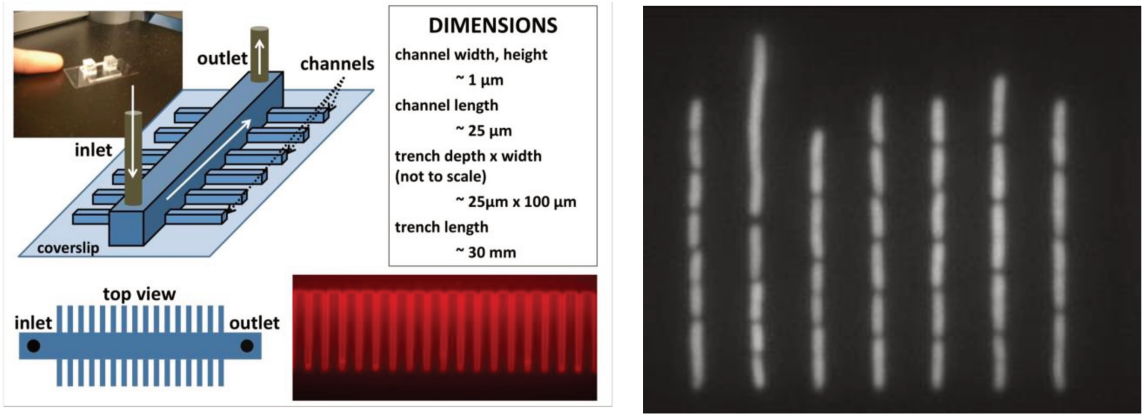

Figure 3-2 – From Wang et al.(P. Wang et al. 2010) The so-called mother machine microfluidics device. Cells are trapped in chambers of about 1µm width and grow in single cell lines. The cells at the bottom of the chambers are not pushed away and remain there until they die. Others are pushed to the end of the chamber and washed away in the main channel.

I started working on this type of device with molds that were provided by our collaborators. The first results were promising and I started making my own molds in the clean room. Since the details on those circuits were close to the theoretical minimal resolution of photolithography, the traditional method for creating microfluidic devices, I had to use a combination of traditional photolithography for the larger elements, like the flow channels, and Electron Beam Lithography for resin insulation for the narrow chambers.

To accelerate my research and obtain more data at once, I designed a new device based on our collaborators' mother machine design that can run up to six independent experiments at once. Alignment of multiple elements between EBL and photolithography is tedious and error-prone, and the development of entire wafers with up to 6 different EBL motifs required significantly more time and care to develop than photolithography-based wafers. Because of their dimensions, the wafers are also more delicate. To summarize, one of the inconveniences of this method is that wafers are difficult to obtain and cannot be replicated. The destruction of the master wafer would stall the experiments, and it has to be handled with extra care.

Cell loading is another known problem in mother-machine-type devices. Because the chambers are narrow and only one of their ends is open, it is difficult to get the cells inside

them. Two methods exist: 1) let cells grow everywhere in the microfluidic chip. The cells end up colonizing the chambers and at this point the cells in the flow channel are flushed away. This method has various limitations: first of all, it requires setting up the experiment a long time in advance to let the cells grow into the chambers. The second problem is that flushing will not be 100% efficient, and some cells will remain in the channel flow after flushing and grow in the flow channel. 2) Centrifuge cells into the chambers. I designed a microfluidic chip centrifugation arm that can be used with a spin coater to ease cell loading. Once the cells were loaded I would set up the chip in the input delivery and flow apparatus and on the microscope for single cell observation.

3.1.4 Early device designs: Population chambers

I based my first wafers on the device developed by Jannis Uhlendorf in our team for his control platform for yeast (Uhlendorf et al. 2012). I adapted it to get the right height for the chambers to make sure that my bacteria would not grow in several layers in the chambers or be able to move at all once in there, but of course the chambers also had to be high enough to allow the cells to enter in the first place. I found that the optimal height for the growth chambers was 0.7µm. Because the chambers were wide compared to their height of 0.7µm, the ceiling of the chambers would often collapse.

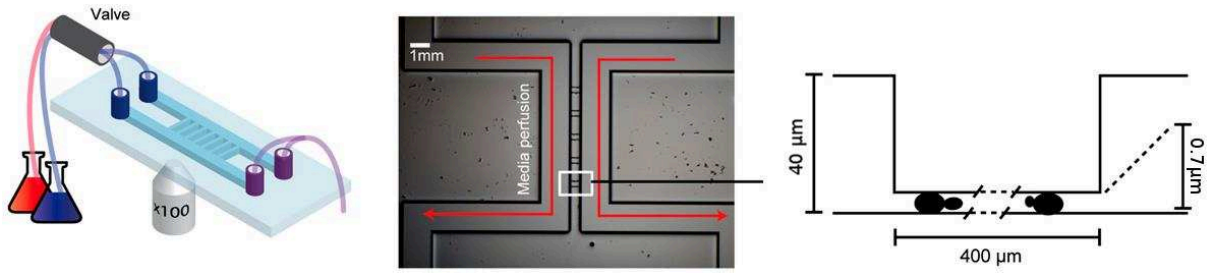

Figure 3-3 – Adapted from Uhlendorf et al.(Uhlendorf et al. 2012) The first device that I used was based on this design. (left) the general microfluidic setup, the device is connected to a valve that selects between two media. The media are flown through the channels and end up in the waste (purple tubing). The chambers are between the two channels and nutrients and drugs are transmitted to the bacteria by diffusion. Epifluorescence microscopes are used to image the cells from below, inside the chambers. (middle and right) The chambers are 0.4mm in length, and about 0.7 µm in height. The channels are 40 µm in height.

To work around this issue I then adapted another device developed for yeast by Clément Vulin, another Ph.D. student on the team, to my organism. This second device was similar to the previous one, except that it featured chambers of various widths, which solved the collapsing ceiling problem. I was able to obtain good images of my bacteria where they would grow as a single layer. Those two devices did not feature details smaller than 5 microns, and I used only photolithography (section **3.1.2.1**) to create their molds.

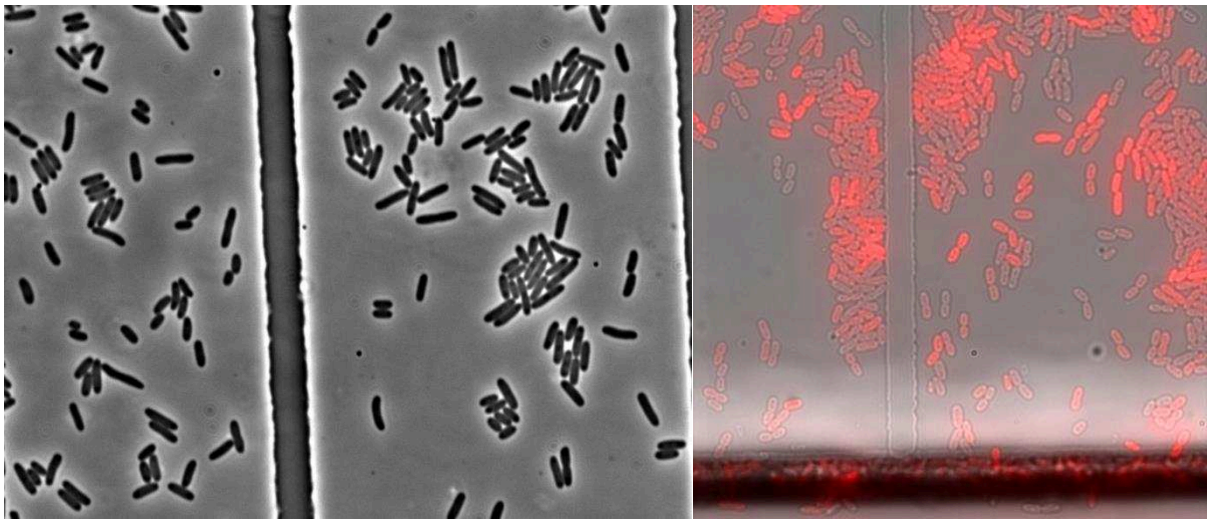

Figure 3-4 – Images obtained with the second device. The cells do not move in the device or grow as multiple layers and the quality is good enough to try single cell segmentation. (left) Phase contrast 100X imaging of E.coli cells. (right) 100X composite image of red fluorescence imaging and transmitted light imaging of E.coli cells.

I could not use those devices for single-cell control experiments. I did use them at the beginning of my PhD for population control, but the lack of robust segmentation algorithms for single-cell tracking and the rapid and free movement of cells in and out the field of view made them not adapted to online single-cell tracking. However the number of cells in the field of view is 10 to 100 times higher than in mother machine devices, and population effects can also be observed in this type of devices. Recent progresses made in cell segmentation might make it possible to monitor the fluorescence level of thousands of single-cells in a 2D layer on-the-fly (see chapter 4 for a short review of the state of the art in cell segmentation).

3.1.5 Surface passivation, clogging, and flushing

Bacteria tend to grow everywhere in microfluidic devices, both on PDMS and glass. This can cause several problems: First of all the accumulation of cells in the channels just outside of the growth chambers can exert pressure on the cells inside the chambers and have an impact on their growth rate. Secondly, the number of cells growing in the delivery channels would keep on increasing, possibly up to a point where nutrients and input chemicals are segregated or consumed by those cells, hindering the controllability of the cells inside the growth chambers. Finally, bacteria tend to aggregate and form biofilms, and an uncontrolled growth in the main flow channel would rapidly lead to clogging of the channel, possibly with biofilms forming in the entire apparatus.

I have faced all the above scenarios, both in mother machine devices and in monolayer devices, and the necessity of good control of bacterial growth outside of the dedicated chambers rapidly became evident. Surface passivation was a first step to help reduce the spread of bacteria in main channel. I started using bovine serum albumine (BSA) with the population device described in section 3.1.4, and the results were promising. I followed the same procedure with the mother machine device but two problems arose: first of all, BSA is a fragile and somewhat expensive protein. Its passivation effect degrades with time and new stocks must be prepared periodically. It also degrades throughout the experiment and after 12 to 16 hours bacteria start growing in the channels again. The second, more inconvenient

problem was that cell loading in the mother machine chambers was harder after BSA passivation. Whether it was caused by accumulation of BSA in the chambers, a higher fluid viscosity or other effects is unknown, but I had to passivate the surface after cell loading, which was both more complicated and less efficient since some cells were already stuck to the surfaces of the device.

I switched to a cheaper and more robust alternative with Pluronic F-127. Surface passivation with pluronic prior to cell loading did not pose the same problem as BSA. Pluronic is a cheaper and more stable molecule, so not only was storage not a problem, I could also mix Pluronic into the growth media that would later be flown through the microfluidic device and autoclave it, which meant that surfaces would not be depleted of Pluronic over time.

To further reduce cell growth outside of the dedicated chambers I would also periodically flush them by running the peristaltic pump at full speed for a short period of time (usually for 30 seconds every 30 minutes). The custom program interfacing with the pump would also communicate with the acquisition session, to ensure that the pump was not flushed during acquisitions: With the pump running at full speed, the minor change in pressure in the chip would suffice to slightly distort the shape of the chambers, thus reducing the performance of the autofocus, drift correction, and image analysis. See section 0 for more details.

3.1.6 Input delivery

Characterizing the library of putative toggle switches and carrying out PI control experiments required the possibility of injecting variable levels of the two inducers in the microfluidic chip, so I had to develop some sort of mixing apparatus. While a variety of microfluidic mixers exist in the literature (Lee et al. 2011), none of them was suited to our approach. Passive mixers introduce delays in the time between mixing and delivery or require complex microfabrication techniques, and active methods require even more complex assembly methods, sometimes alongside the integration of bulky and obstructive hardware close to the chip. Moreover, those systems require some form of pressure control system, which is itself a costly and finicky piece of equipment.

In our implementation of microfluidic mixing, we assumed that the process of delivery through the microfluidic tubing and channels via peristaltic pumping, and the diffusion through the cellular membrane, acts as a low pass filter for chemical signals between the environment and the cytoplasm, and therefore would tend to average out fast oscillatory changes in the environment. Such an assumption makes it possible to use pulse-width modulation in our drug delivery method. To get a certain “sensed” concentration of inducers for the cell, we switch between three inputs: one with a high concentration of IPTG (1mM), one with a high concentration of aTC (100ng/mL), and one without inducers for dilution. For a certain concentration of inducer, the duty cycle is adapted and diffusion in the tubing and the device and through the membrane does the rest (see Figure 3-5). To ensure that the average concentration sent to the cells was the one indicated by our choice of duty cycle, we calibrated the duty-cycle to concentration ratio by running the valve controller for extended periods of time at different duty-cycle values and measured volume depletion. We then used this 3-way PWM valves setup to mix solutions of fluoresceine and rhodamine and measured

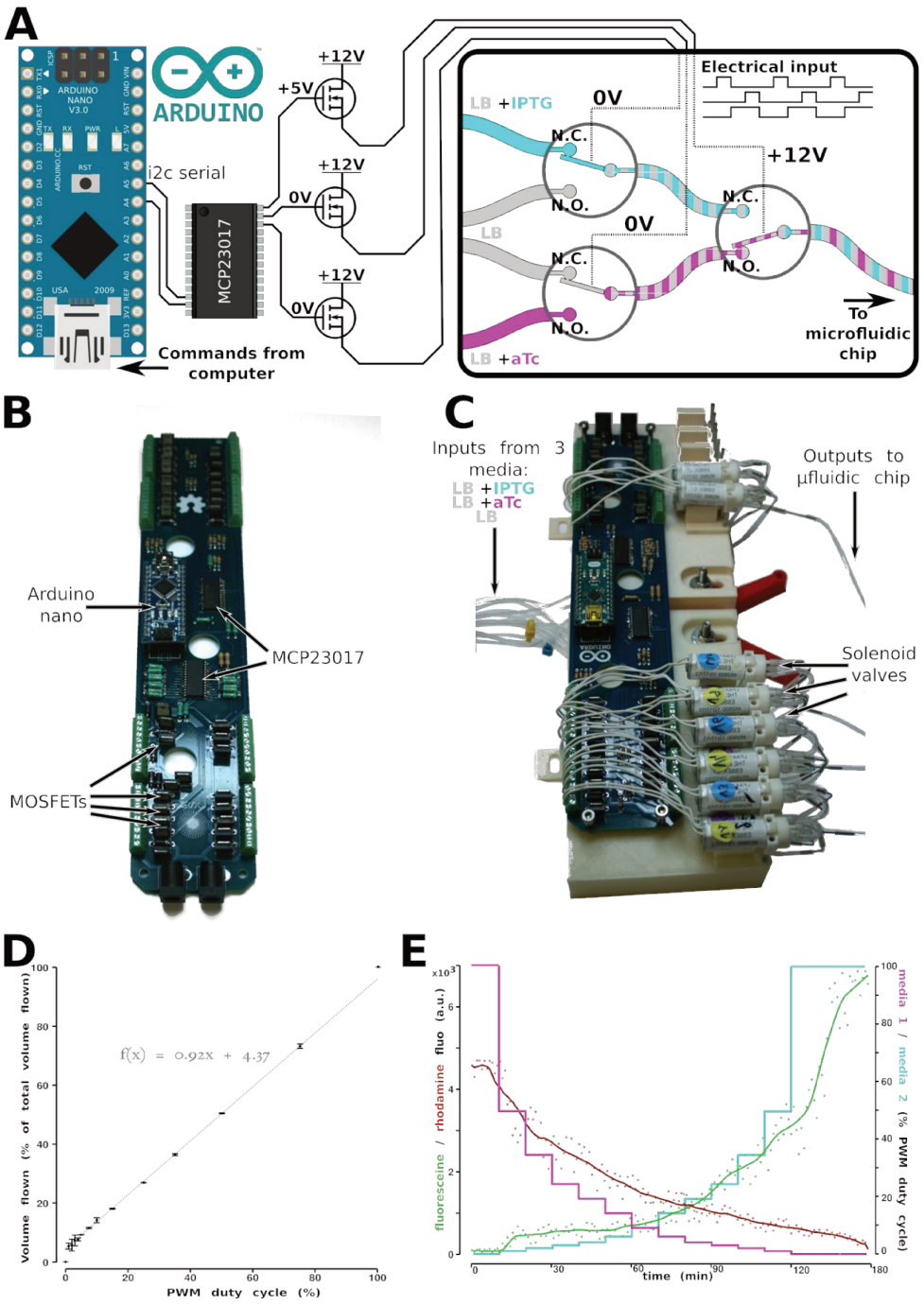

Figure 3-5 The valve controller and mixing apparatus. An Arduino Nano can control up to 8 MCP23017 chips over i2c, and each of those chips can control up to 16 different transistors. The transistors are necessary to handle the power necessary to actuate the solenoid valves used for PWM mixing of the three input media: Luria-Bertani (LB) growth medium, LB + 1mM IPTG, and LB + 100ng/mL aTc. The Arduino receives mixing commands from the computer, and handles the sequential updating of the state of the MCP23017 chips to mix the three media. B) The actual printed circuit board. Up to 4 board can be connected together over i2c to control up to 128 valves with a single Arduino. C) The valve actuation setup partially assembled. A support that also distributes the different input tubes among the valves was 3D printed to fit within the microscope's temperature control box and attach to the microfluidic chip holder. D) Calibration experiment: PWM mixing was run over extended periods of time at different duty cycles, and volume consumption was measured. The results were surprisingly precise, and did not vary much between valves sets or between inputs. However, the correspondence between the PWM duty cycle and the volume of input actually flown through the controller was not completely linear. E) Various concentrations of fluoresceine and rhodamine were flowed through the microfluidic device (mother machine) and fluorescence was measured in the chambers. The fluoresceine level nicely follows the mixing levels imposed by the valve system for media 2. However it was observed that rhodamine would take a longer time to be flushed out of the chip by lower mixing concentrations. Clumps of rhodamine would be visible in the microfluidic chambers. This could also be another explanation for the delays we could observe over actuation.

fluorescence levels in a microfluidic chip to double check our result and also to verify that cells received a mixed level of both inducers (see Figure 3-5).

To be able to alternate fast enough between media and also to resist high flow rates and high pressures, we used valves from the Lee Company. These valves operate on 12 V and can draw up to 100 mA. For those reasons they could not be operated with a simple Arduino and a power adaptation circuit was necessary, especially to drive more than one valve. We designed a custom arduino-based circuit board to host all electronic components and act as an interface between the computer and the solenoid valves. The board can drive up to 36 different valves. To optimize space, adaptability and ease of use, the card is built around an Arduino nano board that drives two MCP23017 chips[5] over i2c serial, which themselves drive power MOSFETs that drive the valves. The entire design and code for Matlab and the Arduino Nano can be found on github[6]. I have since developed new, more compact implementations of this board to drive up to 40 solenoid valves for other projects that can be found on the same page.

3.1.7 Reference crosses

In the next section (3.2) I am going to present some of the image analysis and data extraction algorithms I used to monitor cells fluorescence. But an important step prior to that is the optimization of the acquired images. The image analysis is greatly simplified if the images are of good quality and their acquisition conditions are consistent with eachother. Good focussing is an important aspect of this process. A microscope does not stay in focus on the observed specimen over time if its position is not corrected periodically. Images are acquired automatically and in real time, without user supervision and usually at several different positions on the chip. All these issues further complicate the task of acquiring good quality, consistent data.

Autofocussing is a non-trivial problem in time-lapse microscopy, and online image analysis requires a precise and robust autofocus for optimal quality data. A number of autofocussing algorithms exist, and can be separated in two groups: hardware based autofocus and image-based autofocus. The hardware based ones measure the distance between the objective and the specimen observed, usually using laser triangulation, and try to maintain this distance constant. Although this approach is usually more robust, it does not measure image properties directly, and uses the proxy of the measured distance instead. Unfortunately, the measure can drift over time, especially in the case of immersion oil objective. The changes in focus are usually small, but in the case of bacteria, minor variations in focus can have dramatic effects on the quality of the extracted data. The second approach measures some property of the images acquired (usually average image gradient) and tries to minimize or maximize it. Although this can produce satisfactory results, the robustness of such a method is often inadequate for time-lapse microscopy of live cells. Because the cells are growing in the field of view, the measured properties of the image change over time, which can lead to focus drift

[5] MCP23017 are port expansion chips, made specifically for increasing the number of I/O ports on a microcontroller, like an arduino. Up to 8 chips can be connected on a 2-wires i2c serial bus, which means a single arduino could drive up to 128 valves.
[6] github.com/Lab513/ValveControllerNano

or sometimes complete loss of focus. A significant advantage of microfluidics in this situation is cell containment: there are parts of the device where the cells cannot grow.

Since conventional focusing techniques were another source of uncertainty in the program, I took advantage of this property of microfluidics and combined it with image-based autofocussing techniques. I added crosses to the design of my devices that the cells could not grow into, and used those crosses as reference objects for autofocussing: At the beginning of time-lapse acquisition a z-stack of images of the crosses would be acquired, and at every time point it would be used as a reference for focusing the image. Since the appearance of the crosses did not change over time, the focus was extremely robust. By using an entire stack as reference instead of a single initial image of the cross, information hidden in the out-of-focus frames of the stack significantly increases the precision of the autofocus correction. This is one of the intuitions that led me to developing the segmentation algorithm described in chapter 4.

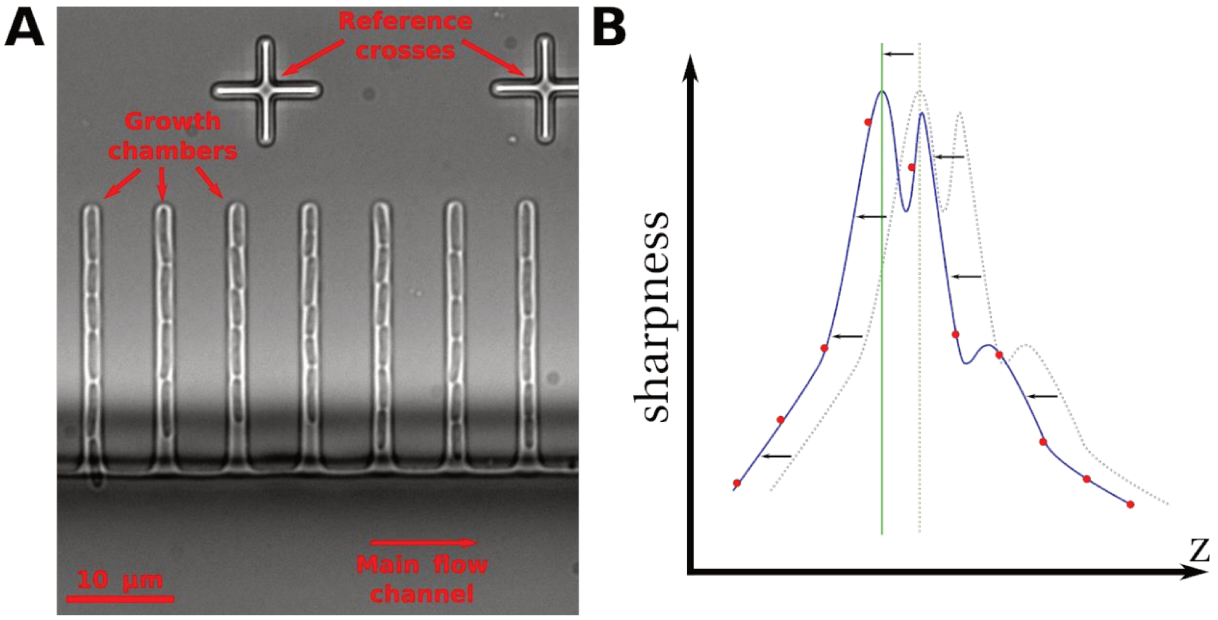

Figure 3-6 Mother Machine and autofocussing. A) The final microfluidic device. B) Typical autofocusing curve. The image sharpness measured through a stack during an acquisition (in red) is compared to the curve acquired at the beginning of the experiment (in blue) and the new in-focus frame is inferred from it (green line). Using the entire stack for comparison is more precise than using a single image as reference.

3.2 Image analysis

Single cell segmentation is a notoriously difficult problem. Tracking is even worse, since segmentation errors tend to cause tracking errors that accumulate over time.

It is especially critical in the context of external control, because control platforms require robust, precise, fully automated, and unsupervised cell segmentation and fluorescence sensing. It is one of the key challenges in external control theory of singe cells. In this platform we greatly simplified the problem by trapping cells in one-cell-wide chambers, where the mother cells can easily be tracked over extended periods of time.

In chapter 4 of this document I will present a segmentation algorithm with high segmentation accuracy on our mother machine images, but also on a wide range of other type of

microfluidic devices or even different organisms. We have used it to perform robust real-time segmentation of our cells on this platform, but not on closed loop experiments. This algorithm has been designed in my last year of PhD and has evolved continuously during that time period. To keep data consistent between experiments as well as to avoid the delays that a full integration into the platform might cause, we decided not to use it in our control experiments. Prior to the development of this technique however, we developed several cell segmentation techniques. The first, most robust one, simply extracts the fluorescence from the end of the mother machine chamber. This simple method was used for online image analysis for fluorescence extraction because it would perform robustly throughout the experiment. Other methods were also developed for a posteriori image analysis.

3.2.1 Robust algorithm for online measurement of fluorescence levels

Cell segmentation is performed a posteriori because it usually requires tinkering and often trials-and-errors to work robustly on an entire dataset, and its performance can vary from one dataset to another to another. Most cell segmentation approaches still require a significant amount of manual corrections by the user to reach satisfactory tracking performance. Because of this, traditional segmentation approaches based on morphology or active contours could not be used on online experiments. A much simpler image analysis technique was used for the majority of the toggle switch analysis.

In time-lapse acquisitions of the mother machine device, the mother cells at the end of the mother machine chambers stay in the same position in the image throughout the entire experiment. Bacteria reproduce by dividing in the middle, so some part of the image always included a significant portion of the mother cell. The algorithm used for online image analysis simply extracts the mean fluorescence from that part of the image.

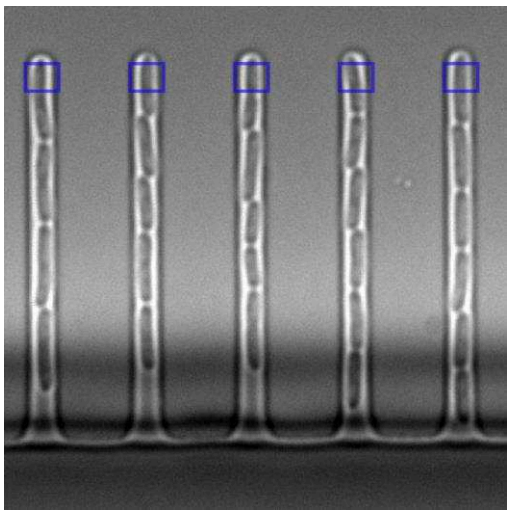

Figure 3-7 The simplest, and most robust method, relies on the shape of the mother machine device, where mother cells are trapped at the end of the growth chambers, to extract fluorescence from a pre-arranged ROI. Motion compensation algorithms are used to ensure that the same part of the chamber is sampled at every timepoint.

At the beginning of the experiment, the user selects the chambers to observe throughout the experiment and draws a rectangular region of interest (ROI) of 20 by 10 pixels where the mother cell is. During the time-lapse acquisition, lateral drift is corrected, and the fluorescence in GFP and RFP is measured and sent to the control algorithm.

3.2.2 Image gradient and ellipsis fitting on mother machine images

With the mother machine microfluidic device, the segmentation problem was greatly simplified, and the algorithm described above was sufficient to extract fluorescence levels for the mother cells, but could not identify daughter cells and extract their fluorescence. Here I describe a simple segmentation algorithm to identify single cells in the chambers and measure their fluorescence.

Identifying growth chambers is a simple task, since they are well separated and of known size and shape. With the chambers identified, the complexity of segmenting cells in each chamber decreased significantly. The cells cannot grow side-by-side, which is a major problem for morphology or active contour methods (see Figure 3-10 and chapter 4), and also grow in a well-aligned strand of cells. One of the disadvantages of the narrow chambers though is the proximity of the cells borders with the chamber walls, which can interfere with morphology operations such as range filtering or with active contour methods.

I developed a custom segmentation program that circumvents the problem of the PDMS walls and identifies the separations between cells robustly. To avoid dealing with the walls altogether, only the image gradient along the chambers axis was considered and not the 2D image gradient commonly used for edge detection. In combination with morphology operations and thresholding the identification of cells separation would be easily and robustly obtained throughout an entire image sequence. Ellipses were then fitted to the identified cell regions to identify elliptic ROIs for fluorescence extraction. Although ellipses are not the best model for describing the shape of Escherichia coli bacteria, the minimal number of their parameters allowed for fast and robust identification with known algorithm. More complex descriptions of the shape of the cells would have taken more time to fit and be more prone to errors. Moreover, it became rapidly apparent that even in the simplified setting of the mother machine, considerable user oversight was still necessary for the segmentation to run smoothly in online experiments.

The main problem emerged from the fact that between experiments lighting and focusing conditions are never exactly the same. Even throughout an experiment and even with state-of-the art autofocussing, focus might drift slightly over time. Aging cells also present visible differences from young cells. All those changes require fine-tuning of the few parameters of the algorithm before each experiment, with the hope that the images will not change too much throughout the experiment. So even though the algorithm I developed can segment an entire time-lapse automatically with almost no segmentation errors a posteriori, it could not be used for online segmentation since in this case the optimal segmentation parameters must be guessed prior to the experiment.

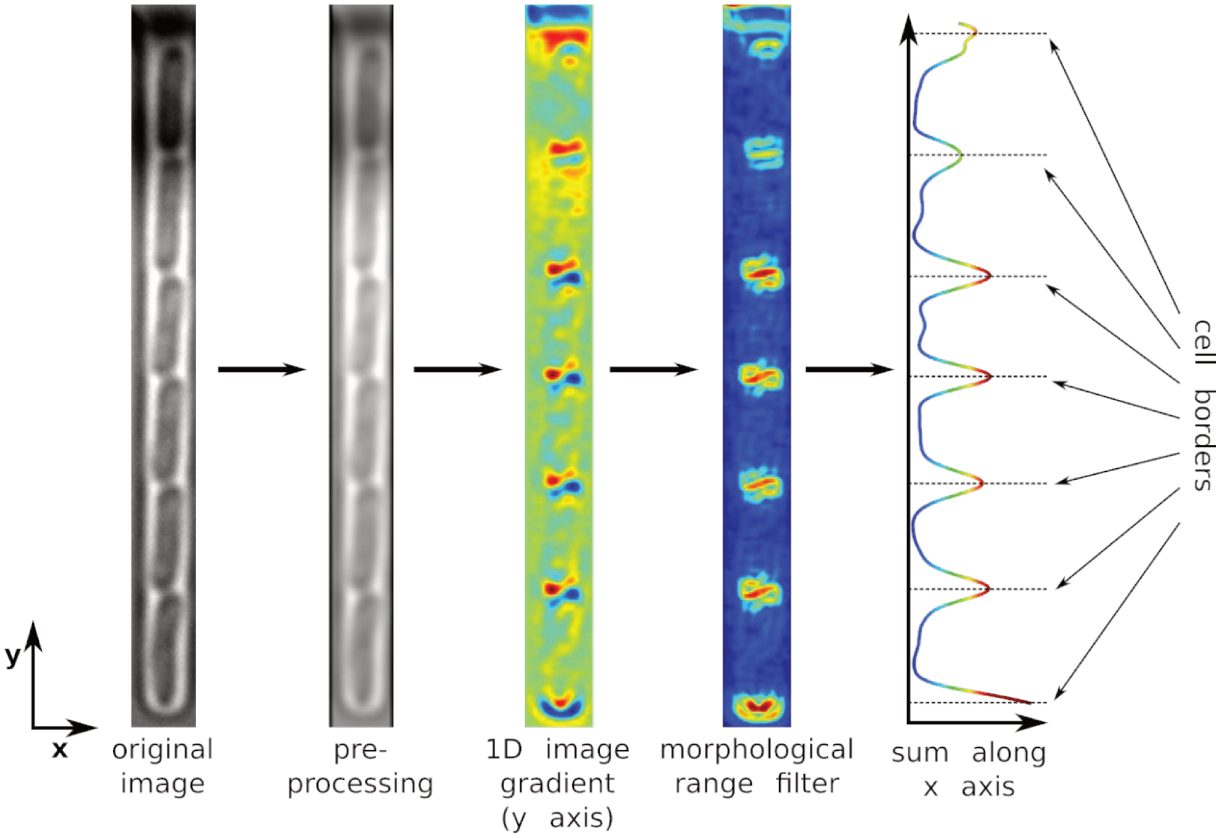

Figure 3-8 Description of the chamber segmentation algorithm (without the ellipsis fitting). The image would be pre-processed by applying a Gaussian filter, and then a 1D gradient along the y axis would be applied. Simple morphological range filtering would highlight the separations between cells in the chamber, which could easily be identified through peaks detection.

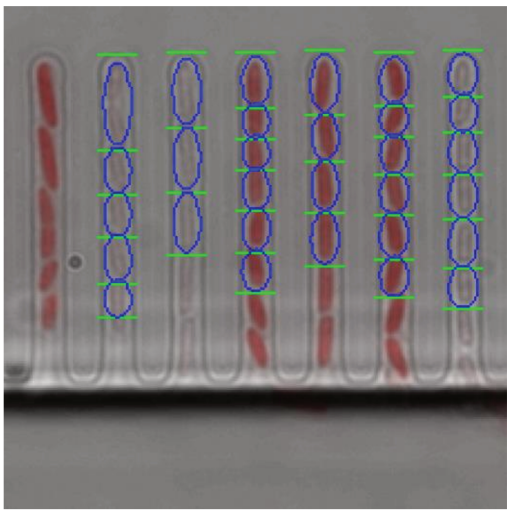

Figure 3-9 The ellipsis-fitting method: The boundaries between cells (in green) are identified (see previous figure), and ellipses are fitted to the image gradient to define a region of interest from which to extract single-cells fluorescence.

3.2.3 Mathematical morphology attempts on single layers of cells

During recent years great advances have been made in single-cell segmentation with recent developments in the field of machine learning. I discuss those results in chapter 4 of this thesis. However, during the first years of this PhD, single-cell segmentation of microscopy images was mostly built around background/foreground extraction with either image filtering techniques or mathematical morphology operations and then cell borders identification with

active contour or feature extraction techniques like the generalized Hough transform, sometimes in combination with a model of the expected cell shape and various cost minimization schemes. I tried most of the available cell segmentation programs I could find in the literature(Kamentsky et al. 2011; Chowdhury et al. 2013; Hilsenbeck et al. 2016; Q. Wang et al. 2010; Young et al. 2012; Delgado-Gonzalo & Unser 2013; Huth et al. 2011; Klein et al. 2012; Hand et al. 2009). Unfortunately, although they can perform well for offline cell segmentation, and near perfectly with a user's supervision, none of them gave satisfactory results for online segmentation and tracking.

The core problem with those methods is that they rely on a long sequence of minor operations that, though efficient and relatively robust when taken separately, amass into a complex segmentation procedure with a lot of independent tuning parameters. Each sequence in the whole operation is optimized by the user to give the best results for a reduced collection of images, and then the algorithm is applied to entire time-lapse sequences. Unfortunately the optimization of the parameters of a list of operations by the user is a very rigid approach, which doesn't generalize well to new situations and is especially detrimental to online segmentation. The thresholds, structuring elements sizes or other parameters will not be optimal for all images or even all cells in a single image, leading to segmentation inaccuracies, and accumulating tracking errors. The user/developer can try to set up rules for the program to adapt and tune its parameters to different situations, but very quickly the algorithm turns into an ever-increasing collection of exceptions and caveats and special cases, and it ends up being even less applicable to other experiments. See Figure 3-10 for an illustration of the problems faced by those algorithms.

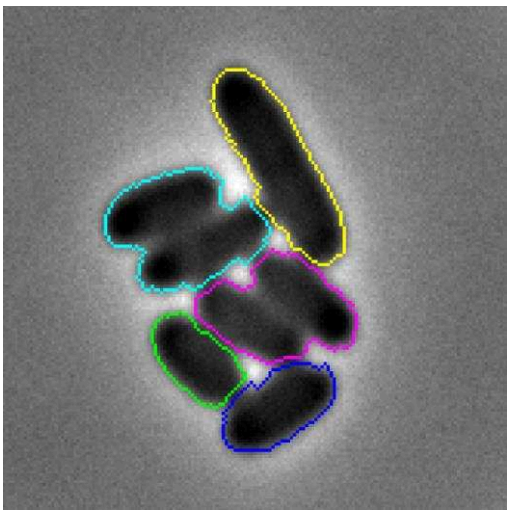

Figure 3-10 Mathematical morphology and active contour methods are often not robust enough for online segmentation. In this example, obtained with MAMLE(Chowdhury et al. 2013) but characteristic of other similar programs, 4 cells are not segmented properly. The details of the borders between cells are below the optical resolution of normal microscopes, and finding segmentation parameters that work for all cells in the image is often impossible, thus requiring user supervision.

3.3 Experiments automation

Experiment automation, although not as obviously demanding as other technical components of the real time control platform, required a substantial amount of work. While numerous microscope management software exist, few allow interfacing with custom programs. We also wanted to be able to implement complex image acquisition routines, and this dictated our implementation of our own acquisition engine from low-level microscope management function. Following the same logic, we developed the other parts in the control loop in a modular structure: Each part of the algorithm would consist of objects with standardized inputs and outputs to make the replacement of any algorithm in the loop seamless, such as a switch between control strategies (e.g. PI, Bang-bang) or between image analysis algorithms. To make the entire experimental automation robust we also divided the automation into four main independent blocks: 1) Microscope management, 2) image analysis, 3) control algorithms, and 4) microfluidic actuator supervision. These four blocks were implemented in Matlab and run separately in four different Matlab sessions that would communicate through TCP/IP. Beyond making the software more robust because the four blocks work independently and asynchronously, it also facilitates parallelization or even distribution of the entire control feedback loop over different machines: i.e. to run the image analysis or control strategy block on a separate machine for computational reasons.

All traditional microscopy brands implement their own in-house version of microscopy platform control software. However, our application requires some interfacing with the microscope software, and the microscopy brands develop closed, proprietary software that will not interface with all other brands. To access this level of flexibility we used the open-source microscope management suite Micromanager (Edelstein et al. 2014). Micromanager is an open-source, cross-platform desktop application, to control a wide variety of motorized microscopes, scientific cameras, stages, illuminators, and other microscope accessories. It supports a wide range of microscopy hardware and can also be interfaced with custom hardware, with native support for Arduino-based devices. Although it features a graphical user interface and scripting abilities, the standalone version of Micromanager was not flexible enough for our usage and could not easily be interfaced with our Matlab analysis functions. But a convenient feature of the program is the possibility to access the Core API of Micromanager, as well as some of its graphical functions, through Matlab. We developed a functional Graphical User Interface to control our microscope via Matlab, and at the same time implemented an advanced time-lapse acquisition engine, with easy interfacing with our custom scripts. It gave us the liberty to call our homemade Matlab scripts in the middle of microscopy acquisitions, and to send information to other Matlab sessions easily. We also implemented custom scripts for autofocussing or lateral drift stabilization.

The other three modules of experiment automation are mainly servers that communicate with each other to close the feedback loop. Each one of them would instantiate object implementations of the image analysis, control and actuators supervision algorithms and forward it the data received. The inputs and outputs of the object methods are standardized so that different algorithms can be used without requiring any re-writing of the main structure of

the code. For example, the image analysis algorithm described in chapter 2 was integrated to the platform with little effort once the algorithm implementation was standardized.

Although still a work in progress, this general code architecture and corresponding user interfaces are proposed as a generic implementation of any gene expression control platform, in which different core algorithms can be used interchangeably on a variety of hardware equipment. All source code as well as documentation can be found on our github page for the project[7].

With the platform assembled and automated, the cell chassis constructed and toggle switch circuits assembled, I could start to acquire characterization data and, with the knowledge gathered from it, develop a mathematical model of my system to simulate control in silico and study its response to dynamic perturbations.

[7] In progress

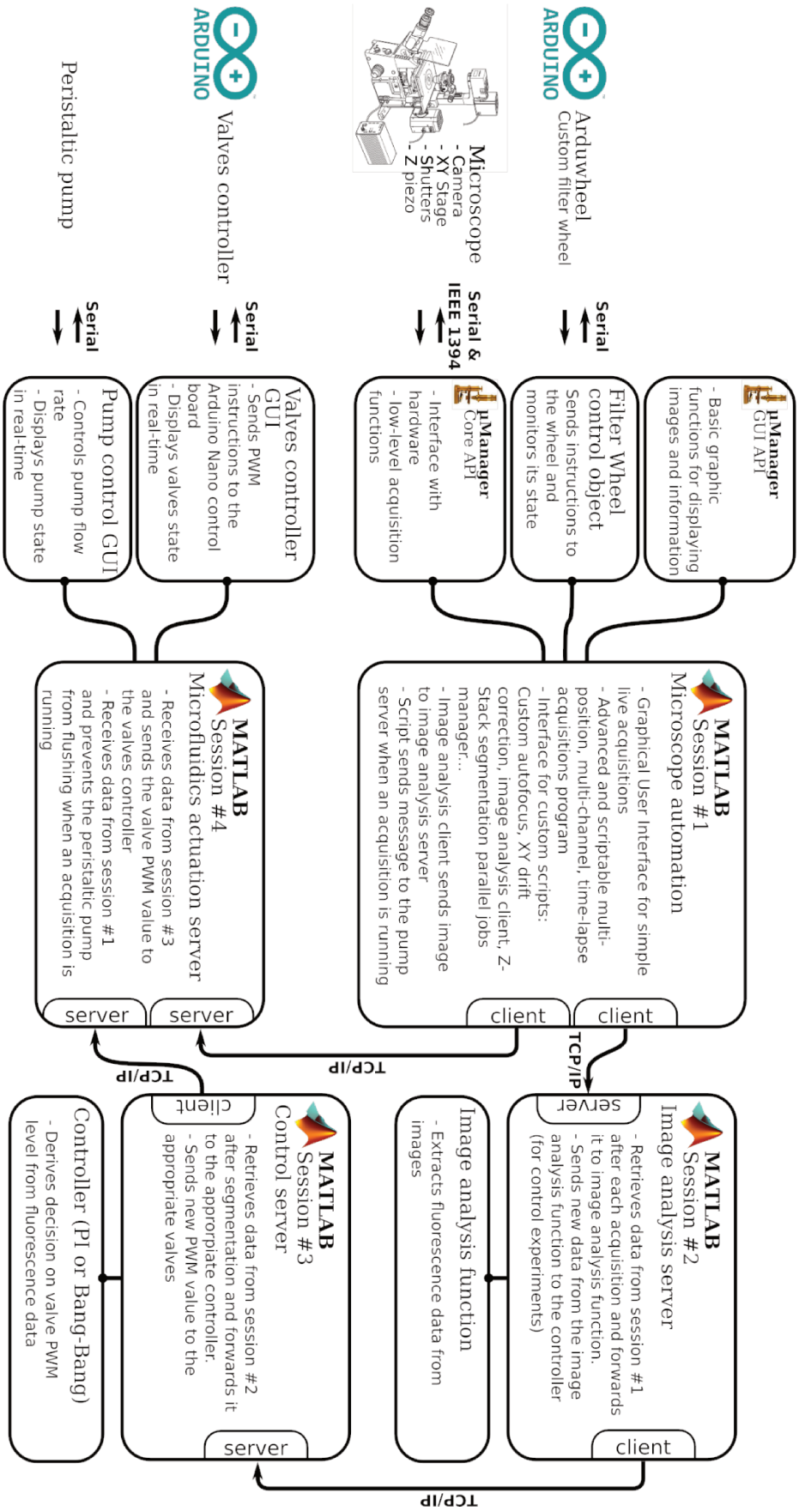

Figure 3-11 Code structure of the control platform. The main advantage of this modular structure is that different parts of the platform can be not only parallelized but also easily distributed over different machines if necessary.

4 Modeling

To understand the dynamics of our system and analyze the results of the control experiments on genetic toggle switches, we developed an in silico model of toggle switch behavior. The mathematical model of the system was fitted to experimental data and used to study theoretically the controllability of the toggle switch with various control strategies. It was then used to analyze the behavior of the toggle switch after surprising control experiments results and helped us understand its response to dynamic perturbations. We developed both deterministic and stochastic simulation approaches to study the effects of noisy gene expression on control performance.

Although models of the toggle switch exist in the literature, none of them have been fitted to data, except for the one described in Wu et al. (2013). But, their LacI-TetR toggle switch was implemented in Saccharomyces cerevisiae, and based on a completely different family of promoters. I used their study of the genetic toggle switch as a starting point for developing a model of the toggle switch. The model is presented here in its final version.

The reaction network of the genetic toggle switch and our modeling choices are described in the following paragraphs. Any gene regulatory network cannot be entirely independent from its host cell, and of course some of the modeling choices are going to appear as over-simplifications of the intricate network surrounding even this simple 2-gene synthetic circuit. Nonetheless, we believe they are sufficient to understand the main dynamics of the toggle switch and study its response to dynamic perturbations and control. We could have reduced the complexity of our model even further, as in early studies of the toggle switch (Gardner et al. 2000; Cherry & Adler 2000), but we are especially interested in the dynamics of our system, and those early models do not account for the inertial nature of the evolution of our toggle switch state, which is a key element in the evaluation of the performance of the different possible control approaches.

4.1 The reaction network

In this section I present a model of the pseudo-reactions describing the functioning of the genetic toggle switch that was used as a basis for developing our model.

4.1.1 LacI-DNA and LacI-IPTG interactions

The lac repressor LacI forms a tetramer of four identical subunits that normally binds tightly to the promoter. However, the repressor can also be only partially bound and not fully inhibit the expression of the genes downstream of the promoter. When IPTG binds to LacI, the protein changes shape and no longer can bind to the DNA. A detailed description of the DNA-LacI-IPTG interactions can be found in (Lewis et al. 1996). A list of the corresponding interactions is then[8]:

[8] Dimer versions of the lac repressor can also bind the operator site on the promoter and can also be bound by IPTG. However, for simplicity here we consider only the binding of the tetrameric version. The binding events linked to the other versions are taken into account (among other things) later on in the associated Hill

$$2\,LacI \rightleftharpoons (LacI)_2$$

$$2\,(LacI)_2 \rightleftharpoons (LacI)_4$$

$$(LacI)_4 + IPTG \rightleftharpoons \overline{(LacI)_4}$$

$$(LacI)_4 + pLac \rightleftharpoons \overline{pLac}$$

With $(LacI)_2$ and $(LacI)_4$ describing the dimeric and tetrameric versions of LacI respectively, while $\overline{(LacI)_4}$ and $\overline{pLac}$ represent the repressed states of the LacI tetramer and lac promoter respectively.

4.1.2 TetR-DNA and TetR-aTC interactions

In a similar fashion, TetR dimers assemble to bind to DNA. Although they do not twist DNA into a loop, the TetR family of repressors strongly binds to DNA through a Helix-Turn-Helix motif. Access to DNA is thus blocked for RNA polymerase binding. However, the TetR protein features a cavity in which tetracyclines can bind. In the presence of tetracyclines the TetR protein changes conformation and can no longer bind DNA. It is not clear whether two dimers can bind a tet operator. For a detailed description of those interactions see (Ramos et al. 2005). The list of interactions is[9]:

$$2\,TetR \rightleftharpoons (TetR)_2$$

$$(TetR)_2 + aTC \rightleftharpoons \overline{(TetR)_2}$$

$$(TetR)_2 + pTet \rightleftharpoons \overline{pTet}$$

With $(TetR)_2$ describing the dimeric state of TetR. $\overline{(TetR)_2}$ and $\overline{pTet}$ representing the repressed states of the TetR dimer and tet promoter respectively.

4.1.3 Transcription and translation

Transcription happens downstream of the unrepressed promoters, and proteins are then translated from the transcribed mRNAs. We do not consider here leaky expression connected to partial binding of the repressor to the toggle switch and consider that a bound dimeric/tetrameric repressor completely represses the expression of the promoter.

$$pTet \rightarrow pTet + mRNA_{LacI}$$

$$pLac \rightarrow pLac + mRNA_{TetR}$$

$$mRNA_{TetR} \rightarrow mRNA_{TetR} + TetR$$

$$mRNA_{LacI} \rightarrow mRNA_{LacI} + LacI$$

---

coefficients, which are not fixed to 4 as a strict mechanistic interpretation would have it, and are given a wide fitting range instead.

[9] Again, monomer binding to DNA and aTC is not represented. Monomer interaction as well as the potential double binding of TetR dimers to DNA is taken into account in the non-preset hill coefficient in the fitting section.

4.1.4 Dilution and degradation

Finally we model the decrease in protein and mRNA levels caused by growth dilution and degradation. In the case of protein dilution we do not model cell volume specifically, we model it instead as a normal degradation reaction for simplicity:

$$mRNA_{TetR} \rightarrow \emptyset$$

$$mRNA_{LacI} \rightarrow \emptyset$$

$$LacI \rightarrow \emptyset$$

$$TetR \rightarrow \emptyset$$

With the symbol $\emptyset$ representing nothing, or null.

4.2 ODE model

To reduce the dimensionality of our model and to reduce the number of parameters involved we model the binding-unbinding event using Hill functions. With this assumption, the ODE model was narrowed down to a 5-dimensional set of coupled ODEs:

Transcription

$$\frac{d\,mRNA_{LacI}}{dt} = \overline{k_L^m} + \frac{k_L^m}{1 + \left(\frac{TetR}{\theta_{TetR}} \times \frac{1}{1 + \left({aTC}/{\theta_{aTC}}\right)^{\eta_{aTC}}}\right)^{\eta_{TetR}}} - \delta_{mLacI} \times mRNA_{LacI}$$

$$\frac{d\,mRNA_{TetR}}{dt} = \overline{k_T^m} + \frac{k_T^m}{1 + \left(\frac{LacI}{\theta_{LacI}} \times \frac{1}{1 + \left({IPTG}/{\theta_{IPTG}}\right)^{\eta_{IPTG}}}\right)^{\eta_{LacI}}} - \delta_{mTetR} \times mRNA_{TetR}$$

The transcription rate consists in two main terms: The "basal rate" $\overline{k_{L/T}^m}$ describing leaky expression from completely repressed promoters, and the modulated expression rate describing the expression from the proportion of unrepressed promoters consisting in two nested Hill functions. The rate of unrepressed transcription is represented by the $k_{L/T}^m$ parameters, and the θ and η terms represent respectively the half occupation level and Hill coefficient for each of the binding-unbinding mechanisms. This structure causes the rate of transcription of each mRNA species to be a decreasing function of the opposing free protein, and the fraction of the free protein is also a decreasing Hill function of the associated inducer. The degradation of the two mRNAs is a simple linear function. The LacI and TetR variables in the model represent all forms of the repressors, monomers, dimers and tetramers, with and without bound inducers.

Translation

$$\frac{d\ LacI}{dt} = k_L^P \times mRNA_{LacI} - \delta \times LacI$$

$$\frac{d\ TetR}{dt} = k_T^P \times mRNA_{TetR} - \delta \times TetR$$

Here the degradation of LacI and TetR is a linear function of the protein. But because it is generally accepted that decrease in protein concentration is mostly due to dilution (Larance et al. 2013), the degradation rate is the same for both proteins.

IPTG delay

$$\frac{d\ IPTG}{dt} = k_{diff}(u_{IPTG} - IPTG)$$

$$aTC = u_{aTC}$$

We observed significant delay in the action of IPTG molecules, which we connected to the removal of the LacY permease that transported it through the membrane (see section 2.1). We added a first order equation for a simplified model of diffusion through the membrane with a constant diffusion term k_{diff}. To keep the model simple, diffusion time of aTC through the membrane was considered negligible. The u_{aTC} and u_{IPTG} variables are input signals provided by the experimenter or the control algorithm.

Matlab code implementing those equations can be found on our modeling github repository[10].

Because the mRNA dynamics and the protein dynamics evolve different timescales, we used a stiff ODE solver (ode23s in Matlab) to integrate the equations in between changes in the driving u_{aTC} and u_{IPTG} variables. In in silico control experiements, the control algorithm is ran outside of the model evaluation algorithm for the sake of implementation simplicity and modularity. Therefore our general implementation considers the ODE set to be a switched nonlinear system and features a custom solver for such systems built around the ode23s solver[11].

4.3 Gillespie's stochastic simulation algorithm

Because noise can greatly affect the quality of control, I also developed a stochastic model based on the pseudo-reactions described in section 4.1[12]. This interpretation is based on a continuous-time Markov chain model of the process and can be solved using Gillespie's Stochastic Simulation Algorithm (SSA)(Gillespie 1977). The Gillespie algorithm is an exact simulation of the stochastic behavior of a process. However, because it was designed to simulate random encounters between relevant molecules due to Brownian motion, it requires simulating every single reaction between two or less molecules/complexes. In our case, we

[10] github.com/Lab513/ToggleSwitch/blob/reOrg/CoreFunctions/CoreModel/toggle_derivative_sim.m
[11] github.com/Lab513/ToggleSwitch/blob/reOrg/CoreFunctions/generate_data.m
[12] Because the ODEs of the deterministic model and the propensities of this model are based on the same network of pseudo-reactions, and because they use the same rates and assumptions, I could use the same parameter values in both approaches.

kept the assumption on the Hill functions done in the ODE model of the reactions network for two reasons. Firstly, parameters were fitted to the Hill-based ODE model, and it would not have been possible to deconstruct the model back into equations where all binding and unbinding events are simulated. Even if a deconstructed model had been fitted to the characterization data, some of the parameters could not have been identified because of current limits on protein data resolution, and to my knowledge state-of-the-art in vivo proteins levels measurement is still a long way from reaching the necessary precision level. The second reason is that binding-unbinding events happen orders of magnitude more often than other events, which led us to the Hill functions in the first place, and simulating those events with Gillespie's SSA would severely increase computational time. Applying this approximation to the SSA in the case of chemical kinetic systems with disparate rates is a common workaround to this issue, and in most cases does not have a major impact on the simulated levels of noise (Rao & Arkin 2003). The pseudo-reactions modeled by our SSA implementation and their propensities are described in Table 4.3-1. The implementation of the stochastic model can be found in our github repository[13].

	Reaction	Propensity
transcription	$\varnothing \rightarrow mRNA_{TetR}$	$\overline{k_T^m} + \frac{k_T^m}{1 + \left(\frac{LacI}{\theta_{LacI}} \times \frac{1}{1 + \left({IPTG}/{\theta_{IPTG}}\right)^{\eta_{IPTG}}}\right)^{\eta_{LacI}}}$
	$\varnothing \rightarrow mRNA_{LacI}$	$\overline{k_L^m} + \frac{k_L^m}{1 + \left(\frac{TetR}{\theta_{TetR}} \times \frac{1}{1 + \left({aTC}/{\theta_{aTC}}\right)^{\eta_{aTC}}}\right)^{\eta_{TetR}}}$
translation	$\varnothing \rightarrow TetR$	$k_T^P \times mRNA_{TetR}$
	$\varnothing \rightarrow LacI$	$k_L^P \times mRNA_{LacI}$
degradation	$mRNA_{TetR} \rightarrow \varnothing$	$\delta_{mTetR} \times mRNA_{TetR}$
	$mRNA_{LacI} \rightarrow \varnothing$	$\delta_{mLacI} \times mRNA_{LacI}$

[13] github.com/Lab513/ToggleSwitch/blob/reOrg/CoreFunctions/CoreModel/toggle_props_stoich.m

dilution	$TetR \rightarrow \emptyset$	$\delta \times TetR$
	$LacI \rightarrow \emptyset$	$\delta \times LacI$

Table 4.3-1 Pseudo-reactions and propensities used in the Stochastic Simulation Algorithm

IPTG diffusion was considered deterministic and the IPTG concentration dynamics were integrated outside of the SSA function through Matlab's ode23s solver.

4.4 Parameter identification

To obtain calibration data, we performed 6 experiments differing by the initial state of the cell population and by the temporal profiles of the inducers and probing in various ways the dynamics of the system (see section 3.1 of chapter 3). Model fitting was made using the global optimization tool CMA-ES (Hansen & Ostermeier 1996). We used the measured fluorescence values for RFP and GFP directly as proxies for measured molecule numbers per cell, which means that the transcription, translation, degradation and dilution rates as well as Hill half occupation levels are fitted to fluorescence levels. As a first approximation, we assume the measured fluorescence levels per cell to be proportional to molecule numbers. The objective for CMA-ES was to minimize the mean squared relative deviations between model predictions and averaged measured fluorescence.

The optimization was repeated 8 times. Despite the use of a simple model not all parameters were fully constrained by the data, however all estimations produced similar dynamics and similar equilibrium points could be inferred from the different sets of parameters. We picked the set of parameter that produced the unstable equilibrium point with the highest levels of LacI and TetR to maximize the signal-to-noise ratio during our control experiments.

The final parameter values obtained were:

Transcription rates	k_T^m	1.607	molec./min
	k_L^m	1.009	
	$\overline{k_T^m}$	0.0238	
	$\overline{k_L^m}$	0.1715	
Translation rates	k_T^P	7.268	
	k_L^P	1.906	
Degradation rates	δ_{mTetR}	0.1477	
	δ_{mLacI}	0.1386	
Dilution rate	δ	0.0199	

Hill coefficients	η_{TetR}	2.093
	η_{LacI}	2.000
	η_{aTC}	2.000
	η_{IPTG}	2.000
Half occupation levels	θ_{TetR}	123.0 molec.
	θ_{LacI}	226.8 molec.
	θ_{aTC}	38.9 ng/mL
	θ_{IPTG}	0.300 mM

Table 4.4-1 Parameter values after CMA-ES identification after fitting the model to characterization data.

4.5 Controllers

The main controllers used in this study are fairly simple although we did use model predictive control algorithms in early theoretical studies of toggle switch controllability. We used the same algorithms for in vivo control and in silico simulations.

4.5.1 Bang-bang control

The so-called bang-bang controller, also known as a hysteresis controller, is a feedback controller that switches abruptly between two states. Although bang–bang controllers can produce optimal control strategies in some cases, they are often implemented because of simplicity or convenience. The input delivery technique can also restrict actuation to a binary all-or-nothing choice in some microfluidic devices. However in our case, the use of a bang-bang controller was dictated by early theoretical results that indicated that PI controllers with a strong proportional term would perform better on a population of cells (see chapter 3). This incited us to try the extreme case of bang-bang control:

$$u_{aTC} = \begin{cases} u_{aTC}^{max}, \ if\ LacI(t) \leq LacI^* \\ u_{aTC}^{min}, \ if\ LacI(t) > LacI^* \end{cases}, \quad \text{and} \quad u_{IPTG} = \begin{cases} u_{IPTG}^{max}, \ if\ TetR(t) \leq TetR^* \\ u_{IPTG}^{min}, \ if\ TetR(t) > TetR^* \end{cases}$$

With $LacI^*$ and $TetR^*$ being the setpoint objectives chosen for control, and $u_{aTC/IPTG}^{min}$ and $u_{aTC/IPTG}^{max}$ as parameters indicating the minimum and maximum values reachable by the input device (Here the maximally closed/open states of the valves).

Code implementation of this control algorithm that was used both in silico and on real control experiments can be found in our github repository[14].

4.5.2 Proportional-integral control

A Proportional-Integral (PI) controller continuously calculates an error value that is the difference between a desired setpoint and the measured level of fluorescence. The controller attempts to minimize the error over time by adjusting the level of inducer. But contrary to the bang-bang controller, the PI controller does not completely open or close the valves, but it can take advantage of the pulse-width modulation implemented with the 3-way valves (see section 3.1.6) and provide intermediary concentrations of inducers to the cells. It also takes into account past errors in an integral term to correct offset errors over time.

$$u_{aTC} = \begin{cases} u_{aTC}^{max}, & if\ PI_{aTC}(t) + u_{aTC}^0 \geq u_{aTC}^{max} \\ PI_{aTC}(t) + u_{aTC}^0, & if\ u_{aTC}^{max} > PI_{aTC}(t) + u_{aTC}^0 > u_{aTC}^{min} \\ u_{aTC}^{min}, & if\ PI_{aTC}(t) + u_{aTC}^0 \leq u_{aTC}^{min} \end{cases}$$

With $PI_{aTC}(t) = \begin{cases} K_p^L\big(LacI^* - LacI(t)\big), & if\ t < t_d \\ K_p^L\big(LacI^* - LacI(t)\big) + K_i^L \int_{t_d}^{t} \big(LacI^* - LacI(s)\big)ds, & if\ t \geq t_d \end{cases}$

[14] github.com/Lab513/ToggleSwitch/blob/reOrg/CoreFunctions/Control/BBcontrollerNew.m

$$u_{IPTG} = \begin{cases} u_{IPTG}^{max}, & if\ PI_{IPTG}(t) + u_{IPTG}^{0} \geq u_{IPTG}^{max} \\ PI_{IPTG}(t) + u_{IPTG}^{0}, & if\ u_{IPTG}^{max} > PI_{IPTG}(t) + u_{IPTG}^{0} > u_{IPTG}^{min} \\ u_{IPTG}^{min}, & if\ PI_{IPTG}(t) + u_{IPTG}^{0} \leq u_{IPTG}^{min} \end{cases}$$

With $PI_{IPTG}(t) = \begin{cases} K_p^T\left(TetR^* - TetR(t)\right), & if\ t < t_d \\ K_p^T\left(TetR^* - TetR(t)\right) + K_i^T \int_{t_d}^{t}\left(TetR^* - TetR(s)\right)ds, & if\ t \geq t_d \end{cases}$

With the $K_p^{L/T}$ and $K_i^{L/T}$ as the proportional and integral parameters respectively, $u_{aTC/IPTG}^{0}$ the reference levels of aTC or IPTG obtained experimentally (20 ng/mL of aTC and 0.25 mM of IPTG for all toggle switch control experiments), and t_d the time delay before the integral term is applied. The last element, the time delay, was added later in the experimental process to avoid overshoot effects caused by the error just after the objective is applied before the system can be in the vicinity of the objective, which typically takes about two hours.

Code implementing this control algorithm and that was used both in silico and on real control experiments can be found in our github repository[15].

5 Conclusion

In this chapter we presented the materials and methods used in this thesis to control a genetic toggle switch. In the first part of this chapter, the biological side of the problem was presented, with details on possible implementation choices that did not work to expose the limits of the approach as well as some of the possible ways of improvements over the current implementation. For example, while the fusion protein design was not adapted to our problem, in other problems it should be possible to use this type of design for sensing the level of a protein of interest. Another important contribution is the development of a chassis strain for IPTG- and aTC- based induction that can be used for other control applications. To the best of my knowledge, no bacterial strain was ever optimized for simultaneous induction of the lac and tet systems. Finally, as a general remark on biological systems development, the construction of a chassis and circuit is a time-consuming process in the workflow of external cell control that should not be overlooked when undertaking such a task.

In the second part of the chapter, hardware and software choices are discussed. In particular, a 3-way pulse-width-modulation mixing apparatus is described in what is, to the best of my knowledge, the first attempt at double chemical induction in microfluidic devices. Variants of the mother-machine microfluidic device are also presented to facilitate long-term acquisition and image analysis. This setup allows us to robustly extract single-cell fluorescence data and control single-cells over extended periods of time. The efforts put into the development, automation and integration of the different parts of the platform can be re-invested in single cell control of other genetic circuits, with the possibility to rapidly modify different parts of the platform thanks to its modular nature. One important aspect of the methodology of

[15] github.com/Lab513/ToggleSwitch/blob/reOrg/CoreFunctions/Control/PIcontrollerNew.m

external control of gene expression that will make reproducibility possible is opening the design of different elements of the platform, and I have already released parts of the designs presented here, and will release the rest of the platform under open-source and open-hardware licenses.

In the third part of this chapter, we discussed the elaboration of an ODE model and an SSA model of the genetic toggle switch to perform deterministic and stochastic simulations of the genetic toggle switch. In the next chapter we use this model to evaluate the possibility to control the toggle switch with different control strategies. Throughout the process of characterization we observed delays in the system that we incorporated into the model. We also discuss fitting of the model to characterization data, and our choices in implementation of the control strategies. A number of improvements could be added to the control strategies, as the two control strategies presented here have been studied extensively and a number of improvements on those techniques as well as other techniques have been presented in the literature.

One of the main domains of application of this in silico feedback platform would be automated characterization of genetic circuits in bacteria, synthetic or endogenous. The control algorithm and the induction techniques presented here are generic enough that the method could be applied to various genetic networks with minor modifications.

Anderson, J.C. et al., 2010. BglBricks: A flexible standard for biological part assembly. *Journal of Biological Engineering*, 4(1), p.1.

Baba, T. et al., 2006. Construction of Escherichia coli K-12 in-frame, single-gene knockout mutants: the Keio collection. *Molecular systems biology*, 2, p.2006.0008.

Chang, A.C. & Cohen, S.N., 1978. Construction and characterization of amplifiable multicopy DNA cloning vehicles derived from the P15A cryptic miniplasmid. *Journal of bacteriology*, 134(3), pp.1141–56.

Chen, H. et al., 2015. Genome-wide study of mRNA degradation and transcript elongation in Escherichia coli. *Molecular systems biology*, 11, p.781.

Chen, X., Zaro, J.L. & Shen, W.-C., 2013. Fusion protein linkers: property, design and functionality. *Advanced drug delivery reviews*, 65(10), pp.1357–69.

Cherry, J.L. & Adler, F.R., 2000. How to make a Biological Switch. *Journal of Theoretical Biology*, 203(2), pp.117–133.

Chowdhury, S. et al., 2013. Cell segmentation by multi-resolution analysis and maximum likelihood estimation (MAMLE). *BMC bioinformatics*, 14 Suppl 1(Suppl 10), p.S8.

Dailey, F.E. & Berg, H.C., 1993. Mutants in disulfide bond formation that disrupt flagellar assembly in Escherichia coli. *Proceedings of the National Academy of Sciences*, 90(3), pp.1043–1047.

Datsenko, K. a & Wanner, B.L., 2000. One-step inactivation of chromosomal genes in Escherichia coli K-12 using PCR products. *Proceedings of the National Academy of Sciences of the United States of America*, 97(12), pp.6640–5.

Delgado-Gonzalo, R. & Unser, M., 2013. Spline-based framework for interactive segmentation in biomedical imaging. *IRBM*, 34(3), pp.235–243.

Dewar, H. et al., 2004. Tension between two kinetochores suf ces for their bi-orientation on the mitotic spindle. *Nature*, 428, pp.93–97.

Edelstein, A.D. et al., 2014. Advanced methods of microscope control using μManager software. *Journal of Biological Methods*, 1(2), p.10.

Engler, C. et al., 2009. Golden gate shuffling: A one-pot DNA shuffling method based on type ils restriction enzymes. *PLoS ONE*, 4(5).

Engler, C., Kandzia, R. & Marillonnet, S., 2008. A one pot, one step, precision cloning method with high throughput capability. *PLoS ONE*, 3(11).

Engler, C. & Marillonnet, S., 2011. Generation of Families of Construct Variants Using Golden Gate Shuffling. In C. Lu, J. Browse, & J. G. Wallis, eds. *Methods*. Methods in Molecular Biology. Totowa, NJ: Humana Press, pp. 167–181.

Espah Borujeni, A., Channarasappa, A.S. & Salis, H.M., 2014. Translation rate is controlled by coupled trade-offs between site accessibility, selective RNA unfolding and sliding at upstream standby sites. *Nucleic Acids Research*, 42(4), pp.2646–2659.

Gardner, T.S., Cantor, C.R. & Collins, J.J., 2000. Construction of a genetic toggle switch in Escherichia coli. *Nature*, 403(6767), pp.339–42.

Gillespie, D.T., 1977. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25), pp.2340–2361.

Govantes, F., Andújar, E. & Santero, E., 1998. Mechanism of translational coupling in the nifLA operon of Klebsiella pneumoniae. *EMBO Journal*, 17(8), pp.2368–2377.

Hand, A.J. et al., 2009. Automated tracking of migrating cells in phase-contrast video microscopy sequences using image registration. *Journal of Microscopy*, 234(1), pp.62–79.

Hansen, N. & Ostermeier, A., 1996. Adapting arbitrary normal mutation distributions in evolution strategies: the covariance matrix adaptation. In *Proceedings of IEEE International Conference on Evolutionary Computation*. IEEE, pp. 312–317.

Hilsenbeck, O. et al., 2016. Software tools for single-cell tracking and quantification of cellular and molecular properties. *Nature biotechnology*, 34(7), pp.703–6.

Huth, J. et al., 2011. TimeLapseAnalyzer: Multi-target analysis for live-cell imaging and time-lapse microscopy. *Computer Methods and Programs in Biomedicine*, 104(2), pp.227–234.

Ikebe, T., Iyoda, S. & Kutsukake, K., 1999. Structure and expression of the fliA operon of Salmonella typhimurium. *Microbiology*, 145(6), pp.1389–1396.

Kamentsky, L. et al., 2011. Improved structure, function and compatibility for CellProfiler: modular high-throughput image analysis software. *Bioinformatics*, 27(8), pp.1179–80.

Kato, N. & Lam, E., 2003. Chromatin of endoreduplicated pavement cells has greater range of movement than that of diploid guard cells in Arabidopsis thaliana. *Journal of cell science*, 116(Pt 11), pp.2195–201.

Klein, J. et al., 2012. TLM-Tracker: software for cell segmentation, tracking and lineage analysis in time-lapse microscopy movies. *Bioinformatics*, 28(17), pp.2276–2277.

Knight, T., 2003. *Idempotent Vector Design for Standard Assembly of Biobricks*, Boston, MA.

Kudryashov, V. et al., 2003. Grey scale structures formation in SU-8 with e-beam and UV. *Microelectronic Engineering*, 67–68, pp.306–311.

Kuntz, J., Oyarzún, D. & Stan, G.-B., 2014. Model Reduction of Genetic-Metabolic Networks via Time Scale Separation. In *A Systems Theoretic Approach to Systems and Synthetic Biology I: Models and System Characterizations*. Dordrecht: Springer Netherlands, pp. 181–210.

Larance, M. et al., 2013. Global subcellular characterization of protein degradation using quantitative proteomics. *Molecular & cellular proteomics : MCP*, 12(3), pp.638–50.

Le, T.T., Guet, C.C. & Cluzel, P., 2006. Protein expression enhancement in efflux-deleted mutant bacteria. *Protein expression and purification*, 48(1), pp.28–31.

Lee, C.-Y. et al., 2011. Microfluidic Mixing: A Review. *International Journal of Molecular Sciences*, 12(12), pp.3263–3287.

Lewis, M. et al., 1996. Crystal structure of the lactose operon repressor and its complexes with DNA and inducer. *Science (New York, N.Y.)*, 271(5253), pp.1247–54.

Lin-Chao, S., Chen, W.-T. & Wong, T.-T., 1992. High copy number of the pUC plasmid results from a Rom/Rop-suppressible point mutation in RNA II. *Molecular Microbiology*, 6(22), pp.3385–3393.

Llamosi, A. et al., 2016. What Population Reveals about Individual Cell Identity: Single-Cell Parameter Estimation of Models of Gene Expression in Yeast J. Stelling, ed. *PLOS Computational Biology*, 12(2), p.e1004706.

Lutz, R. & Bujard, H., 1997. Independent and tight regulation of transcriptional units in Escherichia coli via the LacR/O, the TetR/O and AraC/I1-I2 regulatory elements. *Nucleic acids research*, 25(6), pp.1203–10.

Ma, D. et al., 1995. Genes acrA and acrB encode a stress-induced efflux system of Escherichia coli. *Molecular Microbiology*, 16(1), pp.45–55.

Maizels, N., 1974. E. coli lactose operon ribosome binding site. *Nature*, 249(5458), pp.647–649.

Mali, P., Sarkar, A. & Lal, R., 2006. Facile fabrication of microfluidic systems using electron beam lithography. *Lab Chip*, 6(2), pp.310–315.

Marbach, A. & Bettenbrock, K., 2012. Lac operon induction in Escherichia coli: Systematic comparison of IPTG and TMG induction and influence of the transacetylase LacA. *Journal of Biotechnology*, 157(1), pp.82–88.

Marshall, W.F. et al., 1997. Interphase chromosomes undergo constrained diffusional motion in living cells. *Current biology : CB*, 7(12), pp.930–939.

McCarter, L.L., 2006. Regulation of flagella. *Current Opinion in Microbiology*, 9(2), pp.180–186.

Miller, J.H., 1992. *A short course in bacterial genetics: a laboratory manual and handbook for Escherichia coli and related bacteria.*, CSHL Press.

Ng, J.M.K. et al., 2002. Components for integrated poly(dimethylsiloxane) microfluidic systems. *ELECTROPHORESIS*, 23(20), pp.3461–3473.

Oppenheim, D.S. & Yanofsky, C., 1980. Translational coupling during expression of the tryptophan operon of Escherichia coli. *Genetics*, 95(4), pp.785–795.

Ozbudak, E.M. et al., 2004. Multistability in the lactose utilization network of Escherichia coli. *Nature*, 427(6976), pp.737–40.

Ozbudak, E.M. et al., 2002. Regulation of noise in the expression of a single gene. *Nature Genetics*, 31(1), pp.69–73.

Ramos, J.L. et al., 2005. The TetR Family of Transcriptional Repressors. *Microbiology and Molecular Biology Reviews*, 69(2), pp.326–356.

Rao, C. V. & Arkin, A.P., 2003. Stochastic chemical kinetics and the quasi-steady-state assumption: Application to the Gillespie algorithm. *The Journal of Chemical Physics*, 118(11), p.4999.

Rasmussen, B. et al., 1991. Molecular basis of tetracycline action: identification of analogs whose primary target is not the bacterial ribosome. *Antimicrobial Agents and Chemotherapy*, 35(11), pp.2306–2311.

Rebatchouk, D., Daraselia, N. & Narita, J.O., 1996. NOMAD: a versatile strategy for in vitro DNA manipulation applied to promoter analysis and vector design. *Proceedings of the National Academy of Sciences*, 93(20), pp.10891–10896.

Rosenfeld, N., Elowitz, M.B. & Alon, U., 2002. Negative Autoregulation Speeds the Response Times of Transcription Networks. *Journal of Molecular Biology*, 323(5), pp.785–793.

Sambrook, J. & Russell, D.W., 2003. *Molecular cloning : a laboratory manual* 2nd ed., Cold Spring Harbor Laboratory.

Santillán, M., Mackey, M.C. & Zeron, E.S., 2007. Origin of bistability in the lac Operon. *Biophysical journal*, 92(11), pp.3830–42.

Schaefer, E.M. et al., 1989. Ribosome-binding sites and RNA-processing sites in the transcript of the Escherichia coli unc operon. *Journal of Bacteriology*, 171(7), pp.3901–3908.

Suñé, G.R., 2008. *Electron beam lithography for Nanofabrication*. Universitat Autònoma de Barcelona.

Torgov, M.Y., Janzen, D.M. & Reddy, M.K., 1998. Efficiency and frequency of translational coupling between the bacteriophage T4 clamp loader genes. *Journal of bacteriology*, 180(17), pp.4339–43.

Trötschel, C., Albaum, S.P. & Poetsch, A., 2013. Proteome turnover in bacteria: Current status for Corynebacterium glutamicum and related bacteria. *Microbial Biotechnology*, 6(6), pp.708–719.

Tseng, A.A. et al., 2003. Electron beam lithography in nanoscale fabrication: Recent development. *IEEE Transactions on Electronics Packaging Manufacturing*, 26(2), pp.141–149.

Uhlendorf, J. et al., 2012. Long-term model predictive control of gene expression at the population and single-cell levels. *Proceedings of the National Academy of Sciences of the United States of America*, 109(35), pp.14271–6.

Veltkamp, E. et al., 1979. Replication and gene functions of the bacteriocinogenic plasmid CloDF13. *Contributions to microbiology and immunology*, 6, pp.111–21.

Vulin, C., 2014. *A quantitative approach to microbial population growth using tailored cylindrical yeast colonies*. Paris 7 Diderot.

Wang, P. et al., 2010. Robust growth of Escherichia coli. *Current biology*, 20(12), pp.1099–103.

Wang, Q. et al., 2010. Image segmentation and dynamic lineage analysis in single-cell fluorescence microscopy. *Cytometry*, 77(1), pp.101–10.

Webb, C.D. et al., 1997. Bipolar localization of the replication origin regions of chromosomes in vegetative and sporulating cells of B. subtilis. *Cell*, 88(5), pp.667–74.

Weber, E. et al., 2011. A modular cloning system for standardized assembly of multigene constructs. *PloS one*, 6(2), p.e16765.

Wilkinson, D. a et al., 2011. Regulation of flagellum number by FliA and FlgM and role in biofilm formation by Rhodobacter sphaeroides. *Journal of bacteriology*, 193(15), pp.4010–4.

Wu, M. et al., 2013. Engineering of regulated stochastic cell fate determination. *Proceedings of the National Academy of Sciences*, 110(26), pp.10610–10615.

Yanisch-Perron, C., Vieira, J. & Messing, J., 1985. Improved M13 phage cloning vectors and host strains: nucleotide sequences of the M13mp18 and pUC19 vectors. *Gene*, 33(1), pp.103–19.

Young, J.W. et al., 2012. Measuring single-cell gene expression dynamics in bacteria using fluorescence time-lapse microscopy. *Nature protocols*, 7(1), pp.80–8.

Zgurskaya, H.I. & Nikaido, H., 1999. AcrA is a highly asymmetric protein capable of spanning the periplasm. *Journal of Molecular Biology*, 285(1), pp.409–420.

Chapter III
Toggle switch stabilization

Table of Contents

1 Introduction 94
2 Characterization experiments 94
3 Fitted model 95
3.1 Fitting results 95
3.2 State space analysis 97
3.3 Stable states and bifurcations 99
4 Single-cell toggle switch control 102
4.1 PI control 103
4.1.1 In silico control 104
4.1.2 In vivo control 104
4.2 Bang-bang control 107
4.2.1 Model 107
4.2.2 Experiments 107
5 Open-loop dynamic stabilization 108
6 Conclusion 111

1 Introduction

With the 2-inputs-2-outputs feedback platform presented in the previous chapter, we investigated the possibility of controlling the genetic toggle switch. The toggle switch system is a fundamental element in numerous natural and synthetic gene regulation networks. In this chapter I am going to present dynamical control attempts, and successes, at stabilizing single toggle-switch-bearing cells in the unstable area of this double-negative feedback system. I start by presenting the characterization experiments used for fitting the model of our system, and then present a study of the stability of the fitted model for different values of the inducer concentrations. The main in silico and in vivo results of this thesis are then presented, demonstrating the possibility to control the bistable toggle switch in real-time in the vicinity of the unstable point. Finally, surprising results for population control of toggle-switch-bearing cells are analyzed and a new regime of stability for the genetic toggle switch is demonstrated.

2 Characterization experiments

In order to gather some knowledge on the system as well as identify parameters for the toggle switch model, the selected toggle switch circuit (see section 1 of chapter 2) was submitted to a series of characterization experiments. In the first series of experiments the cells were submitted to long-term exposure to either of the two inducers IPTG or aTC to observe basic dynamic switch responses. A surprising result contrasting with the overnight experiments in batch cultures was that the toggle switch, in absence of inducers, would not remain in its LacI-RFP-dominant state, and would instead drift towards a TetR-GFP-dominant state, even when switched in the opposite state initially. Less surprisingly, this TetR-GFP-dominant state would produce a lower level of green fluorescence than fully activated. Although models predict that this monostable behavior in the absence of inducers is possible, it is relatively surprising that microfluidics experiments provoke a behavior that differs from the batch culture ones. It should be noted however that cells in mother machine devices are submitted to a constant flow of nutrients, which both washes the inducers away continuously and efficiently after they are removed, and puts the cells in a continuous exponential growth phase since they never lack nutrients. Microfluidic device culture conditions and batch culture conditions differ significantly.

Theory suggests that, when a toggle switch can be switched between its too stable states, there exists a concentration of base levels of aTC and IPTG that should transform the landscape of the toggle switch into a so-called “balanced” landscape where the system is bistable and the basins of attraction are symmetrically separated. We empirically obtained concentrations of aTC and IPTG, subsequently called reference concentrations, for which both states were stable in microfluidic experiments. If, when ‘releasing’ the system, we switched to those reference concentrations of aTC and IPTG instead of no inducers, the system would stay in whichever state (LacI-RFP-high/TetR-GFP-low or LacI-RFP-low/TetR-GFP-high) it was before the release. These concentrations for which our cells are bistable are aTC = 20 ng/mL and IPTG = 0.25 mM.

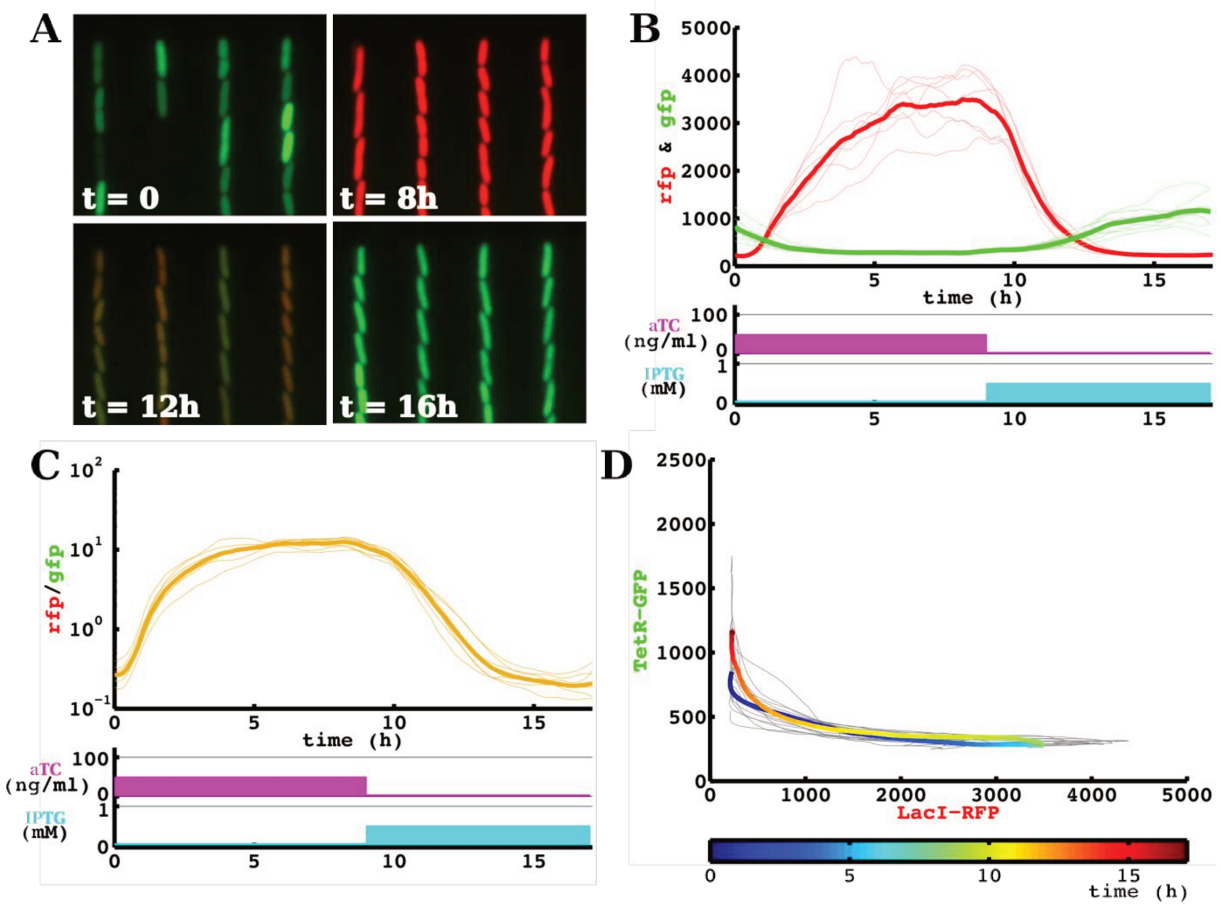

Figure 2-1 Switching experiment. The cells, initially in the TetR-GFP-high state, are submitted to 50ng/mL of aTc for 9 hours and then switched to 0.5mM of IPTG for ~8 hours. A) The cells switch between an RFP-high/GFP-low and an RFP-low/GFP-high state over time. B) The measured fluorescence levels for the two fluorescent proteins are represented over time. Thin lines represent individual cell behaviors, and the thicker lines represent the average fluorescence of the population. Below the fluorescence curves, the mixed concentrations of aTc and IPTG over time are shown. C) The ratio of the two fluorescence levels is shown. Although this representation removes some of the information it shows the correspondence between the RFP and GFP levels of individual cells better than the previous panel. D) The trajectory of individual cells through the protein space [TetR-GFP, LacI-RFP] are shown in gray. The mean trajectory of the population is color-coded in time from blue to red. The color bar below the graph indicates the correspondence between time and color. Taken together, all those representations help better understand the complex data resulting from our single-cell timelapse experiments.

3 Fitted model

The model described in section 3 of chapter 2 was fitted to the characterization data presented in Figure 3-1. From this dynamical model we could study the behavior of the toggle switch under a feedback loop, and also its behavior under static or dynamic inducer stimulations, but we also needed the model to localize the unstable equilibrium in the protein space and use these levels as a reference for our setpoint control objectives.

3.1 Fitting results

The fitting was performed using CMA-ES on a set of characterization experiments presented in Figure 3-1. To test how the fit of the parameters was constrained by the data, we ran several instances of CMA-ES in parallel on the same dataset and obtained different fitted parameters sets. Ideally instances of CMA-ES would converge to the same parameter values. Although a similar fit to the data was obtained with all parameter sets, some parameters had

significantly different values in the different parameter sets. The estimations were not completely constrained by the data, which is a common issue in parameter fitting for biochemical models (Ashyraliyev et al. 2009; Villaverde & Banga 2013). However the localization of the equilibrium points, stable and unstable, was well constrained by the data and, for all parameters sets, the unstable equilibrium points in reference conditions were close to each other. As a reference for our control experiments, we used the unstable equilibrium point that had the highest level of both proteins at LacI-RFP ≈ 750 and TetR-GFP ≈ 320. We made this choice because the levels of the estimated unstable states are relatively low compared to their dynamic range, and at those levels the signal gets noisy.

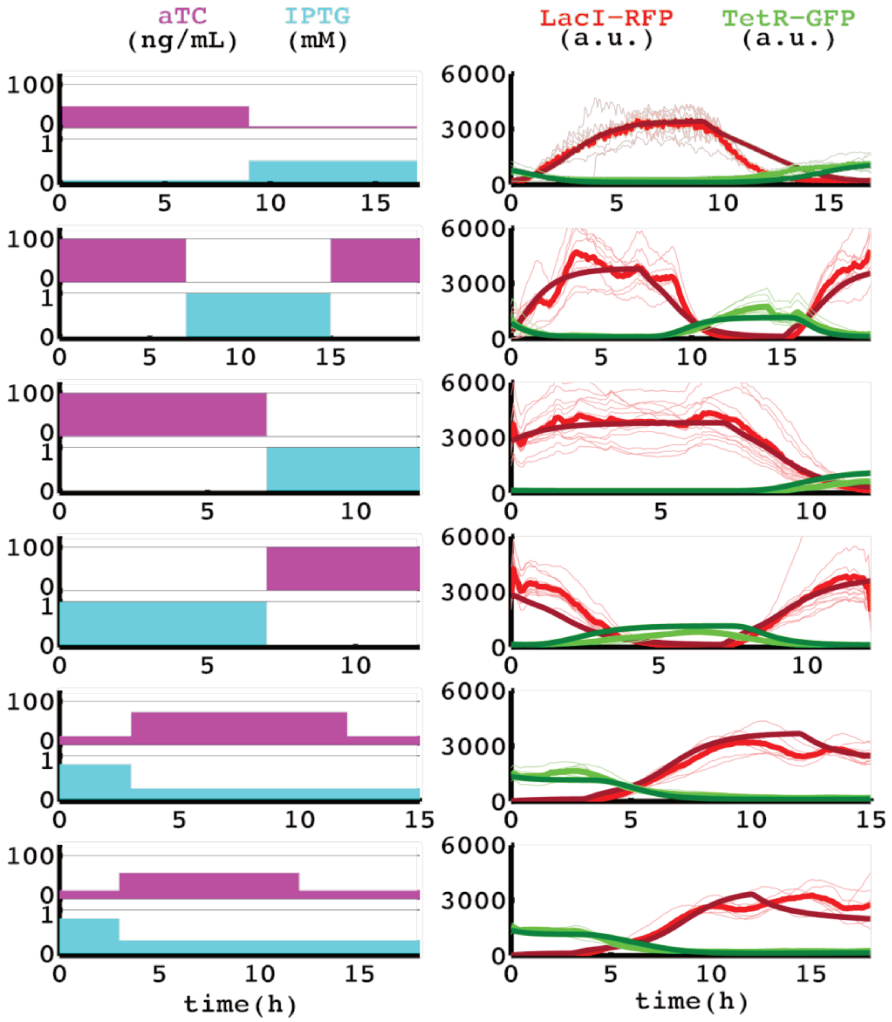

Figure 3-1 Characterization experiments that were used to fit the toggle switch model. The concentration profiles of each inducer on the left produced the experimental data on the right. The thin, light red/green curves represent the individual cell trajectories. The thick red/green curves are the population averages to which the model is fitted. The thick, dark red/green curves represent the deterministic simulation of the response to the inducer inputs from the fitted model.

Although the predictions extracted from the fitted model presented here are satisfactory, as illustrated in the following sections and in Figure 3-1, they should be interpreted mostly qualitatively for several reasons: first of all, the study of noise in genetic expression tells us that a significant part of it is caused by so-called extrinsic sources of noise (Raser & O'Shea 2005; Elowitz et al. 2002). In other terms, cell-to-cell variability causes significant variations in gene expression levels, and each cell in the experiment might differ significantly in its

behavior from the others, and from any predicted behavior. Our stochastic simulations are based on the Gillespie algorithm (see chapter 2), which can produce an exact simulation of noise in biochemical reactions, but is limited to intrinsic noise. The study of noise and cell-to-cell variability in gene expression is a well-studied problem but has remained largely limited to simulations of stochasticity in gene expression. Only recently were models fitted to high-quality single-cell datasets to represent the variability of gene expression between cells as a consequence of phenotypic differences in the population (Llamosi et al. 2016). Although, for the purpose of our study the Gillespie-based simulation framework allowed us to test the behavior of our control algorithms when exposed to noisy gene expression, it does not accurately simulate the extrinsic variability of gene expression in a population of different cells. With the quality of the data acquired for this study, an interesting development of our modeling approach would be to apply the aforementioned population models, known as mixed-effect models, to the study of noise and cell-to-cell variability in a populations of toggle-switch-bearing bacteria.

3.2 State space analysis

To better understand the behavior of our toggle switch and be able to intuitively investigate its response to external stimuli, we computed the vector field, nullclines and equilibrium points of our ODE system. The graphical representation helped us visualize the underlying rules that govern the dynamics of our system.

The vector field consists of computing the derivative of our system in its state space (or, in our case, the protein space). It is then typically represented by a so-called "quiver" plot in which arrows represent the gradient vectors for each of the evaluated elements of the state space. Because it is impossible to represent the vector field in the five dimensions that counts our ODE model, and because we are primarily interested in the evolution of the protein level, we did a quasi-steady-state approximation on the mRNA and IPTG levels and used this reduced model to compute the vector field presented in Figure 3-2. The quasi-steady-state approximation of our system can be performed because of the timescale separation of the transcription and translation processes: The time constants for mRNA expression differ from the protein expression time constants by at least one order of magnitude. The pahse portrait is a powerful representation to illustrate key concepts in the dynamics of the genetic toggle switch. In a similar spirit we also represented the nullclines of the toggle switch, which are the curves representing the steady state values for each of the two proteins as a function of other variables of the system. Again, to be able to represent the nullclines in 2 dimensions, the additional approximation of quasi-steady-state mRNA levels was made, i.e. the represented curves are solutions to the ODE model such that: $\frac{d\,LacI}{dt} = 0$ and $\frac{d\,mRNA_{LacI}}{dt}(TetR) = 0$, for all values of TetR, and $\frac{d\,TetR}{dt} = 0$ and $\frac{d\,mRNA_{TetR}}{dt}(LacI) = 0$, for all values of LacI.

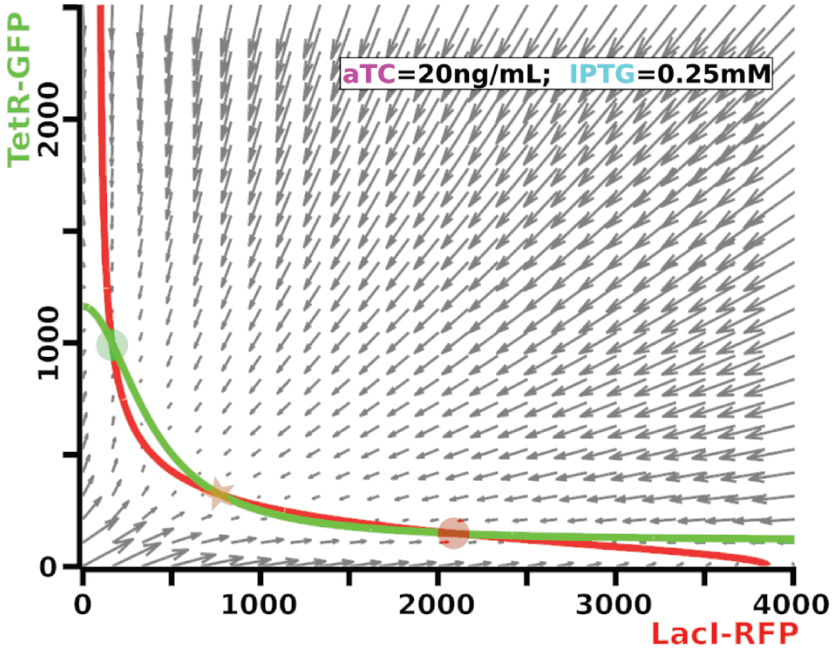

Figure 3-2 State-space representation of the dynamics of the toggle switch from the fitted model for the reference concentrations of inducers. The vector field is represented in gray. The vector field is directed towards the stable equilibrium points, and we can see the "saddle" topology around the unstable equilibrium point. The nullclines for the LacI-RFP and TetR-GFP proteins are represented by red and green lines, respectively. The two stable points are presented by disks of the appropriate color. The unstable point is represented by an orange star. The unstable equilibrium point is located at LacI-RFP = 768 and TetR-GFP = 320. The LacI-RFP-high and TetR-GFP-high equilibrium states were estimated at [2057,159] and [221,997] respectively.

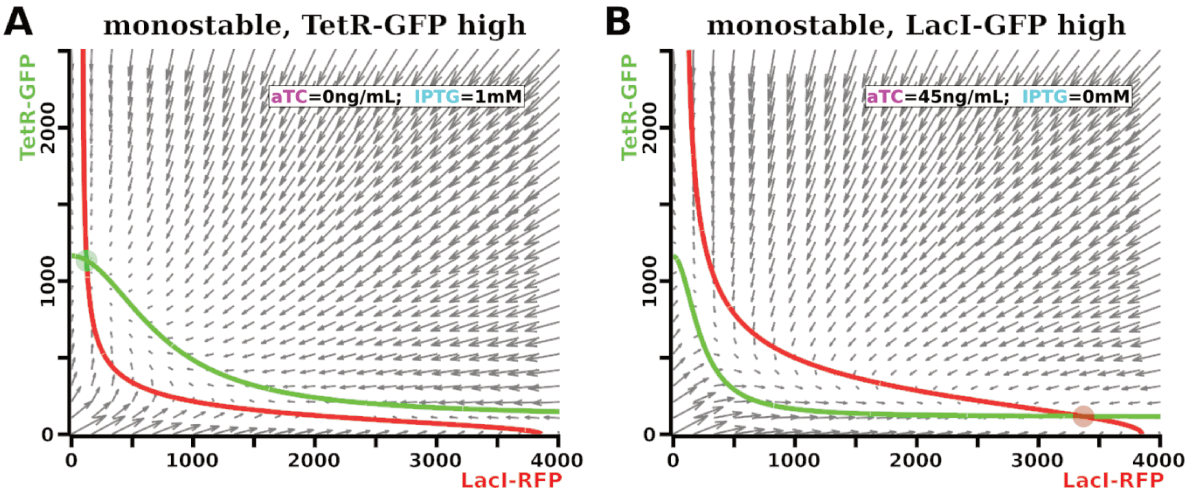

Figure 3-3 State space representation for different values of aTC and IPTG. High concentrations of either of the two inducers makes the system monostable.

This representation helped us better understand how the landscape is modified by changes in the concentrations of IPTG or aTC (see Figure 3-3). Although, as we demonstrate in the next section, an entire part of the state space cannot be made stable with static concentrations of inducers, it is possible to drive the system between the two stable states, and make it pass in the vicinity of the unstable point as shown in Figure 3-4. This result suggests that it should at least be possible with external control to maintain the system in the vicinity of the unstable point.

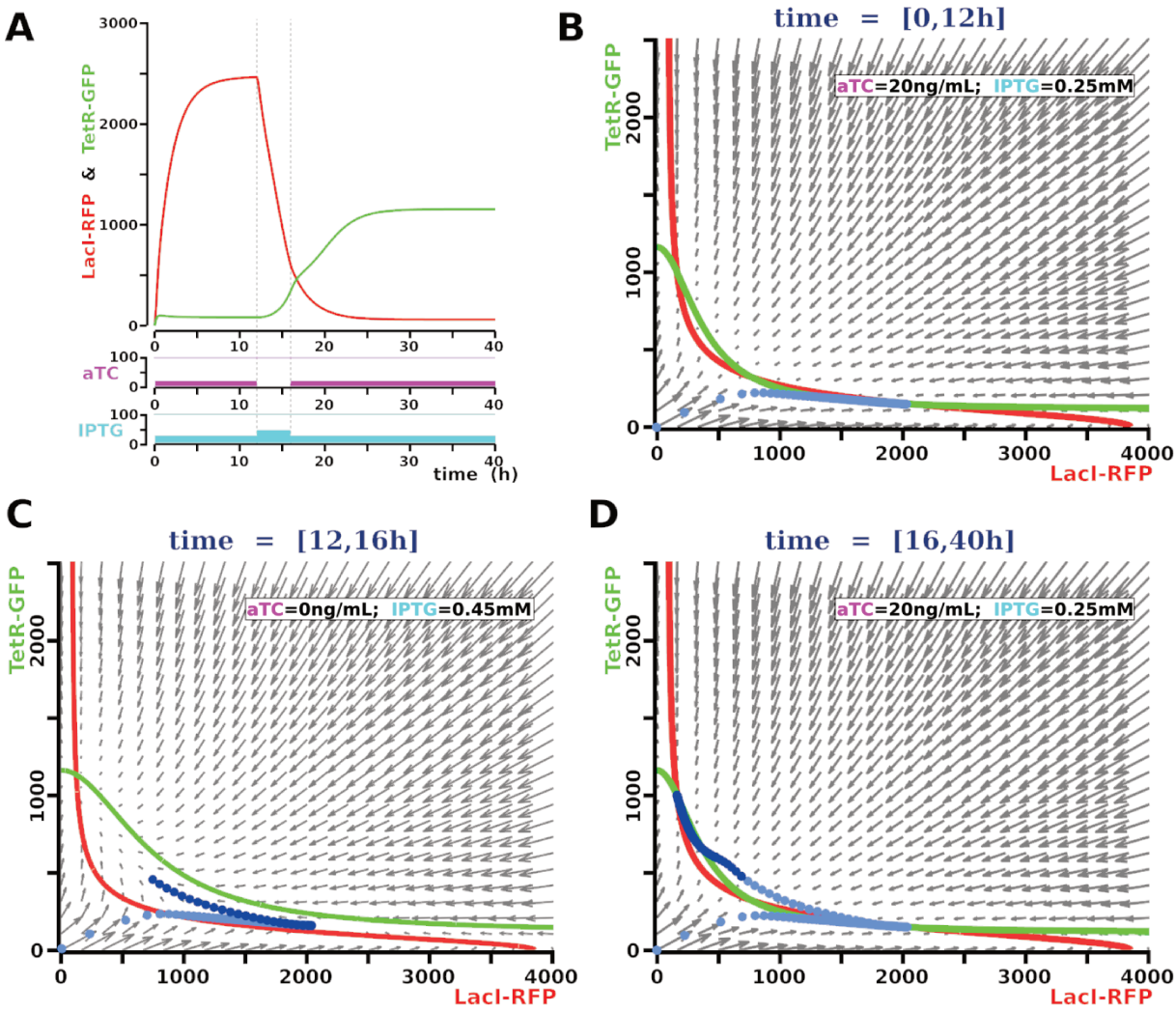

Figure 3-4 Simulated trajectory of a toggle switch. Our toggle switch model is subjected to 3 different environments: First, starting with zero levels of the mRNAs and proteins, the reference concentrations of inducers (20 ng/mL of aTC and 0.25 mM of IPTG) are applied for 12 hours. Then aTC is removed and 0.45 mM of IPTG are applied to the toggle switch for 4 hours. Finally, inducers are switched back to their reference concentrations for 24 hours. A) The LacI-RFP and TetR-GFP responses are plotted against time. The system converges towards the LacI-RFP-high stable state for the first 12 hours, and is then driven towards the TetR-GFP stable state from 12h to 16h. Finally the toggle switch falls in the opposite basin of attraction, TetR-GFP-high, when the reference concentrations of inducers are re-established. B-D) State space representation of the system's dynamics during the 3 different periods of the simulation. The blue dots represent the state of the simulated switch in the LacI-TetR space every ten minutes. Light blue dots represent the trajectory from past periods and darker blue dots represent the trajectory for the current period. An interesting observation, that is primarily visible in panel D, is that the trajectory does not follow exactly the orientation of the vector field. At the beginning of the 3rd period, right after the switch back to balanced concentrations, the systems trajectory follows a trajectory that is closer to the dynamics illustrated the vector field in panel C. As mentioned in the text, this is because the quasi-steady state approximation that was used to draw the vector field in 2D hides some of the delays and inertias inherent to our 5-dimensional model, especially because of the large IPTG delay.

3.3 Stable states and bifurcations

The study of equilibrium points in multistable systems is a domain of research in itself. In this section, in an approach similar to bifurcation analysis, we analyze the stability of our system for different values of IPTG and aTC in the inputs space. Bifurcation analysis aims at identifying key properties of dynamical systems, such as attractors, limit cycles or saddle points, in systems of ordinary differential equations under parameter variation. The domains of application are as diverse as fluid mechanics, electronics, chemistry or theoretical ecology (Strogatz 1994) as this approach is part of dynamical systems theory. In particular stable

points emerge or disappear in the deformable landscape of multistable genetic circuits when environmental conditions change. The emergence of multistability throughout the evolution of the cell is a core concept in cellular decision making, as famously illustrated by Waddington's epigenetic landscape (Waddington 1957). The toggle switch, a fundamental topology for decision making, is an interesting example in this regard (Sekine et al. 2011; Wu et al. 2013). To better understand the dynamics of the toggle switch under changing conditions, we used our model to perform an analysis of its stability similar to bifurcation analysis.

From the model, we could infer the equilibrium points for any combination of IPTG and aTC. By applying various concentrations of inducers we could identify an entire region of the phase

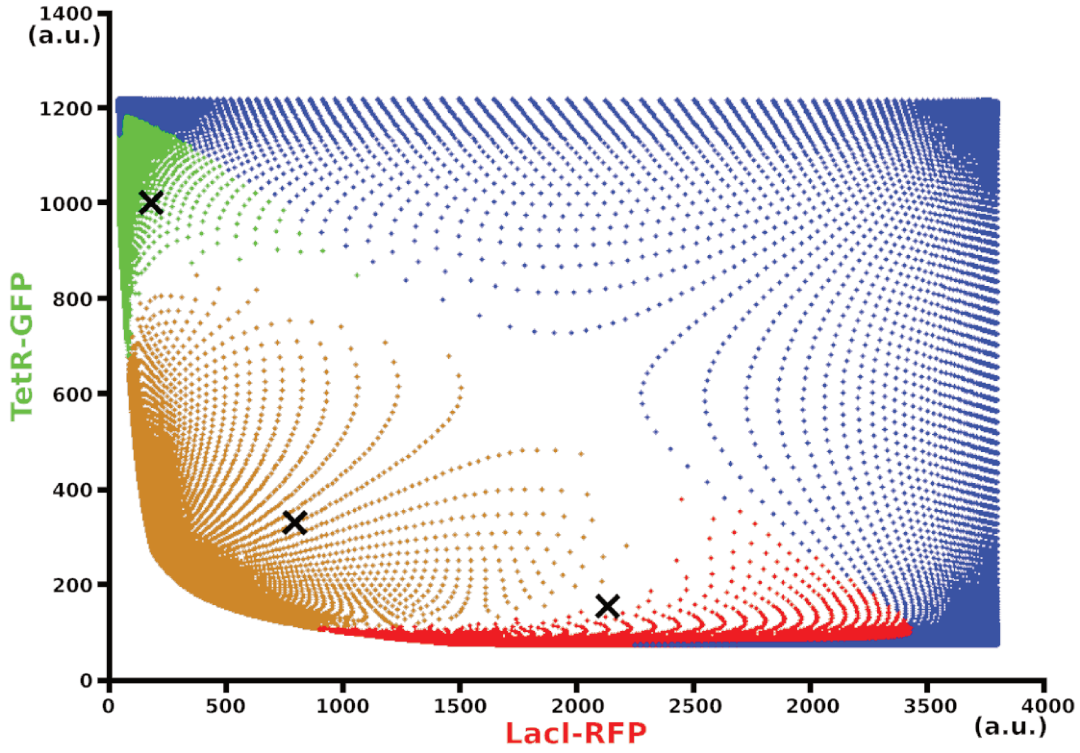

Figure 3-5 Stable and unstable equilibrium points for various concentrations of aTC and IPTG between 0 and 500 ng/mL, and 0 and 5 mM, respectively. The blue dots represent stable equilibrium points for inducer concentrations that make the system monostable. Each one of those blue points is the unique possible steady state of the deterministic system. The red and green dots represent stable equilibrium points for inducer concentrations that make the system bistable. Red dots are the LacI-RFP-high stable states and green dots are the TetR-GFP-high states. The orange dots are the unstable equilibrium points for inducer concentrations that make the system bistable. The area approximately delimited by these dots will be referred to as the unstable area, and no static concentration of inducers can stabilize the system in this area. Driving and maintaining the system in the unstable area requires dynamic control.

space in which no stable equilibrium exist, meaning that the cells could not be maintained with any static concentration of inducers. With a slight abuse of terminology, we call this region of the LacI-TetR space the unstable area. This result demonstrates the necessity of dynamic stimulations to maintain the toggle switch anywhere in its unstable area.

As explained in section 2, we determined experimentally what we call reference concentrations, i.e. concentrations that make the toggle switch bistable, that were then used

for the PI control algorithms (see parameter $u^0_{aTC/IPTG}$ in section 4.3.2 of chapter 2). However in Figure 3-6A we present all the possible levels of aTC and IPTG concentrations that could make the toggle switch bistable. An interesting combination of Figure 3-5 and Figure 3-6A is Figure 3-6C where the TetR to LacI ratio of equilibrium points vs inducer levels is presented. The surface created by all the equilibrium points is similar to a cusp catastrophe curve (Strogatz 1994). The hysteresis behavior that is typically associated with toggle switches can be seen as a trajectory over this landscape. Although catastrophe theory has been applied early on to biological problems (Poston & Stewart 1979), to my knowledge the parallel between the toggle switch and catastrophe theory has not been drawn before.

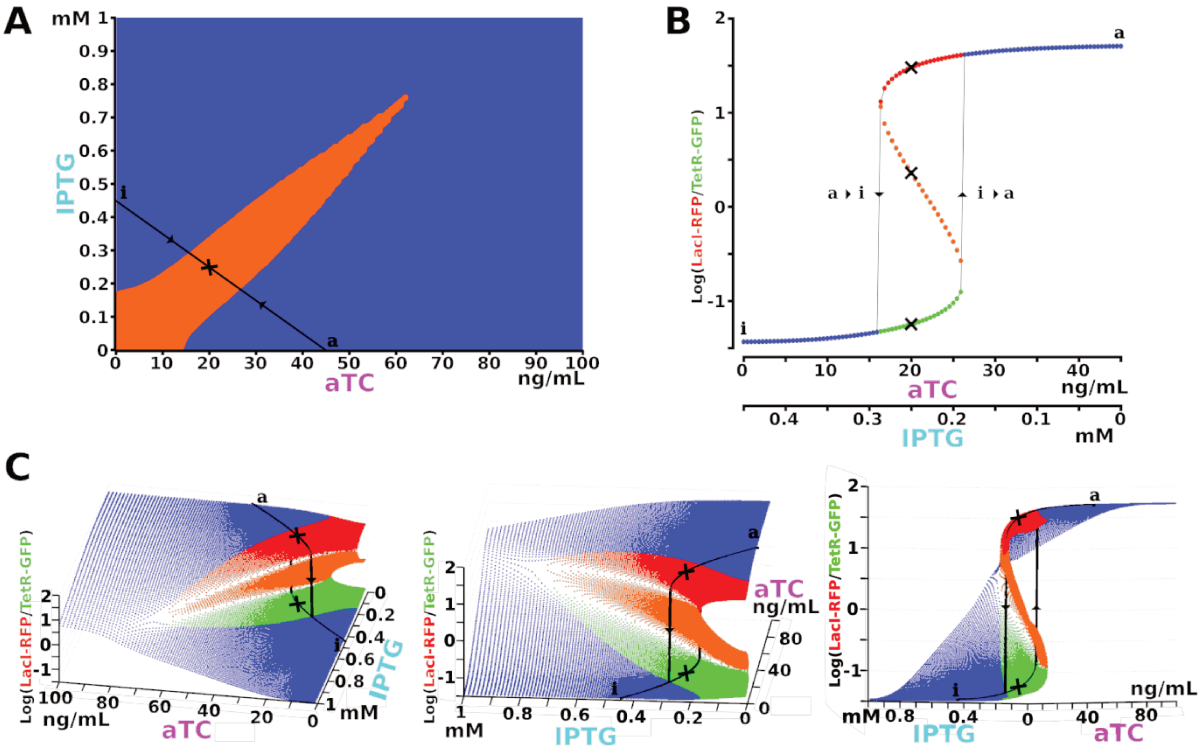

Figure 3-6 Bifurcation analysis. A) The orange region represents values of the inducer concentrations for which the system is bistable. The black cross represents the values of the "reference" concentrations of inducers determined experimentally as IPTG = 0.25 mM and aTC =20 ng/mL. The segment going from point a (IPTG = 0, aTC = 45ng/mL) to point i (IPTG = 0.45mM, aTC = 0) and that passes through the reference concentrations is represented in black. B) The base-10 logarithm of the LacI/TetR ratio of the stable and unstable equilibrium points of the system are represented for different concentrations of IPTG and aTC along the line between point a and point i. The monostable points are represented in blue. The LacI-high and TetR-high stable points of the bistable region are represented in red and green respectively. The unstable equilibrium points of the bistable region are represented in orange. In the bistable region, the steady state of a toggle switch system depends on its history: If the system goes from point a to point i, it will stay in the LacI-RFP-high basin of attraction until it reaches a concentration of approximately 15ng/mL of aTC and 0.3mM of IPTG and switches to the TetR-GFP-high basin. In the other direction, the system will stay in the TetR-GFP-high basin until it reaches about 25ng/mL of aTC and 0.20mM of IPTG and switches. This hysteresis behavior is typical of toggle switches. The black crosses represent the stable and unstable states for the balanced concentrations of inducers. C) Different views of the cusp-catastrophe curve. The base-10 logarithm of the LacI/TetR ratio of the stable and unstable points of the system is represented for different concentrations of IPTG and aTC in their concentrations ranges of 0-1mM and 0-100ng/mL respectively. The hysteresis line from parts A and B of this figure is represented in black. The black crosses represent the stable states for the reference concentrations of inducers (unstable state not represented).

4 Single-cell toggle switch control

With the knowledge acquired from the characterization experiments and the analysis of the fitted model, we focused on single-cell control of a toggle switch. While our ultimate goal was to control cells in vivo, we started by investigating different control approaches in silico and then used the algorithms on the experimental platform.

Our objective is to maintain cells bearing a toggle switch system in the unstable area. For this reason the concentrations for the unstable equilibrium point in the reference conditions (LacI-

RFP = 750 and TetR-GFP = 350)[1] were used as the objective for setpoint control experiments described in this section. As discussed earlier, this is analogue to the benchmark control problem of the inverted pendulum, in which the objective is to maintain a pendulum at its unstable equilibrium point. As a benchmark problem, a large number of control frameworks have been tested on this system, and in a similar spirit we present the genetic toggle switch as a model of a multistable gene regulation network for evaluating the potential of external control platforms for gene expression. In this section we use two feedback control frameworks to address this control problem: proportional-integral control and bang-bang control.

We started by studying the feasibility of controlling the switch in silico using deterministic or stochastic models to simulate the effects on populations of cells. After the in silico study we performed the control experiments on single cells bearing the toggle switch.

4.1 PI control

Because of their simplicity, we considered using two independent proportional-integral (PI) controllers to drive each protein to its target level. In this framework, the LacI-RFP controller modulates the aTC concentration and hence drives the pTet promoter activity, and the TetR-GFP controller modulates the IPTG concentration (see Figure 4-1). Two aspects make this problem non-trivial. First, inducers alleviate the repression of their corresponding protein on gene expression. Therefore, when a protein is in low concentration modulating the level of the inducer has very little impact. Second, because of the structure of the network the two controllers, acting independently, often compete against each other. Indeed, by maintaining the level of their controlled protein to relatively low levels, they decrease the effectiveness of the other controller's actions.

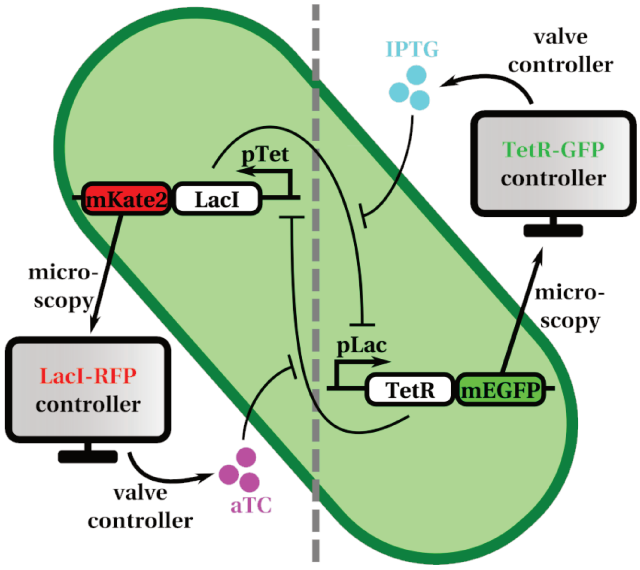

Figure 4-1 The control approach: Two independent controllers are used to drive each branch.

[1] As described in section 2, the estimated levels for the unstable equilibrium point are actually slightly different, with LacI-RFP = 768 and TetR-GFP = 320, but the decision for the value of the control setpoint was based on an earlier estimation of those levels.

4.1.1 *In silico* control

We implemented two Proportional-Integral controllers presented in section 4.5 of chapter 2 on each branch of the simulated toggle switch. The toggle switch was simulated deterministically and stochastically. We controlled a stochastically simulated toggle switch, but also simulated the response of other cells submitted to the same environment to investigate their response. The non-controlled cells eventually drift away from the cell being controlled and fall back into one of the basins of attraction of the stable equilibrium states.

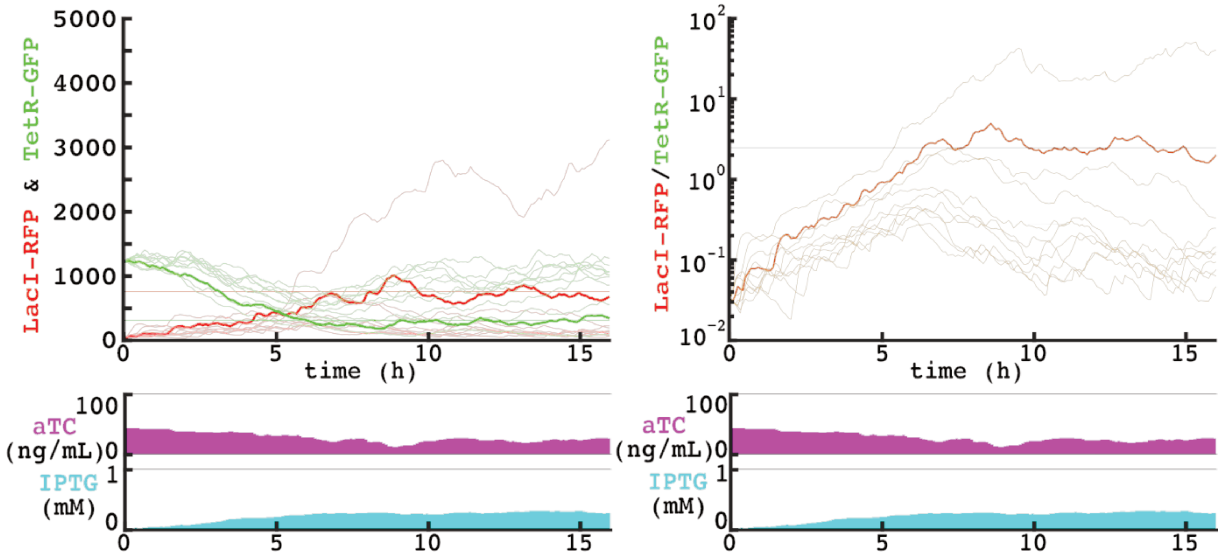

Figure 4-2 PI control results for the stochastic model. The left panel represents the simulation results for the levels of each protein, and the right panel represents the same results but as a ratio between the two fluorescence levels. The thicker, darker curves represent the controlled cell and the thinner lines represent other simulated cells submitted to the same inputs. The straight lines represent the control setpoint . At the bottom are the concentrations of each inducer applied to the cells as a decision from each controllers. After a while, all non-controlled cells start drifting away from the objective levels.

4.1.2 *In vivo* control

We initially used the parameters from the in silico control experiments. However the control actions were too weak (i.e. the proportional parameters were too low) to extract the system out of its basin of attraction. After a few manual adjustments of the parameters of the two controllers, we obtained a control strategy that was effective to maintain an individual cell, chosen at random among the observed cells, far from the two stable equilibrium states and in the vicinity of our target control point (see Figure 4-3). We also observed large oscillations around target values, a likely consequence of the hysteresis of the circuit with respect to inducer changes.

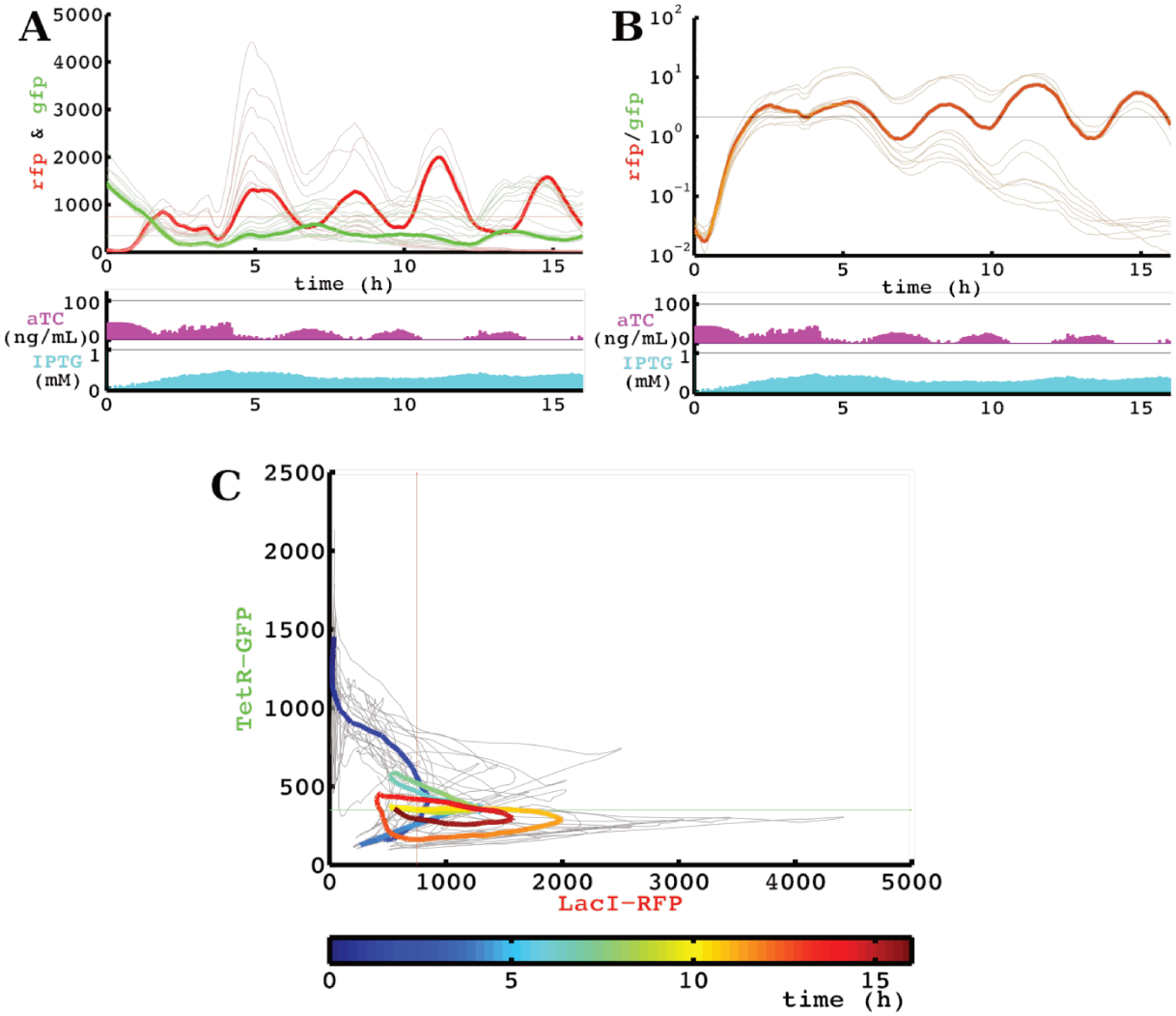

Figure 4-3 Experimental PI control results. The highlighted trajectory in each plot is the controlled cell. The controlled cell is maintained in the vicinity of the setpoint objective, whereas the other cells drift away.

Very surprisingly, we observed in some control experiments that the applied inputs, defined so as to drive one particular cell, were effective to drive the whole cell population (see Figure 4-4). Because of the stochasticity of gene expression and the existence of significant cell-to-cell differences, it was highly unexpected that the same stimulations can maintain several cells far from their equilibrium states. Following our analogy, it would be equivalent to driving a set of inverted pendulums differing by their height and size and each under individual disturbances by applying a unique force (see Figure 4-5). Note that this entrainment of other cells is of variable duration and is not always observed. Using our model, we investigated whether similar responses were found in silico.

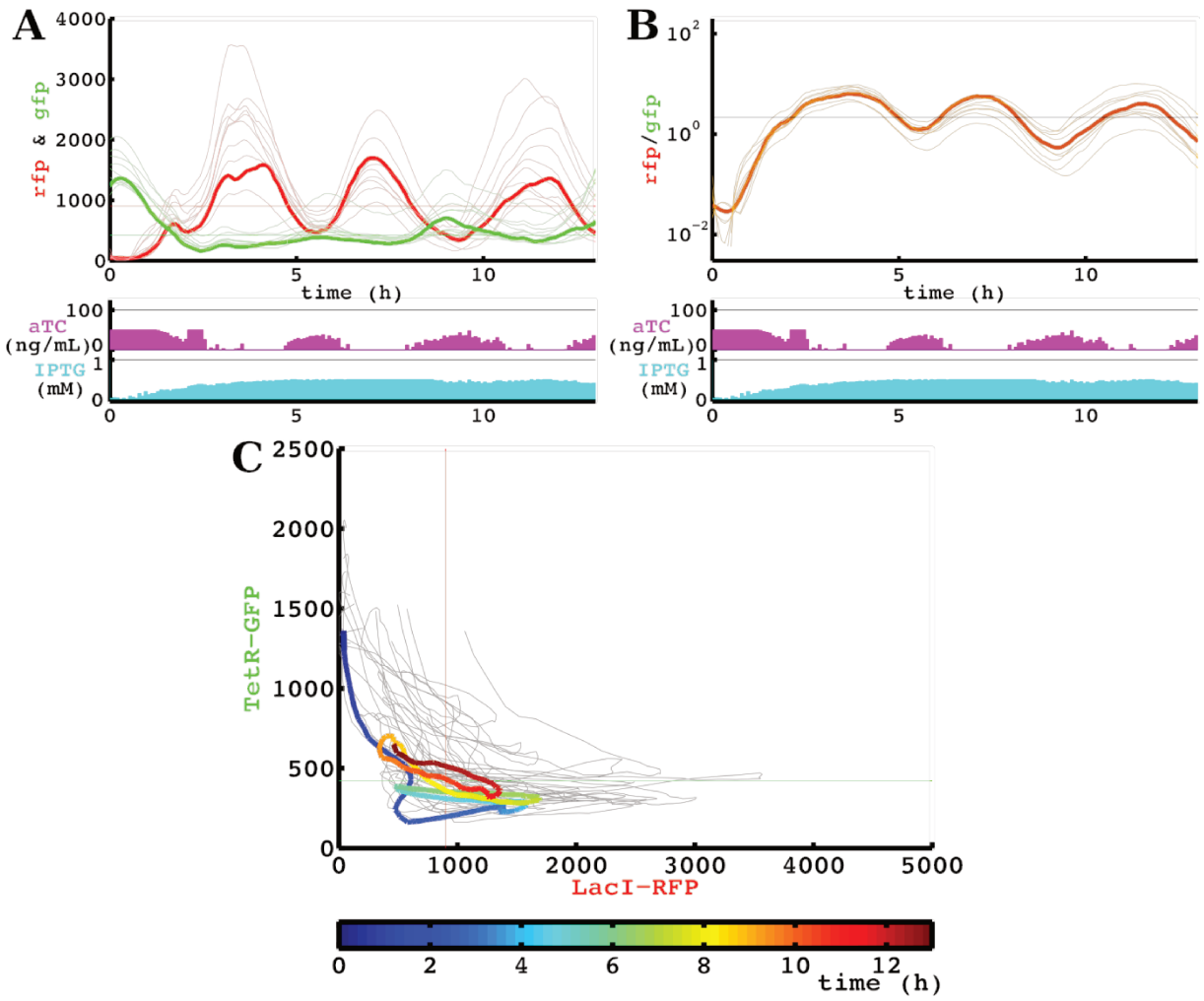

Figure 4-4 Experimental PI control result. All cells stay in the vicinity of the objective setpoint.

Numerical simulations suggested that the application of strong control actions helped to keep non-controlled cells away from the two attracting states, at the cost of creating more pronounced oscillations around the target point. This motivated us to investigate the use of a bang-bang control framework using all-or-none control decisions.

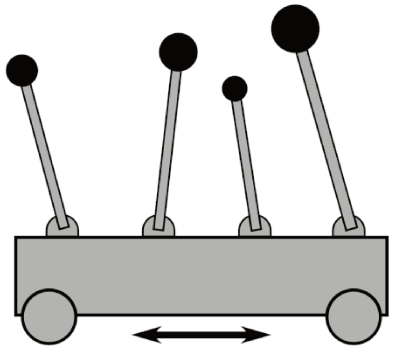

Figure 4-5 The challenge of controlling a population of toggle switches. Since all cells are different, it seems difficult to keep them all together in the unstable area by applying the same inputs. In our analogy with the inverted pendulum, it corresponds to trying to maintain several different pendulums upright with a single driving force.

4.2 Bang-bang control

In bang-bang controllers, either the minimal or maximal concentration of inducers is applied depending on the sign of the difference between the target and the observed fluorescence values. This crude control requires minimal knowledge from the experimenter and is sometimes an optimal control strategy for some applications, especially in minimum-time problems (Brogan 1990).

4.2.1 Model

We observed both numerically and experimentally that stronger PI parameters would produce better results at the population level. Following that logic we implemented a dual bang-bang controller, as described in section xx of chapter 2, and investigated its effect on simulated deterministic toggle switches and on simulated populations of stochastic toggle switches. We observed better results at the population level with this control framework, but the results at the single-cell level are fundamentally limited by the inertial dynamics of the system.

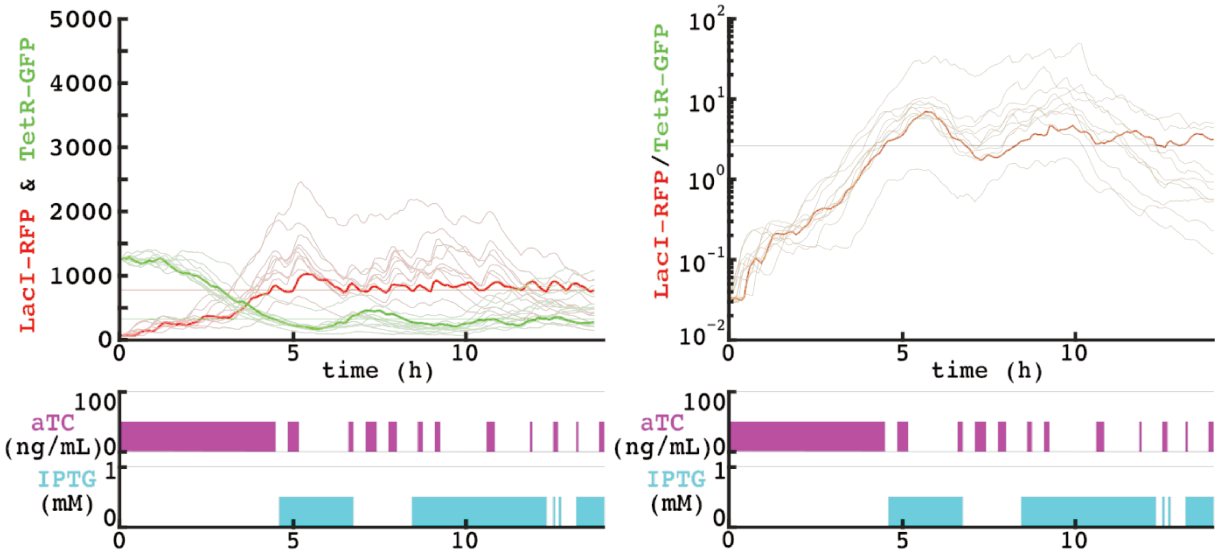

Figure 4-6 In silico bang-bang experiments. With this approach better control results are obtained at the population level.

Numerical simulations indicated that better control performance can be obtained with smaller oscillations around the target values and with superior entrainment capacities for the non-controlled cells.

4.2.2 Experiments

Bang-bang control experiments showed control performances that are comparable to the double PI controller but with better population entrainment capacities (see Figure 4-7). At the level of individual proteins, TetR-GFP followed accurately its target value, whereas LacI-RFP showed marked oscillations.

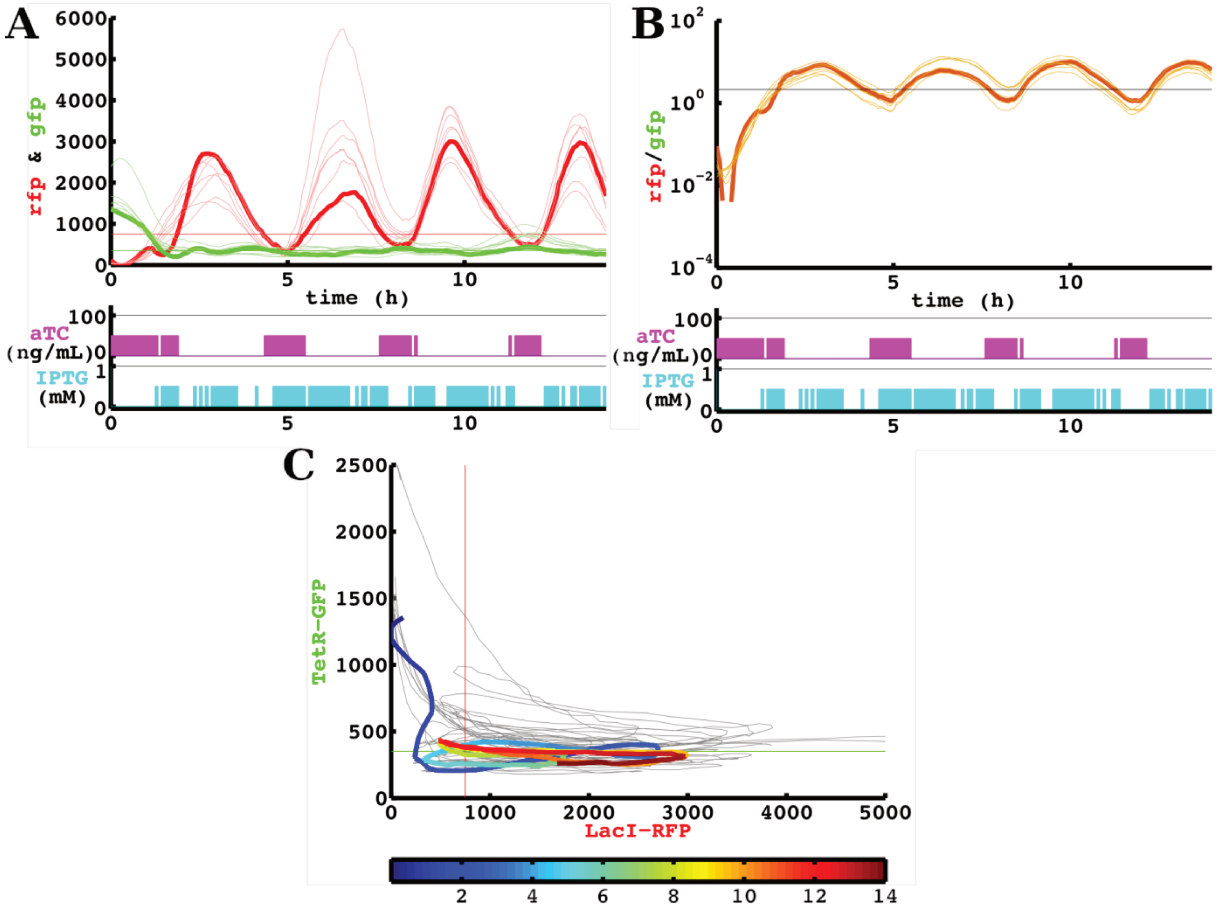

Figure 4-7 Experimental bang-bang control experiment. The cells seem closer together, suggesting that driving forces are more homogenous for the cells in the population in this type all-or-none stimulations. The dynamics of the system make the controlled cell strongly overshoot in RFP in this bang-bang approach.

5 Open-loop dynamic stabilization

To understand why the entire population followed the controlled cell in some of the PI and then in the bang-bang control experiments, we studied the dynamics of the system under strong stimulations. The crude control approach of the bang-bang strategy, made of repeated maximal induction of the two branches of the circuit suggested that fast, periodic alternations of the aTC and IPTG levels could be used to drive cells away from both attracting states in an open-loop manner. And indeed, in silico simulations showed that periodic stimulation of the two branches of the circuit could drive all cells in a state of balanced expression (see Figure 5-1).

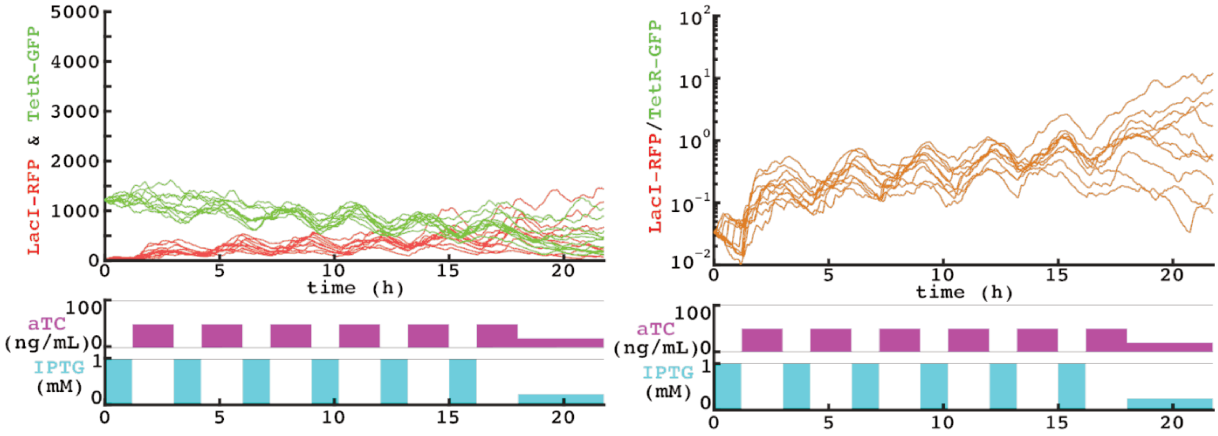

Figure 5-1 Stochastic simulation of periodic stimulations. The cell population can be stabilized in the unstable area by alternating between the two inducers.

We tested this hypothesis experimentally and were able to drive the fluorescence of the cells into the unstable area of the state space (see Figure 5-2), albeit for input profiles that differed from the ones predicted in silico. Strikingly, we were able to drive the cell population close to the separatrix of the two basins of attraction of the toggle switch in reference conditions as evidenced by the observed partitioning of the cell population observed when interrupting the periodic oscillations (Figure 5-3).

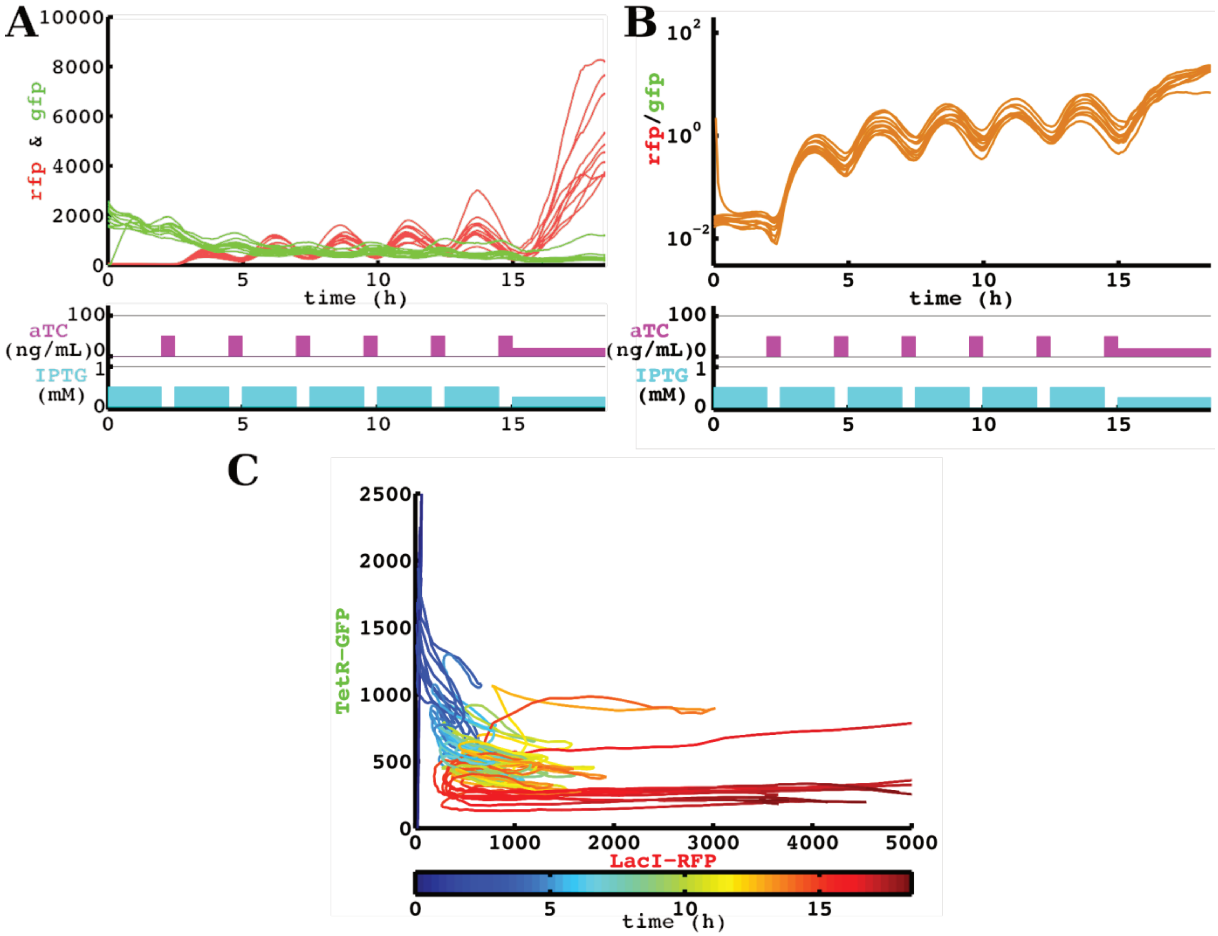

Figure 5-2 Experimental results showing dynamic stabilization coming from periodic stimulations. No particular trajectory is highlighted. In panel C) all cell trajectories in the are represented proteins space and color-coded in time. With periodic

alternations between the two inducers, the population of toggle switches is maintained in its unstable area for about 11h, and then falls into the LacI-RFP-high basin of attraction when periodic stimulations are interrupted.

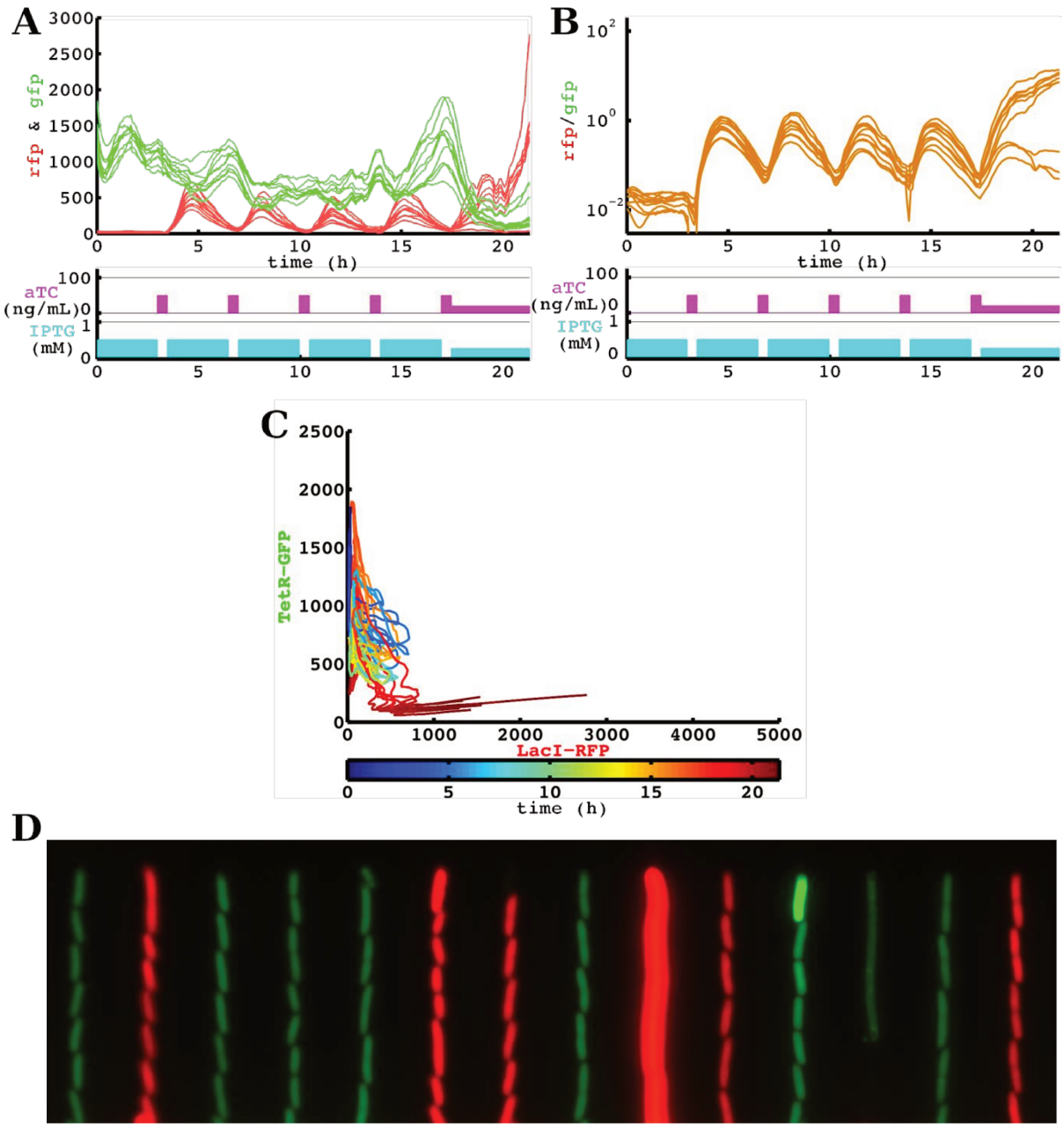

Figure 5-3 Experimental dynamic stabilization. In first two panels A) and B), the highlighted trajectory is the average trajectory of the population of cells. In panel C) all cell trajectories in the are represented proteins space and color-coded in time. E) Composite RFP-GFP image at the end of the experiment.

This stabilization of the system in the unstable region is a dynamic stabilization phenomenon. Dynamic stabilization is the process by which a system, unstable in at least parts of its operating state space, is trapped dynamically within its domain of instability due to the influence of an external high-frequency oscillating input and of the systems' own inertial forces (Landau & Lifshitz 1976). Dynamic stabilization was first introduced by Kapitza for an inverted classical pendulum stabilized by rapidly oscillating external modulations (Kapitza 1951). In short, a pendulum which base is subjected to fast vertical oscillations becomes stable in its upward, unstable equilibrium position. Dynamic stabilization techniques were soon applied to a number of stabilization problems, with remarkable achievements in domains

including synchrotron beam focusing (Courant et al. 1952), plasma containment (Osovets 1960), and quantum-mechanical systems (Grozdanov & Raković 1988). The main advantage of this type of approach is that, unlike real-time feedback control, it does not require observation of the controlled system.

Using our fitted model of the toggle switch, we computed the representation of the dynamics of the system in the protein space averaged in time over a period of our oscillatory driving inputs. Figure 5-4 helps understand the observed synchronous behavior of the cell population. This time-averaged vector field presents a unique null point, acting like a globally-attractive equilibrium point for the slowly-evolving genetic system. Therefore periodic oscillations of the environment were able to reset the state of a population of cells into an "undifferentiated" state, with low levels of either genes being expressed, from which several cell fates were possible once the external, global forcing was released.

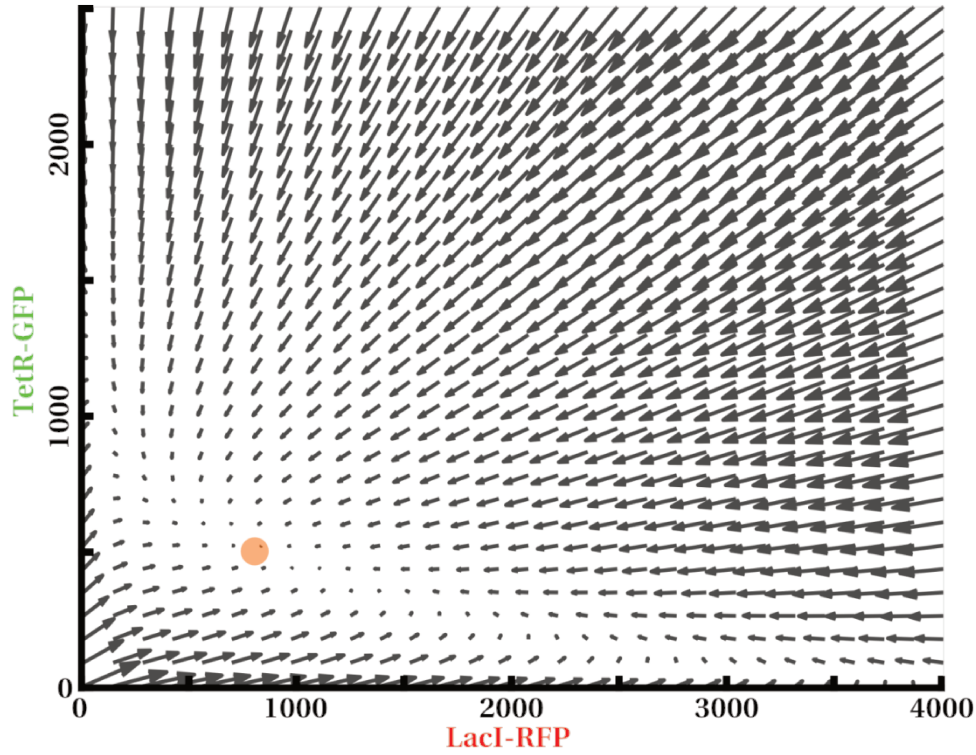

Figure 5-4 Vector field of the system averaged over a period of aTC and IPTG stimulation. A null point appears in the unstable area.

6 Conclusion

In this chapter we studied the possibility of real-time control on a synthetic bistable gene regulation network, the genetic toggle switch. Here, we showed two approaches to maintain a bistable genetic circuit close to its unstable equilibrium state: closed-loop control and periodic stimulation. Strikingly, in most cases, the proposed solution is effective to maintain the whole cell population away from the two attractive states despite the stochasticity of gene expression and phenotypic heterogeneity. This result was unexpected but could be explained by time-

scale separation: subjected to fast periodic stimulations, the comparatively slower genetic circuit approximately follows a dynamics corresponding to the time-averaging of two opposing vector fields that presents a globally-stable equilibrium point at intermediate concentrations for LacI and TetR proteins.

As a contribution to in silico cybergenetics, we demonstrated that single-cell control of a bistable system can be achieved with relatively simple control frameworks and very little a priori knowledge of the system. This result opens the door to the study of gene regulation networks of increased complexity and in operating domains unreachable or unsustainable thus far. The architecture of our control framework demonstrates the possibility for two controllers to steer two connected processes without communicating with each other. It illustrates the possibility to perform multiple-input-multiple-output control studies of genetic networks, which would facilitate and accelerate the examination of internal network dynamics, but it also demonstrates that control can be performed on networks where, due to only partial knowledge of the regulatory linkage, the controllers might compete against each other.

The quality of the single-cell data obtained with our platform and the prolonged observations of the toggle switch system in its domain of unstability also allowed us to expand the general knowledge on this fundamental circuit. Toggle switches, and multistable systems in general, are known to play a central role in cell fate determination. Indeed, decision processes are understood as the continuous transformation of a stable equilibrium in which cells reside into an unstable one, separating the state space in several basins of attractions of novel equilibria, corresponding to the different possible futures of the cells (Waddington & Kacser 1957; Balázsi et al. 2011; Wang et al. 2011). The toggle switch is also a fundamental component of synthetic biology circuits. Therefore, in addition to its importance as a test assay for control of complex circuits in living organisms, the control methods we outlined in this article are relevant to drive and understand cell decisions, cell fate, differentiation and dedifferentiation dynamics, but also analyze the input-output functions of core elements of synthetic biology circuits.

Finally, the stabilization of an entire population of toggle switch cells in the unstable area of the system in open-loop experiments is a new observation that was suggested by closed-loop control experiments and new knowledge acquired on the dynamics of the system. This dynamic stabilization phenomenon, similar to the inversion of stability in the Kapitza pendulum (Kapitza 1951), could be used both as a control approach for multistable systems, possibly to reset the state of cellular decision making circuits, but also in general dynamic stabilization could be studied as a possible range of operation for multistable networks. Oscillating morphogen stimulations have already been shown to trigger differentiation processes in embryos (Kirst et al. 2016; Sorre et al. 2014; Aulehla & Pourquié 2010). The study of dynamic stability in cellular decision networks in fluctuating concentrations of inducers can expand our understanding of those processes and create a new framework of analysis for those systems.

Ashyraliyev, M. et al., 2009. Systems biology: parameter estimation for biochemical models. *FEBS Journal*, 276(4), pp.886–902.

Aulehla, A. & Pourquié, O., 2010. Signaling gradients during paraxial mesoderm development. *Cold Spring Harbor perspectives in biology*, 2(2).

Balázsi, G., van Oudenaarden, A. & Collins, J.J., 2011. Cellular decision making and biological noise: from microbes to mammals. *Cell*, 144(6), pp.910–25.

Brogan, W.L., 1990. *Modern Control Theory* 3rd ed., Prentice Hall.

Courant, E.D., Livingston, M.S. & Snyder, H.S., 1952. The Strong-Focusing Synchroton—A New High Energy Accelerator. *Physical Review*, 88(5), pp.1190–1196.

Elowitz, M.B. et al., 2002. Stochastic gene expression in a single cell. *Science*, 297(5584), pp.1183–6.

Grozdanov, T.P. & Raković, M.J., 1988. Quantum system driven by rapidly varying periodic perturbation. *Physical Review A*, 38(4), pp.1739–1746.

Kapitza, P.L., 1951. Dynamic stability of a pendulum with an oscillating point of suspension. *Journal of Experimental and Theoretical Physics*, 21, pp.588–597.

Kirst, C., Timme, M. & Battaglia, D., 2016. Dynamic information routing in complex networks. *Nature Communications*, 7(7), p.11061.

Landau, L.D. & Lifshitz, E.M., 1976. *Course of theoretical physics Vol. 1. Mechanics* 3ed., Pergamon.

Llamosi, A. et al., 2016. What Population Reveals about Individual Cell Identity: Single-Cell Parameter Estimation of Models of Gene Expression in Yeast J. Stelling, ed. *PLOS Computational Biology*, 12(2), p.e1004706.

Osovets, S.M., 1960. Dynamic stabilization of a plasma ring. *Journal of Experimental and Theoretical Physics*, 12(2), pp.221–224.

Poston, T. & Stewart, I., 1979. *Catastrophe Theory and Its Applications* First Edit., Pitman.

Raser, J.M. & O'Shea, E.K., 2005. Noise in Gene Expression: Orgins, Consequences, and Control. *Science*, 309(5743), pp.2010–2013.

Sekine, R. et al., 2011. Tunable synthetic phenotypic diversification on Waddington's landscape through autonomous signaling. *Proceedings of the National Academy of Sciences of the United States of America*, 108(44), pp.17969–17973.

Sorre, B. et al., 2014. Encoding of Temporal Signals by the TGF-β Pathway and Implications for Embryonic Patterning. *Developmental Cell*, 30(3), pp.334–342.

Strogatz, S.H., 1994. *Nonlinear Dynamics and Chaos*, Westview Press.

Villaverde, A.F. & Banga, J.R., 2013. Reverse engineering and identification in systems biology: strategies, perspectives and challenges. *Journal of The Royal Society Interface*, 11(91), pp.20130505–20130505.

Waddington, C. & Kacser, H., 1957. *The strategy of the genes*, London: Allen & Unwin.

Waddington, C.H., 1957. *The strategy of the genes* Routledge., Routledge.

Wang, J. et al., 2011. Quantifying the Waddington landscape and biological paths for development and differentiation. *Proceedings of the National Academy of Sciences of the United States of America*, 108(20), pp.8257–8262.

Wu, M. et al., 2013. Engineering of regulated stochastic cell fate determination. *Proceedings of the National Academy of Sciences of the United States of America*, 110(26), pp.10610–10615.

Chapter IV
Stack segmentation

"Many facets of [cell segmentation] appear to be well within the grasp of present-day *technology."*

Prewitt & Mendelsohn 1966

Table of Contents

1 Introduction: Automated cell segmentation, a core problem for single-cell control 119

1.1 Cell segmentation: Filtering, mathematical morphology and active contours 119

1.2 Deep learning: the revolution. 122

1.2.1 Artificial neural networks 122

1.2.2 Convolutional Neural Networks 124

1.2.3 U-Net 125

1.3 Hyperspectral imaging 128

1.3.1 Support Vector Machines 129

1.3.2 The framework of hyperspectral imaging 131

2 Z-stacks and focal signatures 132

3 Training 134

3.1 Training set and evaluation set construction 134

3.2 Data normalization and focus invariance 135

3.3 Feature extraction 137

3.4 SVM fitting 138
3.5 Evaluation 139
4 Classification 140
4.1 Class prediction and confidence map 140
4.2 Results 141
4.2.1 Escherichia coli – Mother Machine 141
4.2.2 Saccharomyces cerevisiae 145
4.2.3 Saccharomyces cerevisiae & Escherichia coli 145
4.2.4 HeLa cells 146
5 Segmentation 148
5.1 Escherichia coli – Mother Machine 148
5.2 Saccharomyces cerevisiae & Escherichia coli 149
5.3 HeLa cells 150
6 Towards 3D segmentation 152
7 Conclusion 153

1 Introduction: Automated cell segmentation, a core problem for single-cell control

The amount of data available to biologists dramatically increased at the turn of the century. With the democratization of next generation sequencing techniques, microarrays, RNA-Seq, and imaging techniques, the field has definitively entered the era of high throughput acquisitions and big data analysis. A new field, quantitative biology, emerged out of these innovations and draws upon a multitude of approaches from the physical sciences and engineering to make biology quantitative and predictive. However, even though progresses were made in the field of multi-channel time-lapse microscopy imaging, the analysis of the formidable amount of data generated by such methods still often requires manual input from the experimenter. Such a procedure typically requires a lot of work from the experimenter and not only limits the throughput of the approach, but also makes on-the-fly segmentation impossible, which is crucial in the case of single-cell control.

1.1 Cell segmentation: Filtering, mathematical morphology and active contours

Before the changes brought about by microscope automation and microfluidics on one hand, and the advent of digital image processing on the other, cell segmentation and data extraction used to be performed manually. Since then a number of image analysis and segmentation tools have been developed to help the experimenters in their task. So far, the tools proposed can be considered semi-automatic as they require a significant amount of post-processing to achieve satisfactory segmentation results and make cell tracking possible.

The first, simplest form of automatic image segmentation is filtering and thresholding. In cases where the cells are well separated and their pixel intensity is homogeneous and different from the background (for example, in phase contrast microscopy images or in fluorescence imaging), a simple histogram threshold can segment cells at minimal computational cost. It can also be quite robust: Automatic threshold selection (Otsu 1979; Glasbey 1993) is usually preferred to hard-set thresholds, which has the double advantage of finding an optimal threshold value without the need for heuristics and allows the algorithm to adapt to changes in illumination between images. Another improvement is the use of adaptive thresholding (Yanowitz & Bruckstein 1988), which partitions an N x M image in overlapping windows of n x m pixels and finds local thresholds for each. A thresholding map is then interpolated for the entire image. This partitioning approach is especially useful in images where, due to uneven illumination, the luminosity and contrast between the cells and the background varies within the image. However these thresholding methods become insufficient as soon as the cells in the image are not well separated, which is almost always the case in time-lapse movies of growing populations of cells.

The next level of cell segmentation is feature extraction. A more robust approach is to identify cell features, usually cell boundaries. Instead of identifying the cell region entirely, the boundaries are identified and cell region is inferred from it. This type of approach uses a variety of linear image filters that transform the image into a map of the features of the image first and then applies a threshold to the result to identify the borders. One of the advantages of

this approach is that it is inherently robust to changes in illumination, as long as the contrast between the cells boundary and the background is not too low for efficient thresholding. The boundary detection methods usually apply first-order (gradient) or second-order (Laplacian) operations to the image to enhance edges or ridges, and then threshold the transformed image. The most common algorithms for edge detection usually combine these operations with various preprocessing steps, and then apply thresholding methods discussed previously to identify boundaries(Davis 1975; Canny 1986). If the boundaries identified do not form closed paths, either the two endpoints are connected directly by a line, or the threshold level can be lowered progressively until the path is closed.

Thresholding and feature detection are usually considered a first step in the majority of cell segmentation algorithms. In most cases, the objects identified via thresholding actually consist of multiple cells next to one another. The boundaries identified via edge detection often over- or under-segment because of noise, to which first and second order derivatives are especially sensitive, or because the magnifications and numerical apertures of microscope objectives limit the optical resolution of the images, and the details at the intersection between cells are not discernable. Simple thresholding or edge detection do not succeed at identifying those separations, since in most cases they do not produce a visually different region between cells. Another problem arising from noisy images or other objects in the environment of the cells is false positives, where the algorithms wrongly identify parts of the image as cells. The opposite also happens, in some cases some cells can look significantly different from the rest of the population, and are left out by the segmentation algorithm. At this point, spatial considerations come into play.

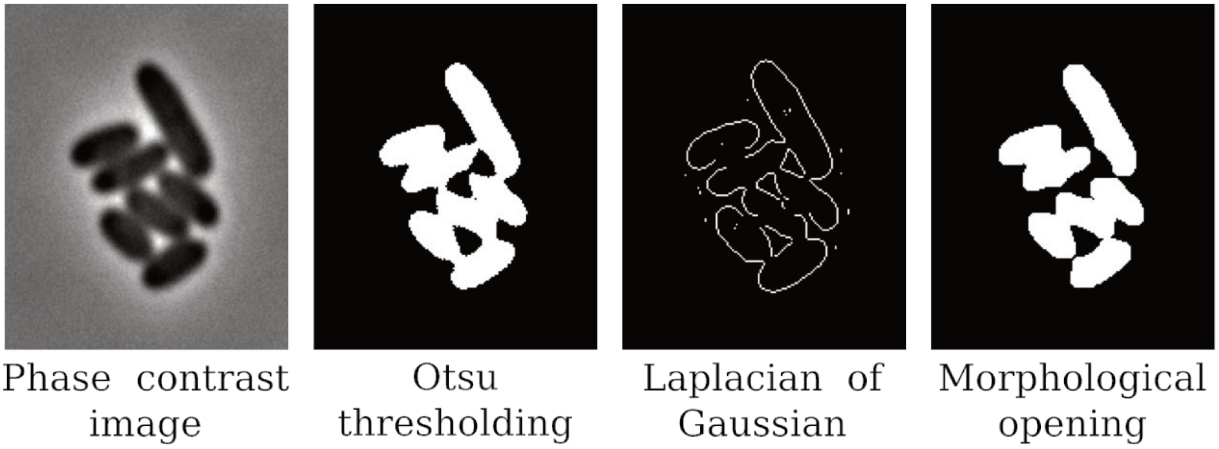

Figure 1-1 Binarization and post-processing of a 100X phase contrast image of Escherichia coli. The Otsu algorithm is a method to automatically identify the threshold level to binarize an image by maximizing the inter-class variance between the two identified regions. All bacteria are still connected. Laplacian of Gaussian (LoG) is an edge detection technique consisting in the application of a Gaussian 2D filter on the image and then applying the Laplacian operator on the image. Thresholding is then performed to identify edges. Morphological opening with a disk structuring element of radius 4 was applied to the Otsu thresholding image. Morphological opening can separate connected objects, but it can lead to over- or under-segmentation.

Mathematical morphology is often used as a post-processing step for edge detection and thresholding, and consists of operations of arbitrary complexity based on a small number of fundamental, non-linear operators(see Dougherty & Lotufo 2003 for a review). Mathematical morphology was initially designed for binary images in the 1960s, but later got expanded to

grayscale images and even RGB images. But binary morphology operations can be used to smooth objects contours, filter out small objects or to fill holes in the shapes. It can also be used to connect unconnected objects depending on the distance separating them, or to separate connected objects with narrow 'bridges' between them. The most commonly used morphology operations in cell segmentation are the distance transform and the watershed segmentation method. The former computes the distance of each pixel in the binary input image from its nearest '0' pixel, and the latter identifies boundaries between objects from a gray scale image by seeding each object in local minima and spreading them iteratively through the levels until the objects connect (see Figure 1-2). This method is a low level assumption on the shape of the cells: In essence, it segments the objects into convex sub-parts. Because the shape of the cells is often irregular, this method has a tendency to over-segment. A number of improvements can be added to the technique, but still more processing steps are usually required after watershedding.

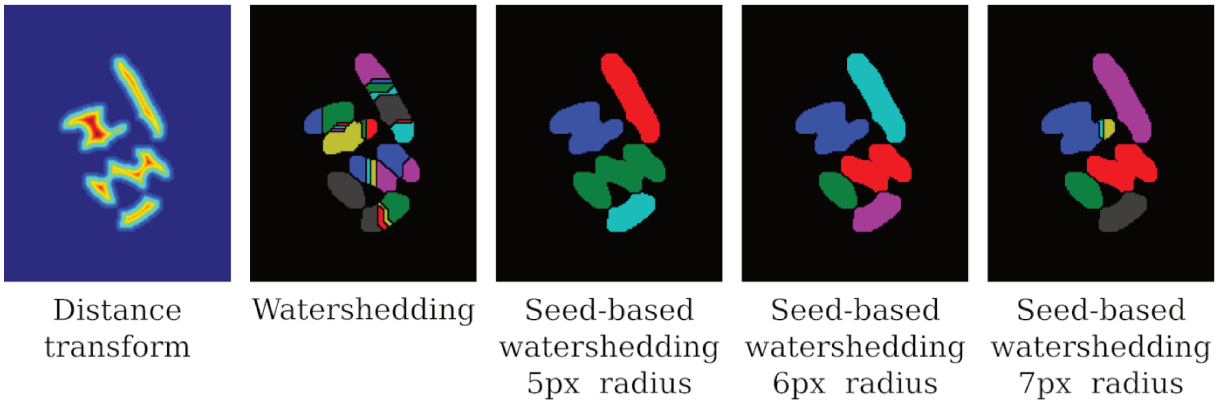

Figure 1-2 The distance transform was applied to the final image of Figure 1-1. For each "1" pixel in the image, its distance from the nearest "0" pixel is measured and attributed to the corresponding pixel in the distance transform image. Watershedding is then applied to the distance-transformed image. In the case of automatic watershedding, local minima in the input image are used as seeds. This approach often results in over-segmentation. Seed-based, or marker-based, watershedding follows the same procedure as watershedding, but instead of using local minima in the image as seeds, the user provides the seeds. Erroneous seeds will yield poor segmentation results. Here the seeds are identified through a simple procedure of marking all elements below a certain distance in the distance transform as seeds, thus removing small local minima from the seeds.

Finally, until recently the methods that yielded the best results in the domain of cell segmentation were based on active contours and deformable models of cell shape. Even though the two terms are often used interchangeably, the former describes a family of methods based on splines while the latter uses specific models of the shape of the objects they are supposed to segment. Active contours and deformable models both function in similar ways though, in that they are first applied to rough estimates of cells positions, for example from the output of a watershedding step, and are then fitted to the image (or usually one or several features of the image, like edges) through the optimization of some cost function. The advantage of this type of approach is that in the case of time-lapse acquisitions the previous cell position can be used as a 'seed' to find the new position of the cell. Another advantage is that the parameters of the shape, which not only represent the position of the cell but also what it "looks like", can be used for tracking.

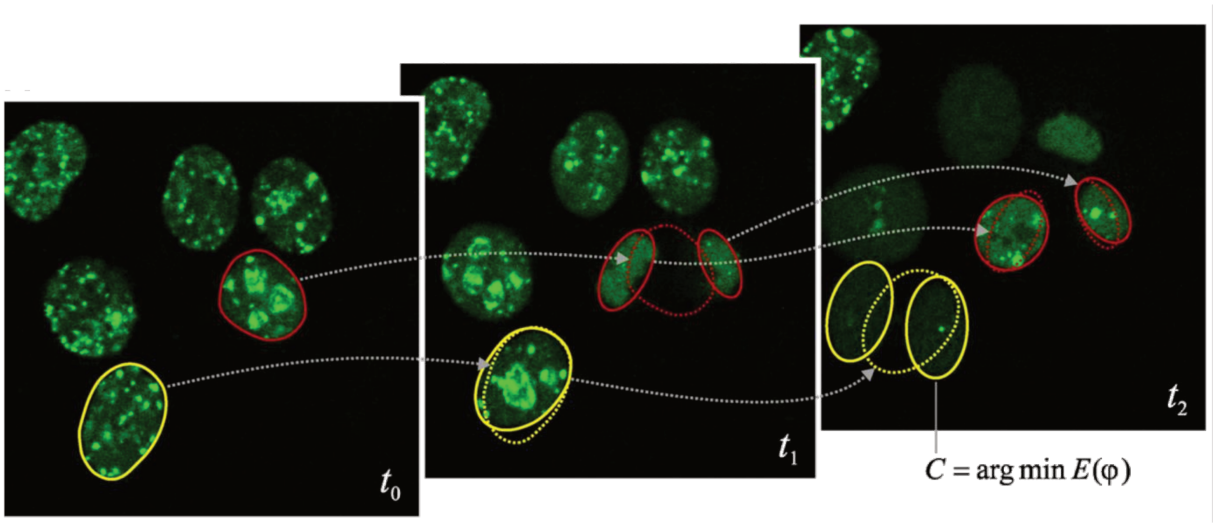

Figure 1-3 From (Meijering et al. 2009). A deformable model of the cell shape is applied to roughly-identified cell regions. Then, the identified model from the first frame is used as a seed to identify the new region the next frame. The model can also be split into two regions to account for divisions. An advantage of those models for tracking is that the parameters of the model give valuable information on the position but also the shape of the cell, which can be used to discriminate between different possible cells between one frame and the next.

An interesting (if only a bit short) review on the evolution of the field until 2012 can be found in (Meijering 2012). The conclusion is particularly interesting in that the author does not foresee the advent of new machine learning methods that are about to completely change the field.

1.2 Deep learning: the revolution.

In recent years a paradigm shift occurred in the field of image analysis with the advent of deep learning methods. In contrast to previously described algorithms, these techniques do not require strenuous algorithm elaboration for each special case of object segmentation and meticulous parameters selection after each experiment. The principle behind all these new techniques is the training of some learning algorithm with a set of manually or semi-automatically labeled images, and then the trained algorithm is used to classify images or objects in the images. Different types of deep learning approaches exist, and in this section, after introducing the concepts behind Artificial Neural Networks (ANN), I am going to briefly discuss the family of deep learning algorithm that is used the most in image analysis, and cell segmentation problems, today: Convolutional Neural Networks (CNN or ConvNet).

1.2.1 Artificial neural networks

Artificial neural networks take inspiration from the central nervous system of animals, in particular, certain parts of the brain. They consist of a network of elementary units called, unsurprisingly, artificial neurons that imitate the behavior of biological neurons: Artificial neurons receive one or more inputs (representing dendrites) and sum them to produce an output (representing a neuron's axon). The sum of each input is weighted, and is then passed through a (usually sigmoidal) activation function, that approximates a threshold function that will decide whether the neuron is activated (thus mimicking the firing of biological neurons).

Artificial neural networks were initially designed as fully-interconnected layers of neurons, and this family of network architectures is called perceptron networks, in reference to the first

work on ANNs (Rosenblatt 1957). The principle of these networks was to be applied on a set of inputs, typically images, that were pre-labeled by the experimenter. This set of labeled inputs is hereafter referred to as the groundtruth. The output of the network was compared to the groundtruth, and the parameters of each neuron are updated to minimize the measured error.

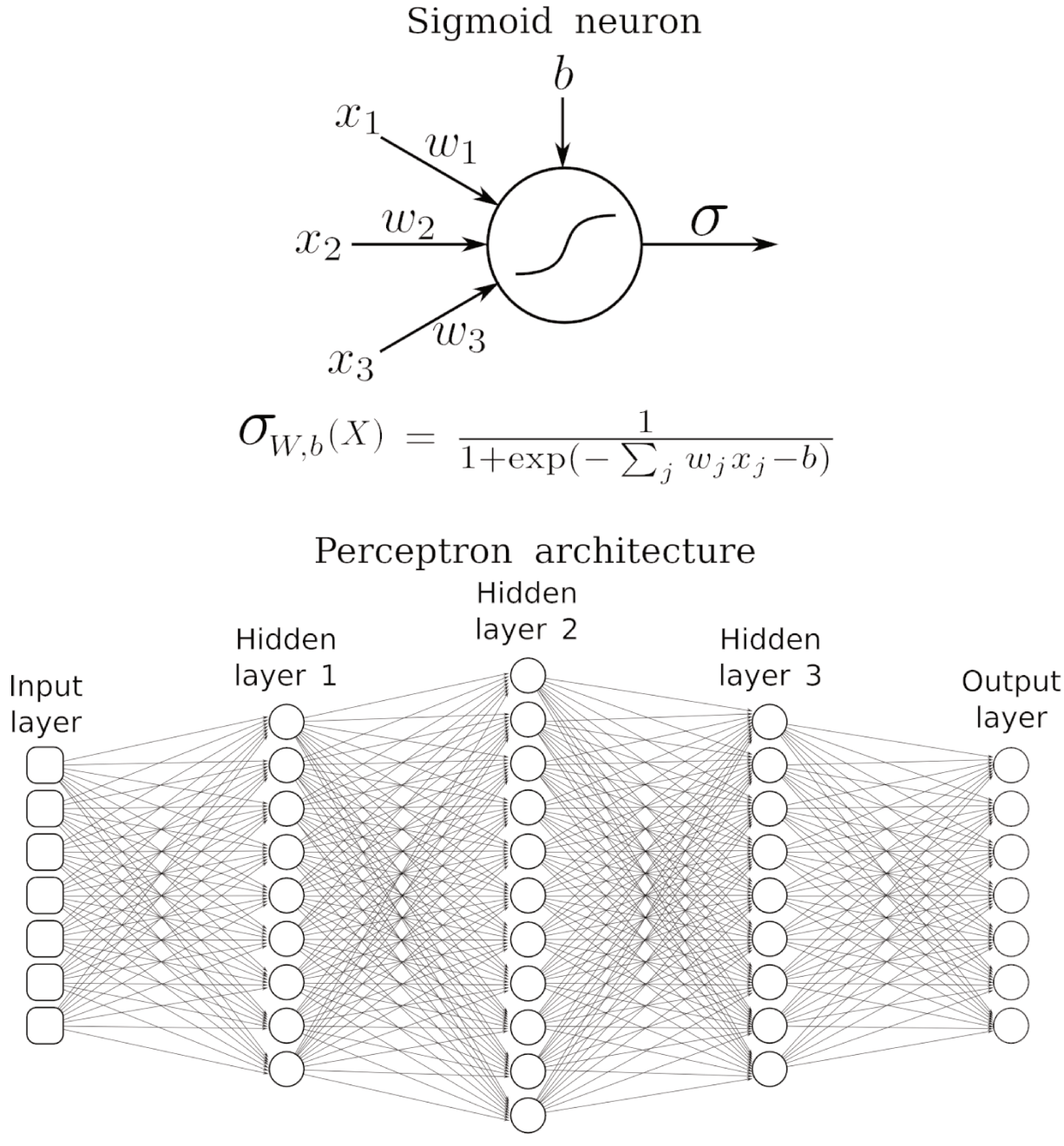

Figure 1-4 Artificial neural networks (ANN). The sigmoid neuron is used in the vast majority of ANN today. It receives n inputs from other neurons or from a data input, and multiplies them by weights ($\boldsymbol{W} = \{\boldsymbol{w_1}, \ldots, \boldsymbol{w_n}\}$). Their weighed values are then summed, with a bias parameter b. A sigmoid function is applied to the result as a sort of soft threshold to produce the neuron output $\boldsymbol{\sigma}$. In non-convolutional networks, the weights and biases of all neurons are independent, which means that each neuron has its own set of parameters.
The traditional perceptron architecture consists of one output and one input layers, and an arbitrary number of intermediate layers called hidden layers. The number of neurons per hidden layer can vary, but each neuron in any layer is connected to all neurons in the next one. While this architecture can formulate complex, high-level models of the operations to perform on the input, its number of parameters grows exponentially with the number of neurons per layer. For the simple network illustrated here, 360 parameters must be fitted to each of the inputs in the training set.

The number of parameters in this type of networks increases exponentially with the number of neurons in each layer, and one might fear that training a model with such a high number of parameters would require impractically long computational times. While this is mostly true, and one of the reasons why it took about 50 years for large-scale artificial neural networks to emerge, two things must be noted: First, ANNs require a surprisingly low number of training inputs compared to the number of parameters they use to produce satisfactory classification/identification results. This is probably related to the yet-unknown reason why they are better than other learning algorithms at generalizing their results to completely new observations (Nielsen 2015). For that reason, even small networks were able to perform reasonably well compared to other image classification/segmentation algorithm, if only a bit slower. The second, more remarkable thing is that developments on ANN training since the 1950s have greatly sped up the training process by a) an optimization of error gradient computation for each neurons layer, called the backpropagation algorithm(Rumelhart et al. 1986), b) a drastic reduction of total computation time by randomly subsampling training data, called stochastic gradient descent (Bottou & Bousquet 2007), and c) improvements in the implementation of the algorithms that made massive parallelization on GPUs possible (Raina et al. 2009). All these improvements ultimately led to the multi-record-breaking deep learning revolution: The development and training of large-scale networks was possible at last, and the large number of neurons layers made it possible to estimate intricate recognition functions. For a review of the history of deep learning as well as a discussion on recent advances in the field see (LeCun et al. 2015).

1.2.2 Convolutional Neural Networks

The CNN class of network architectures has proven especially efficient in image analysis and segmentation (LeCun et al. 2015). Convolution is a fundamental operation in image analysis and signal processing. In image processing it consists in evaluating for each pixel a weighted sum between the pixel and its neighborhood, with a mask, or kernel matrix, of weights applied to each pixel. This approach is used to implement image filters, including average, Gaussian, or gradient-type filters, amongst others. But the same principle can be applied to layers of neurons: Instead of fully connecting layers of neurons, i.e. using all the outputs of a layer as inputs for each neuron of the next layer and fitting the weights, the convolutional approach connects a small local set of neurons on one layer to one corresponding neuron on the next layer. Another property of CNNs is that the parameters for all neurons on a convolutional layer are shared, which means that different sets of weights are not fitted per neuron, but instead one “global” set of weights is fitted for each convolutional layer, and the same local operation is applied to the whole image. We can immediately see how this approach drastically reduces the number of parameters to fit. From an image processing point of view, convolutional layers also make sense since they can also mimic the actions of traditional image processing operators such as de-noising filters, gradient estimation and feature extraction, or even thresholding. Finally Convolutional Neural Networks usually consist of a number of different convolutional layers that are then used as an input to more inter-connected layers as one goes deeper in the network, up to the point where the layers become fully interconnected like in a perceptron type of architecture. These more fully connected layers usually perform more advanced, or “high-level” functions, while the less

connected convolutional layers can be seen as “low-level” operations. Not only does this type of network structure simulate the structure of the animal visual cortex, it also resembles closely the evolution of image analysis and algorithm structures used prior to deep learning: The first low-level operations like thresholding, feature detection and filtering are performed on the raw images. Then more advanced morphological operations are performed on the results of the preceding steps, and finally active contour and deformable models are applied as a high-level final operation for segmentation. The great advantage of deep learning and CNNs is that it is so-called representation learning: It takes raw data in, and with little necessary knowledge of image processing techniques or arbitrary decisions required from the user, directly outputs classification/segmentation data. Where an extensive knowledge of image processing and tedious and error-prone programming was required to produce a segmentation algorithm that would often not generalize robustly to new observations, CNNs can segment robustly an image with minor intervention from the experimenter once the training set has been created.

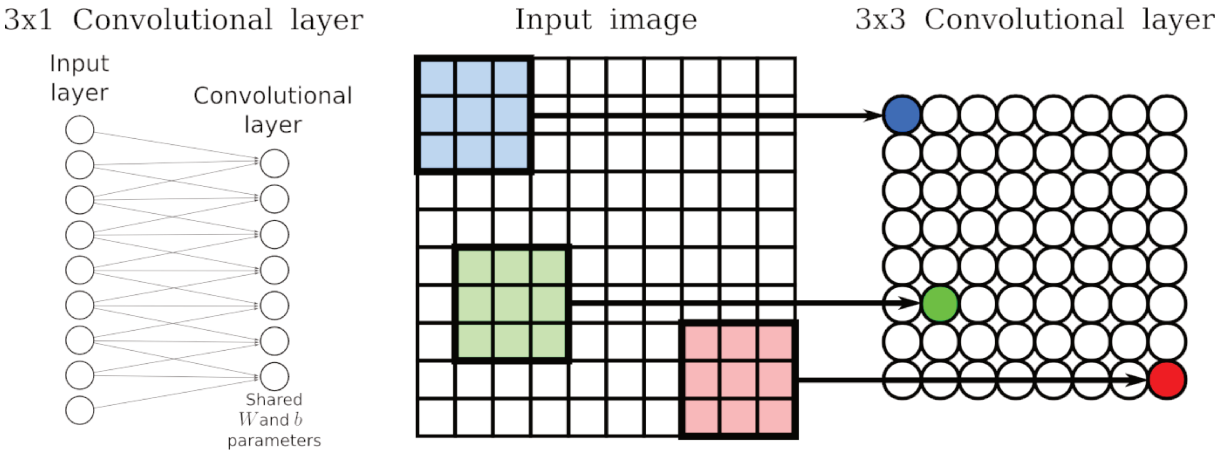

Figure 1-5 Convolutional layers. Neurons in convolutional layer are connected to input data or neurons from an upstream layer, however they are not connected to all neurons or inputs in the previous layer but to a subset of neighbor neurons. Furthermore, in each convolutional layer the parameters, like weights and biases, are shared among all neurons. This way, all neurons perform the same operation, like detecting features such as edges, circles, or textures. This structure drastically reduces the number of parameters to fit to training data.

1.2.3 U-Net

The first notable application of deep learning to cell segmentation dates back to 2012, when biological neuron membranes were identified in electron microscopy images (Ciresan et al. 2012). However the approach chosen by the authors can appear unconventional and counter-productive: Instead of training and using their deep neural network on the entire image, they apply it on a small sliding window of the image and use it to classify the center pixel of the window. While this network easily won the ISBI 2012[1] challenge for its category, it has two major drawbacks: First, the procedure is relatively slow because the network is used on hundreds of thousands of overlapping windows to classify one pixel each time, and second,

[1] The IEEE International Symposium on Biomedical Imaging is a joint initiative from the IEEE Signal Processing Society (SPS) and the IEEE Engineering in Medicine and Biology Society (EMBS). It proposes several cell segmentation challenges every year for the attendees to compete on and compare cell segmentation techniques.

[2] This family of methods is called loss functions. The most common one used with SVMs is the hinge loss function (Gentile & Warmuth 1998; Steinwart & Christmann 2008)

the choice of this reduced view of the entire image deprives the network of some of the context in the entire image that could have improved its performance. It is fairly easy to understand though why the authors chose to implement their network this way: Historically deep learning algorithms have been developed to classify images, not to segment them. The networks had been structured to take an image as an input, and identify what type of food or dog breed was in the image, but not delimitate where it was. The most straightforward way to apply these structures to segment images was then to use it on sliding windows and classify each of them as “image-with-a-neuron-membrane-in-the-center” or not.

The necessary tradeoff between localization and context remained a central problem in deep learning until very recently, when a team proposed a new network architecture that contracts from convolutional layers to a fully connected network, and then expands progressively back into convolutional layers until its output layer produces a segmented mask of the original image (Long et al. 2015). This architecture was soon applied to cell segmentation, breaking preceeding records in a number of segmentation challenges by large margins (Ronneberger et al. 2015). Another interesting aspect of their work is the procedure devised for data augmentation, namely random elastic deformations, which is important in deep learning since those networks require a large number of training samples to be able to fit the millions of parameters they are using. With this procedure they were able to train their relatively large network with only 20 original training images. Although the segmentation results are still not perfect, the U-Net architecture is a leap in cell segmentation with 77.5% correct segmentation, to be compared to the previous record of 46% for the same image set.

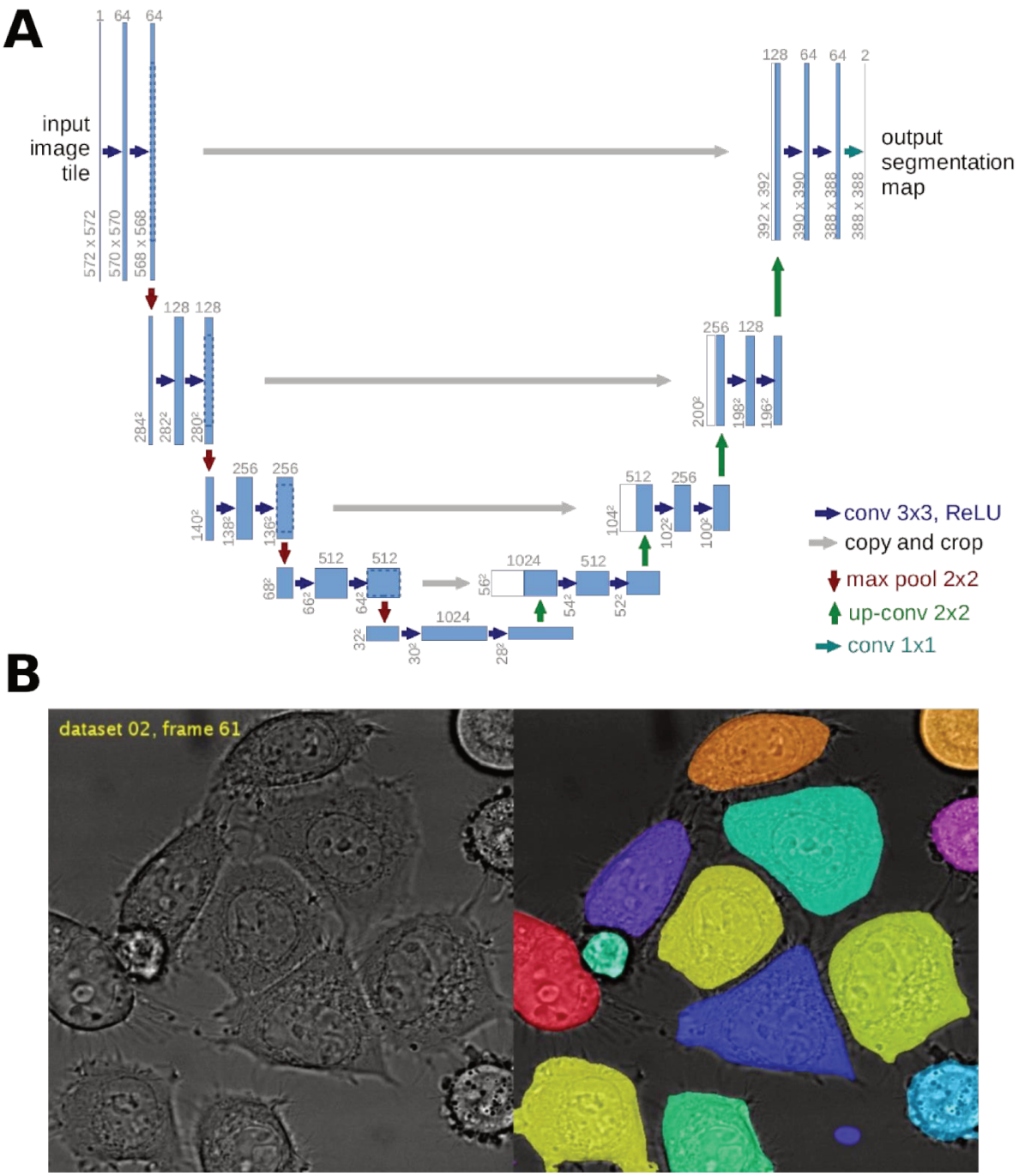

Figure 1-6 From (Ronneberger et al. 2015) . A) The U-Net architecture: 64 convolutional layers are applied as a first step to identify low-level features. The results of the convolutional layers are progressively aggregated through s-called pooling operations to progressively identify higher-level features from the lower level ones. Once high level objects have been identified, the results are propagated and expanded back into the original image format by so-called up-convolution operations. B) Segmentation result of the U-Net approach on images of HeLa cells, a notoriously difficult segmentation problem.

1.3 Hyperspectral imaging

Before I fortuitously re-discovered biology through the iGEM competition in 2011, I was studying signal processing. At that time, I was contemplating the idea of doing my PhD in hyperspectral data analysis. The field of hyperspectral imaging emerged with the apparition of digital image sensors that could sense dozens to hundreds of narrow wavelength ranges on the electromagnetic spectrum. Where a standard digital camera can sense one or three wavelengths (depending on whether they acquire 'gray scale' images or RGB images), a hyperspectral sensor can provide high-resolution spectral signatures for each pixel in the acquired image. The signatures of each pixel can then be analyzed to extract useful information about the elements in the picture, e.g. the materials they are made of. The common application of hyperspectral imaging is remote geosensing (see van der Meer et al. 2012 for a review), where high-spectral and -spatial resolution sensors are mounted on satellites or aircraft (recently drone-mounted sensors started appearing) to monitor agricultural soils, map land or prospect for resources. But interesting results were also obtained for food quality assessment (Gowen et al. 2007) , and, more in line with our interests, on hyperspectral microscopy images of living cells (Vermaas et al. 2008; Gao & Smith 2015).

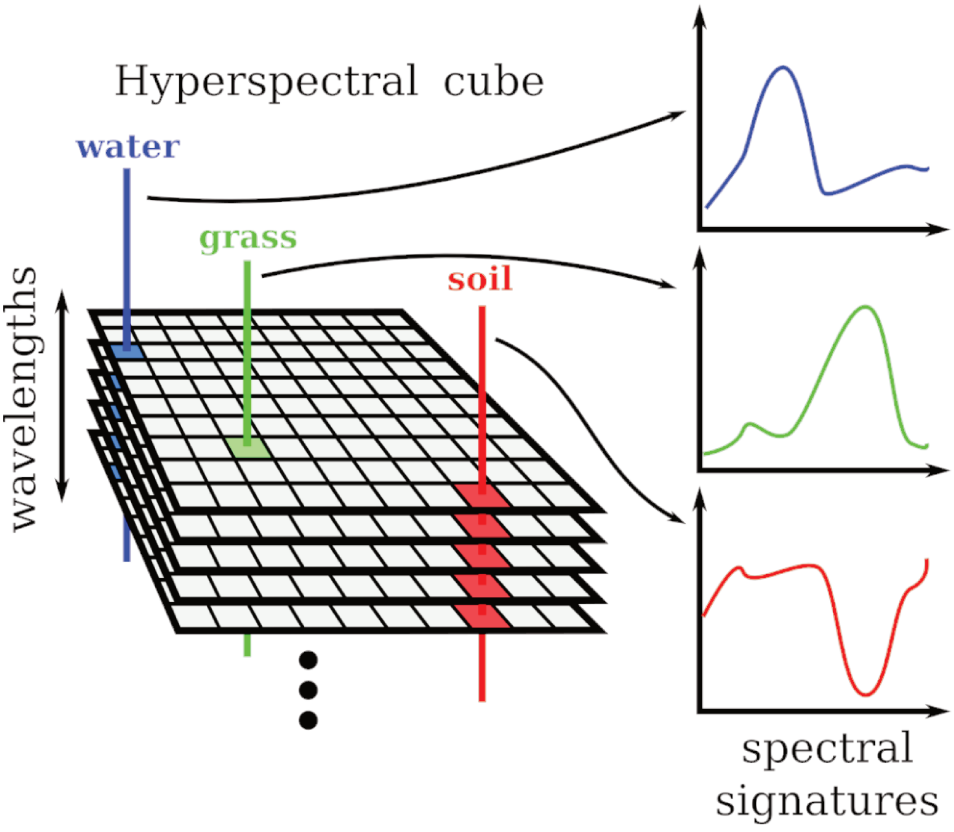

Figure 1-7 Hyperspectral data structure. Hundreds of images of an object/area are acquired at different wavelengths. For each pixel in the compiled "cube", different types of objects, matter or chemicals produce different spectral signatures.

Although hyperspectral sensors were developed in the early 1990s, it was not until the mid-2000s that efficient data analysis methods were developed and that the main analysis framework was established (Melgani & Bruzzone 2004). Hyperspectral data analysis resisted the revolution of deep learning for a surprisingly long time, with deep learning methods

exceeding the performance of the more traditional support vector machines (SVMs) only recently (Chen et al. 2014; see Zhang et al. 2016 for a review).

1.3.1 Support Vector Machines

Spectral signature identification is a classification problem: Whether some region of the image represents a type of soil, buildings or cellular organelles boils down to a classification choice. Out of the myriad of classification algorithms developed for different statistical problems, SVMs have long offered the best results for hyperspectral data analysis, and are today second only to recently developed Deep Neural Networks.

SVMs, Support Vector Machines, have been around since the 1960s (Vapnik 1963) and rely, in their simplest form, on identifying a hyperplane that can separate (with a maximal margin) all n-dimensional observations in a provided training set of m observations. In less obscure terms, SVMs (or, more precisely, linear SVMs) identify and optimize linear boundaries between classes of training points. To reduce computational cost, instead of estimating the coordinates of the optimal hyperplane by using all datapoints, support vector machines use a subset of the datapoints that lie at the boundary between classes, which are called support vectors. It has been demonstrated that the optimal hyperplane for separating the two classes and maximizing generalization performance is the one maximizing the margin between the itself and the support vectors of the two classes (Vapnik & Kotz 1982).

Once the optimal hyperplane has been estimated, when new data is acquired and needs to be classified the algorithm evaluates whether a new observation is situated on one side or the other of this boundary. The result, usually called a prediction, determines whether this new observation is part of one class or another, but it also provides an estimate of how "certain" that classification is: The Euclidean distance from the estimated boundary is usually extracted from the algorithm as a classification score, which can be useful a posteriori to assert the validity of segmentation results.

Of course, linear classifiers like linear SVMs are often not adapted to real-life data, in which the different classes are often spread in complex intertwined regions of their feature space. A workaround for this problem was developed in the early 1990s (Boser et al. 1992) and made SVMs the go-to classifiers for almost 20 years. The idea was not only to transform the feature space of the observations into another, where the differences could be separated linearly, but also creating new dimensions from the ones existing if necessary. This method is called the kernel trick, because the new dimensions are created through so-called kernel functions, which allow the SVM classifier to operate in a high-dimensional, implicit feature space without ever computing the coordinates of the data in that space. Instead, the dot products that are normally used on the data to compute their classification score in linear SVMs are replaced by the kernel functions. This operation is computationally cheaper than the explicit computation of the coordinates, and makes the separation of nonlinear data possible. A lot of kernel functions have been used in the literature, but in the field of hyperspectral imaging Gaussian radial basis functions typically yield higher performance with respect to speed and accuracy (Mountrakis et al. 2011):

$$K(\boldsymbol{x}, \boldsymbol{x}') = exp(-\gamma \|\boldsymbol{x} - \boldsymbol{x}'\|^2), \quad with\, \gamma > 0$$

$\|\boldsymbol{x} - \boldsymbol{x}'\|^2$ can be recognized as the squared Euclidean distance. γ is estimated by the SVM algorithm to best separate the different classes.

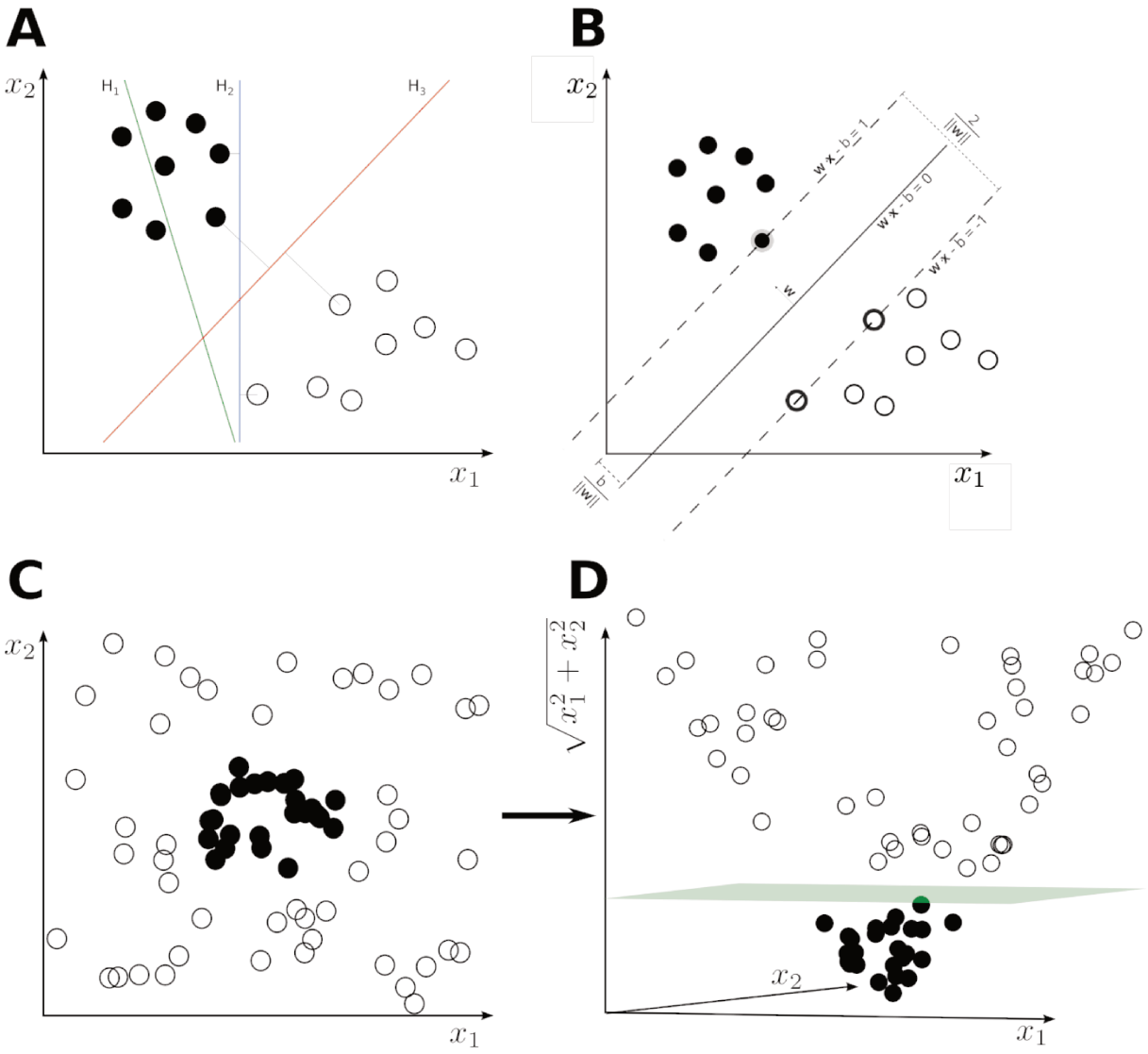

Figure 1-8 The principles of SVMs. A) The hyperplane optimally separating two classes of 2-dimensional data is H3. B) The optimal hyperplane is the one optimizing the margin between itself and the two classes of data. C) In most real-life cases, the classes cannot be separated by a linear boundary. D) To work around this problem, SVMs artificially create new dimensions from the existing ones to be able to separate them linearly.

Another problem that often arises is over-fitting, which is often worsened by the fact that training data can contain errors. Although there is no commonly accepted method to ensure that over-fitting does not occur, and in general the approach boils down to fancy trial-and-error procedures, the problem of mislabeled elements in training datasets has a more agreed-upon solution. The underlying principle is that instead of finding a perfect separation between all elements of each class, a 'soft-margin' approach is used in which misclassified elements

are given a misclassification weight depending on some mislabeling score[2]. A tradeoff parameter between the sum of weighed misclassifications and the size of the feature space is set by the experimenter, and is typically determined empirically (Steinwart & Christmann 2008).

Finally, the curse of high-dimensionality is a problem in SVMs too, and the number of features, or wavelengths, in hyperspectral data reduces the performance of the classifiers (this behavior is known as the Hughes phenomenon (Hughes 1968)). The common procedure is therefore to reduce the dimensionality of the feature space while minimizing information loss via so-called feature extraction procedures. Although a number of algorithms for feature extraction has been used for specific applications, one of the first and most straightforward algorithms to be used and that dominated the field in the early years of hyperspectral imaging is principal component analysis (Hotelling 1933) and a number of its variants(Mountrakis et al. 2011).

The analysis of hyperspectral data with SVMs was first suggested in the late 1990s (Gualtieri & Cromp 1999) but the first paper to compile all the improvements on SVM performance mentioned above and to establish SVMs as the undisputed hyperspectral data classifiers for more than a decade appeared only five years later (Melgani & Bruzzone 2004). For an in-depth technical discussion on SVMs see (Steinwart & Christmann 2008). For an extensive review of SVMs and their ameliorations in hyperspectral data analysis, see (Mountrakis et al. 2011).

As a side note, SVMs have also been used for microscopy image analysis, and cell segmentation in particular. But because SVMs are primarily classifiers that must be wrapped in several layers of image analysis and feature extraction to transform input images into a myriad of observations to classify, their usage remained largely confined to niche applications, like white blood cell identification (Ramoser et al. 2005; Osowski et al. 2009).

1.3.2 The framework of hyperspectral imaging

To summarize, the typical workflow of hyperspectral data analysis with support vector machines is as follows:

1. Training
   1.1. Hyperspectral training and evaluation datasets are acquired, and experts label the different classes in it.
   1.2. A feature reduction step is applied, thus drastically reducing the number of dimensions while preserving a maximum of the original information in the data.
   1.3. A number of SVM are fitted to the labeled training set after feature extraction, each with different sets of parameters (kernel function, misclassification/dimensions tradeoff, maximum number of support vectors…)

[2] This family of methods is called loss functions. The most common one used with SVMs is the hinge loss function (Gentile & Warmuth 1998; Steinwart & Christmann 2008)

   1.4. The classification performance of each SVM is then assessed on the evaluation set, and the best result is used for identification, or the experimenter can go back to step 3 to try to improve performance based on those results.
2. Prediction/Production
   2.1. Acquire new unlabeled hyperspectral data.
   2.2. Transform data into the new feature space from training step 1.2.
   2.3. Classify the signatures of each pixel into the classes from step 1.1.
3. (Optional) Use classified images for target recognition/detection (Vehicles, infrastructures, forests…) with traditional image segmentation techniques.

This procedure inspired the segmentation algorithm I am going to present in this chapter, which replaces the spectral signatures from hyperspectral imaging with so-called "focal signatures" that are acquired by imaging focal plans above and below the cells' imaging plane. The focal signatures are the pixel illumination levels acquired at each focal plane, which vary depending on the element of the in-focus image. Basic optics suggests that these changes in signature between the different elements depend on the shape and composition of the object being observed, thus making object class identification possible in theory.

2 Z-stacks and focal signatures

While working on autofocussing techniques to try to obtain stable time-lapse movies and minimize measurement noise, I had to acquire a number of Z-stacks to use as inputs for the algorithms I was developing (see section 3.1.7 of chapter 2). So-called Z-stacks are stacks of images acquired at different focal planes, usually around the focal plane of interest where the cells are most discernable. If we adopt the usual convention of naming the 2 dimensions along which the images are taken x and y, the axis along which the focal-plane stack is acquired is logically labeled z, hence the name Z-stack. The development of autofocussing techniques was an ongoing task throughout my PhD, but I rapidly realized that, instead of trying to simply identify the best frame in the stack according to some criterion (image sharpness, histogram levels…), results that were both more precise and accurate would be achieved by measuring the same feature in all stacks in the image and then comparing it to some reference "signature" acquired at the beginning of the experiment.

Around the same time, I was also trying machine learning algorithms for cell segmentation. I did not know then of the results acquired with DNNs on neurons membranes (Ciresan et al. 2012), and was trying to use methods typically used for pedestrian identification in images based on SVMs and HOG transforms[3] (Dalal & Triggs 2005). While this method gave satisfactory results on another project I was collaborating on for ant segmentation, it became rapidly clear that the method was not adapted to bacterial cell detection. Somewhat

[3] Histogram of oriented gradients, or HOG, is a feature extractor for 2D images that splits an image into overlapping windows and computes for each of them a histogram of the gradient for different orientations.

paradoxically, the complex shapes humans or ants can adopt made them more easily discernable from other objects or between themselves, while the simple shape of bacteria coupled with the quality of the images at this level of magnification made the approach irrelevant.

But eventually one day, in a perfect illustration of the Eureka phenomenon (Asimov 1971), while precariously riding home on my bicycle after one too many beers and a frustrating number of desperate attempts at segmenting cells with dubious combinations of SVMs and feature extractors, I finally thought of using the information that I knew could be found in Z-stacks to identify cells.

After acquiring new Z-stacks with hundreds of frames[4] and analyzing what would later be called the focal signature of specific objects in the stack, it became evident that not only did the "signature" of some criterion along the Z-axis was specific to the object that was being observed and hence identifiable (see autofocussing method in section XX of chapter 2), but also that the signature of each z-pixel[5] in the stack was closely related to the class of object they were part of. The acquisition process as well as the typical shape of the signatures is detailed in Figure 2-1.

The Z-stacks were acquired on a number of different automated microscopes, two of which, referenced here and amongst labmates as Mustard and Marple, are in our lab and are based on the IX71 and IX81 Olympus chassis and feature a variety of different equipment. The most notable piece of equipment on the IX71-based microscope Mustard is the piezo-driven motorization of the objective nosepiece along the Z axis, which allows for precision positioning with a resolution in the tens of nanometers. For that reason, and also that Mustard was the microscope on which I implemented the control platform and was the most familiar with, the majority of acquisitions were performed on this microscope. Unless otherwise specified, the Z-stacks were acquired on it. Two other microscopes were used by collaborators to acquire Z-stacks for evaluating the performance of the algorithm, one located in the lab of the Biologie et Dynamique des Chromosomes group of CNRS UMR 7212 at Hôpital Saint-Louis in Paris, and another located in the lab of the Molecular Microbial Ecology group at the Department of Environmental Systems Science of ETH Zurich. These two microscopes will be referred to here as St-Louis and Zurich.

The stacks were acquired with varying distance steps (z-steps) between each frame, from 100nm to 1µm. The original number of frames per stack was 100, with 100nm steps between all stacks, but eventually it became clear that much lower numbers of frames were sufficient to achieve satisfactory signature classification performance for robust segmentation, although higher frame numbers increased the performance marginally (see section 4.2.1). I also

[4] I will refer to single images acquired at specific focal plans in a Z-stack as frames.
[5] I call a z-pixel the vector of intensity values along the Z axis that are acquired in each frame of the Z-stack for a specific set of pixel coordinates (x, y) along the X and Y axis. The term might be used interchangeably with focal signature, but is generally preferred here when referring specifically to raw data, while the term signature will be used in a more conceptual way and regardless of the feature space it is represented in.

observed that a non-fixed z-step, i.e. non-linearly spaced acquisitions along Z, yielded better results.

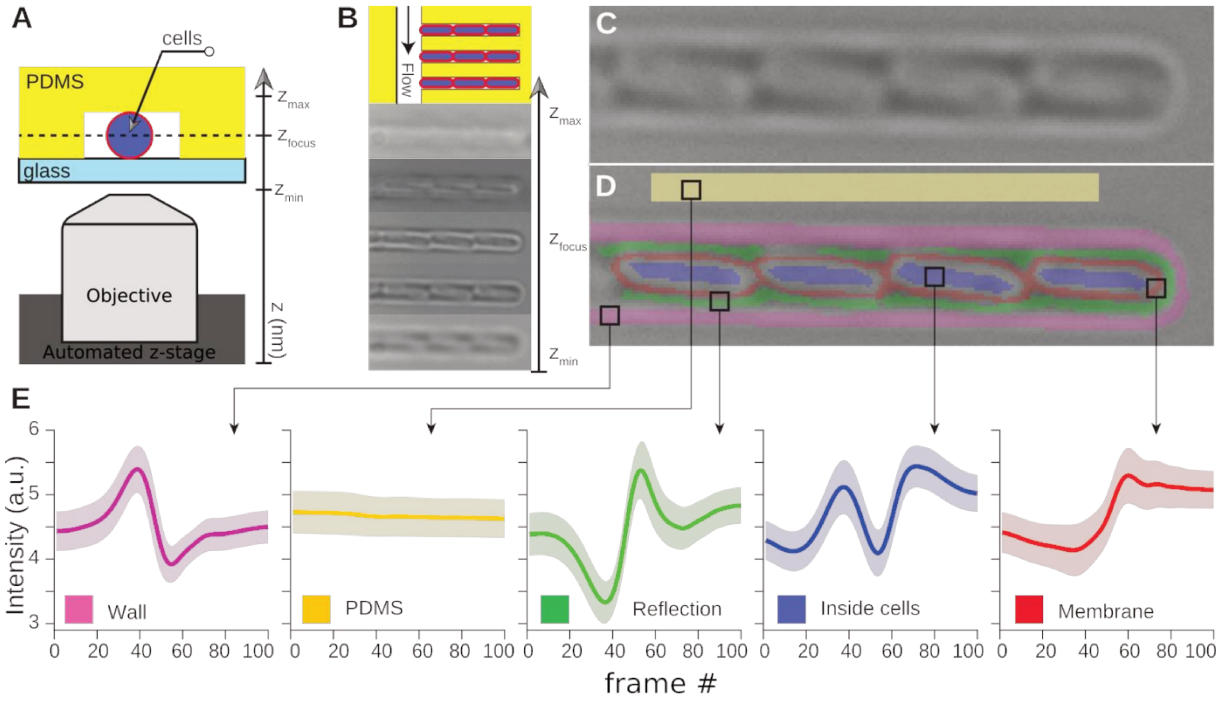

Figure 2-1 Acquisition principle and focal signatures. A) The specimen is placed on a microscope, and an automated z-stage (preferrably piezo-driven) is used to acquire stacks of frames above and below the focal plan of the cells. B) In our mother machine setup, the intensity of the pixels changes the focus. C) Close-up of a chamber. D) Different regions are manually identified by the experimenter. E) The mean and standard deviation of the signatures for each class are extracted. The signatures are visibly different depending on the type of z-pixel.

3 Training

Based on the observation that different classes of z-pixels seemed discernable from each other, I started working on building training sets and fitting SVMs to classify pixels. In this section I describe the procedure for constructing training sets, normalizing the data, training SVMs and finally evaluating the performances of the trained SVMs.

3.1 Training set and evaluation set construction

I developed a graphical user interface (GUI) in Matlab to simplify training set construction. The interface allows the experimenter to build a training set based on any number of Z-stacks, and allows for the creation of any number of different classes.

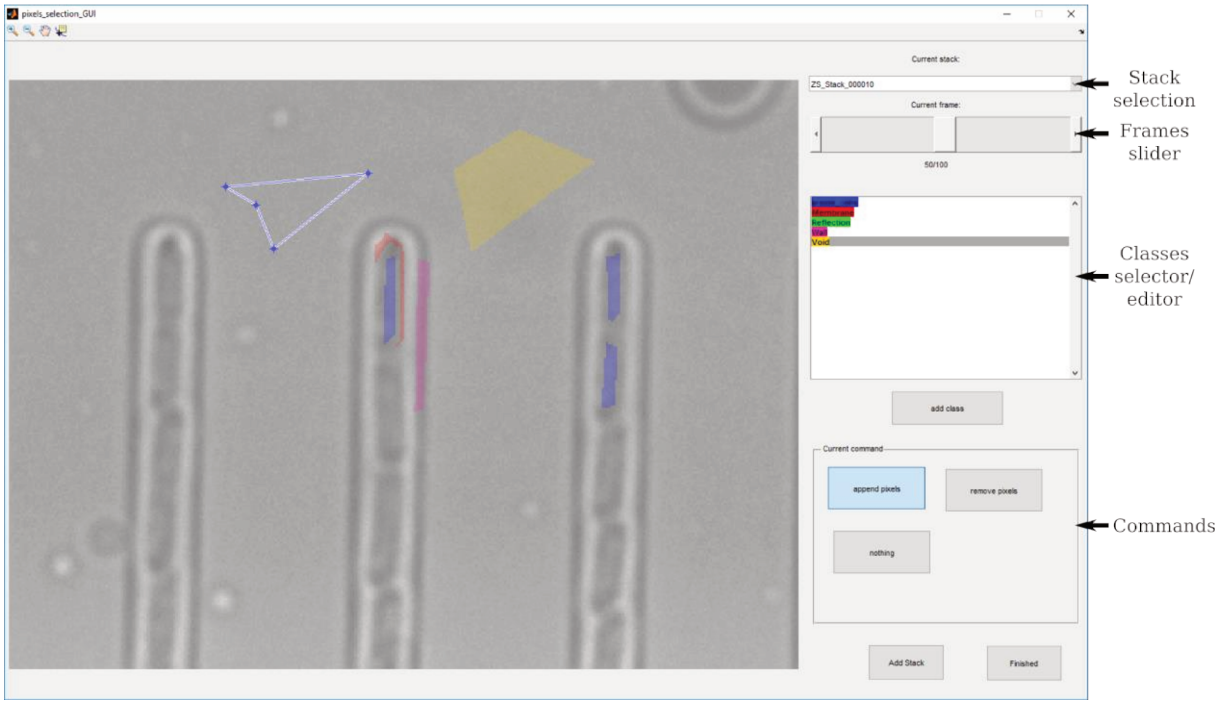

Figure 3-1 Graphical user interface (GUI) for training set construction. Several Z-stacks can be loaded at once, and any number of classes can be created and edited. The user can add and remove regions in the images that will then be extracted from the stacks and concatenated into a training set and an evaluation set.

When constructing new training sets, the experimenter must be careful about several things:

First of all, although I used a hinge loss function to make the training more robust to erroneous outliers in the dataset (see section 1.3.1), the experimenter must take extra care when creating the training set of not mislabeling data. Although this might seem obvious, it is easy for the experimenter to start labeling data a certain way on one part of the set and end up labeling it in another way when they reach the end of the tedious process of creating training datasets. A typical case is, when labeling cell membranes and cell cytoplasm in bacteria, the limit between the two classes is hard to visualize and by a large portion arbitrary. It is therefore preferable to leave a wide margin between the two classes when identifying z-pixels to ensure that the two will be well-separated in the feature space. At the very least, when establishing a new set, the experimenter/expert should check periodically how they did their labeling at the beginning of the process to avoid drifting too much. A major part of the classification performance will depend on the quality of the training dataset, and while initial trials might be done on roughly labeled datasets and still produce decent results, the performance would be improved by careful training set construction.

3.2 Data normalization and focus invariance

Two problems arose early in the development of this segmentation procedure: The shifting and scaling of the input data that could occur between different stacks or different experiments. The scaling is a known problem in hyperspectral imaging: between one acquisition and the next, light intensity can vary and the intensity measured for each wavelength can vary. The same problem happens in our case: Light intensity can change during an experiment, and will almost always change between different experiments. To normalize our data we perform a standard histogram equalization procedure with 1% loss on

the histograms of all the training stacks and, once the SVM were trained, on all new Z-stacks as well, to ensure that all data were spread over the same intensity scale.

The second problem, shifting, does not happen in hyperspectral data. It consists in a shift of the signatures along the Z axis caused by autofocussing error. Because the training Z-stacks are acquired around the focal plan at which the cells are precisely in focus, the frames are taken always at the same distance from the central in-focus frame. That means that the features of the training signatures are acquired at specific points in Z around the in-focus point. However autofocussing is not always perfect, or the specimen observed can be tilted, and this can lead to a shift of the signature in Z. In this case, the shape of all signatures in the stack will appear drastically different to the SVM.

The physics of hyperspectral sensors make such a shift impossible in this type of data, so unsurprisingly there were no solutions to be found in the literature. For a time we experimented with pre-processing methods to re-shift the data in such cases or iterative prediction steps to select the best result. But in the end, a more elegant solution was to simply train the SVM on shifted data: The stacks for training at a high resolution in z were acquired, and then subsampled for the frames we needed in the stack to train for in-focus data. Then, an artificial shift in Z would be applied by re-subsampling for the frames with a shift in Z. Subsamples were acquired for an arbitrary shift range and the SVM were trained on all those data: the in-focus data, but out-of-focus data as well. This procedure not only made the z-pixel classification more robust to autofocussing errors, up to the point where the autofocus could be so wrong that cells could not be identified anymore in the supposedly in-focus frame and still the z-pixels could be identified properly, but it also made the results more accurate when there were no autofocussing errors. We think that the latter is because the shifting procedure works as a sort of data-augmentation method that made the SVMs better at generalizing to new observations. Another advantage of this method is that, except for the shifting range that is decided at the beginning and that can be seen as the "range of robustness to autofocussing error", the algorithm does not require any arbitrary parameter to be decided upon by the user or any knowledge of signal or image processing for correcting focusing errors.

A third problem, which we are currently working on but that only arises when lamp alignment was not performed properly prior to the acquisition, can be referred to as 'skewing'. It consists in a shift of the objects in the image along the X and Y axes when the frames are acquired along the Z axis. In other terms, images are not well-aligned between focal plans. The shift appears to be linear, i.e. the image is shifted by a constant vector (x_s, y_s) every fixed z-step. We still have not found a way to robustly and automatically identify the shift to correct it.

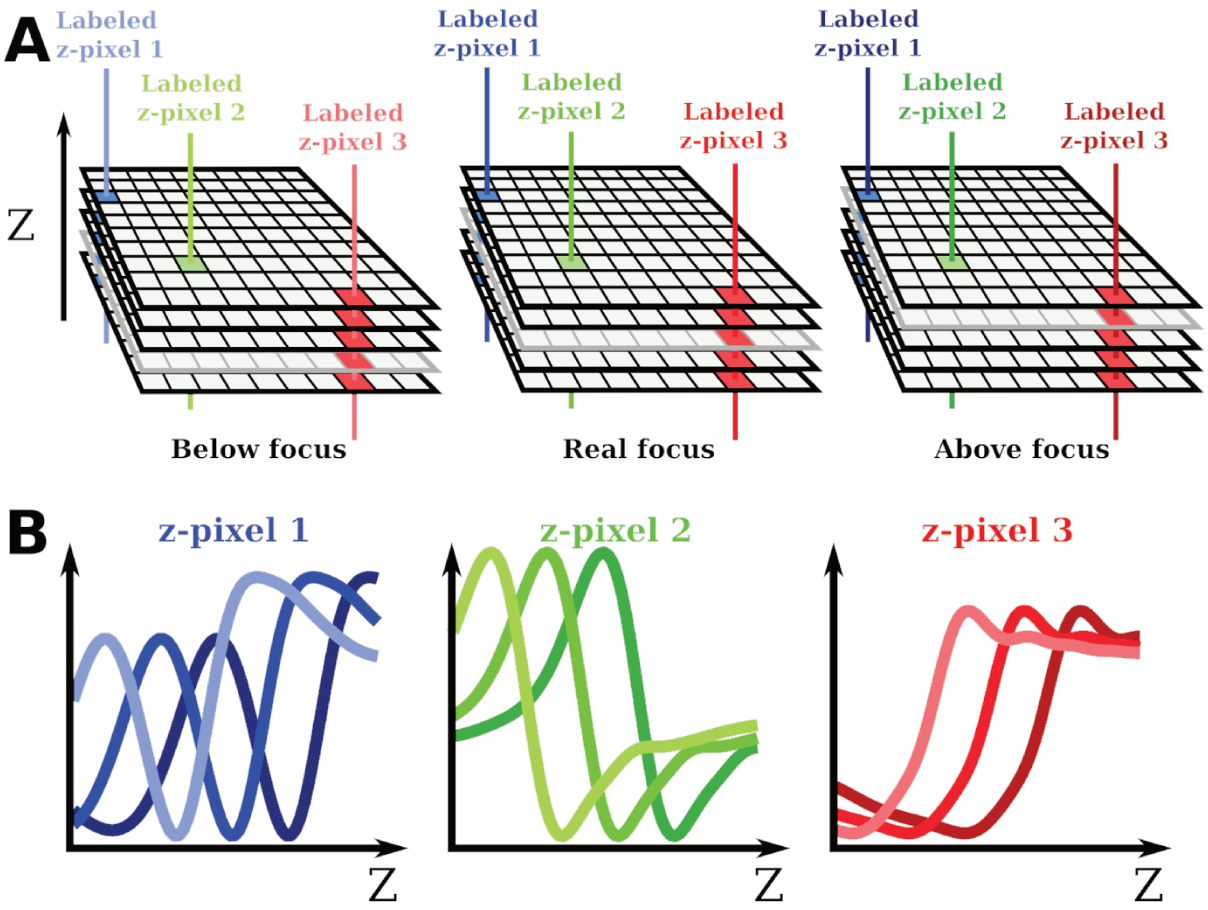

Figure 3-2 Focus shifting. A) The signatures extracted from the stack are artificially shifted, by changing the reference of the in-focus frame (in grey). This procedure mimics what happens when erroneous autofocussing shifts the acquired stack. B) Shifted signatures: The SVM will be trained on shifted variants of same signatures.

3.3 Feature extraction

As discussed in section 1.3.1, SVMs perform poorly in high-dimensional input spaces, and the dimensionality can be reduced by a number of feature extraction algorithms. A common feature extraction algorithm in hyperspectral imaging is principal component analysis (PCA) and it is the approach we used here, but other approaches could yield better results.

PCA is performed by computing the set of orthogonal, unit eigenvectors of the covariance matrix of the data and transforming the input data into this new space. In other terms, the set of orthogonal axes along which the input data varies the most are computed as linear combinations of the original axes that are called principal components, and data is represented in this new base. One of the properties of PCA is that the first component of the new base has the largest possible variance, and then the second has the largest possible variance of what remained and so on… This ordered selection of orthogonal components maximizing variance makes it possible to easily choose an arbitrary number of dimensions in the principle components space that maximizes the variance that could be accounted for.

In our application, we performed PCA on the training data to obtain the principal component coefficients, also known as loadings, which allowed us to transform our data into the principal components space. The same coefficients are applied later to transform the unlabeled data in Z-stacks into the principal components space for prediction. However, in order to reduce

dimensionality, only a subset of the first N' principal component dimensions are used to represent the data that are used to train the SVMs and are later predicted upon. As a rule of thumb, we chose N' such that the amount of variance accounted for by the first N' components was at least 95% of the original variance, except if that meant N' was greater than 20, to ensure we would not train the SVMs on too high-dimensionality data. However for improved performance one should compare the prediction results of the fitted SVMs with different numbers of components to see which performs best for a specific case.

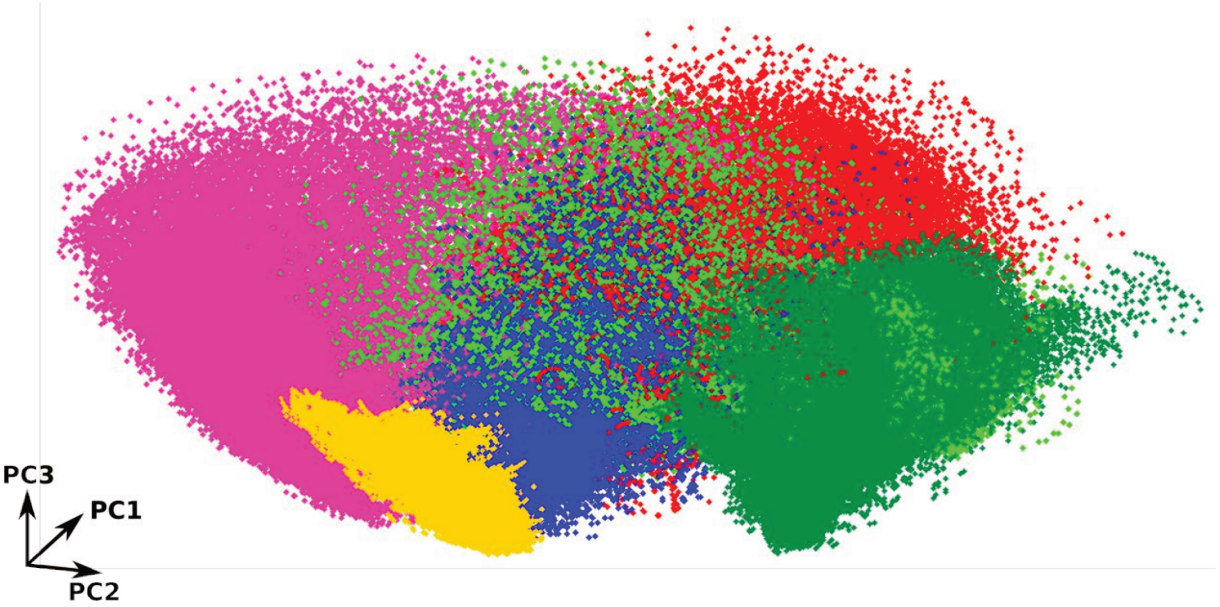

Figure 3-3 Principal components. The signatures from the training set are tranformed into their principal components space. Each point represents a labeled signature in its first 3 components. The different colors represent the different lebels of the signatures. Already in the 3 first components, class clusters of datapoints can be observed.

3.4 SVM fitting

Because SVMs are binary classifiers, the most common way of handling multiclass data identification is to fit an SVM to each class against all others and then to run all SVMs on each signature at the prediction step and take the best result. While other, more computationally efficient techniques based on hierarchical classification trees exist (Yuan et al. 2006; Cesa-Bianchi et al. 2006), this method (known as winner-takes-all SVM, or WTA-SVM) has the double advantage of providing a classification score for each class, which can be used for segmentation as we will see in section 5, but also not require the experimenter to establish a classification tree, which can be a difficult problem. Other non-hierarchical methods than WTA-SVM have been developed over the years. They rely on voting and can give better results, but are usually less computationally efficient since they require a number of SVMs that grows exponentially with the number of classes (Duan & Keerthi 2005; Mathur & Foody 2008).

Although our general approach was winner-takes-all SVMs, which means that z-pixels are classified as whichever class obtains the highest classification score, we will see in section 5.2 that the SVM results for each class can be used in combination with spatial rules to improve the classification.

The SVM implementation was based on the Matlab fitcsvm package, which has the advantage of featuring automatic hyperparameter optimization, with a Gaussian radial basis function as kernel and a hinge loss function (see section 1.3.1). To reduce training time, the fitting of the SVMs for each class was parallelized. The fitting was performed until all SVMs converged (typically 15 minutes to 1 hour). The results were then compared to the evaluation set to estimate the performance of the classification against hyper-hyperparameters such as the number of frames used or the dimensionality of the final feature space.

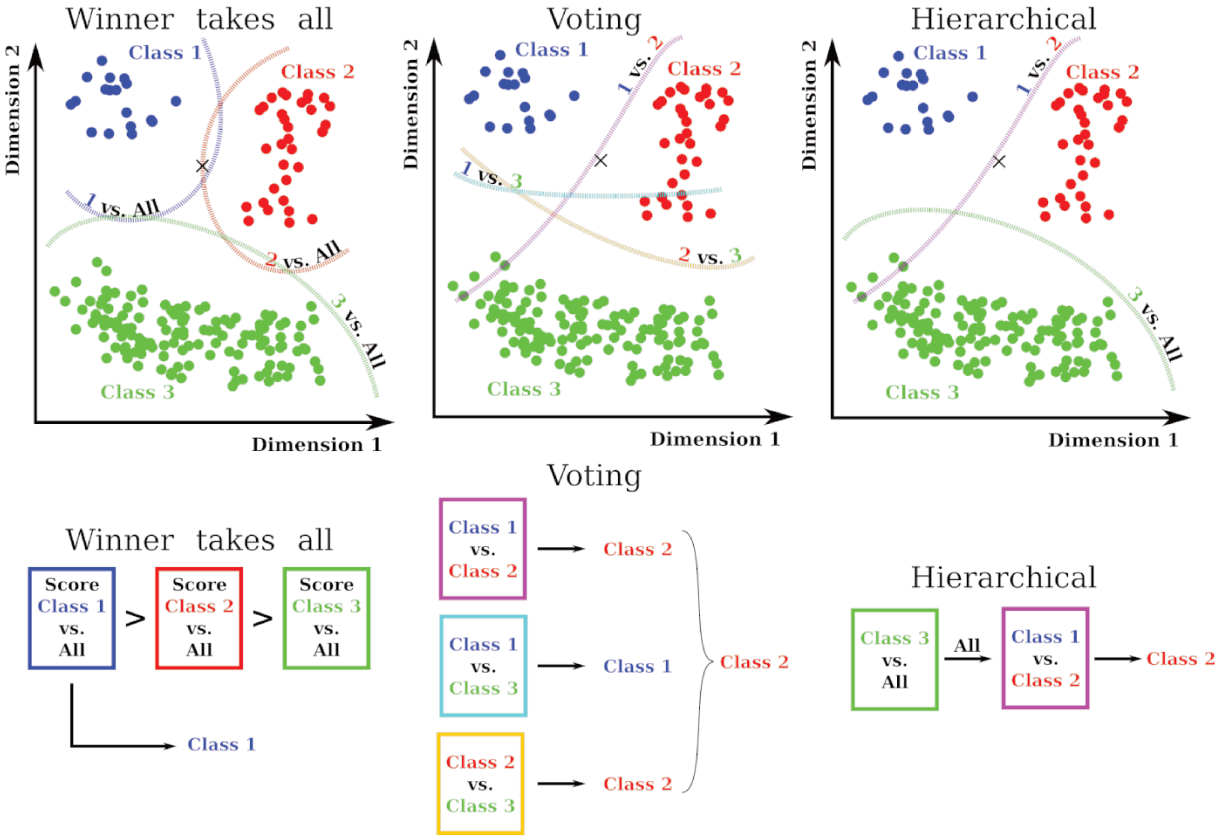

Figure 3-4 Multi-class SVM approaches. Winner-takes-all (WTA) SVM fits SVMs to each class against all others (dashed lines). When a new observation (black cross) is evaluated, the SVM that produces the highest score wins, and the observation is attributed to the corresponding class. Voting-based multiclass SVMs fits SVMs to all possible combinations of 1 class against another. New observations are evaluated by each one of them, and the class with the highest number of votes wins. While this approach usually yields slightly better prediction results, the number of SVMs to fit to the data rises polynomially with the number of classes and can dramatically increase the computation time. Hierarchical approaches partition the classification procedure into a decision tree, and fit the corresponding SVMs. New observations are evaluated sequentially until they are labeled as a specific class. While this approach can reduce computational time, it requires a priori knowledge on the structure of the data.

3.5 Evaluation

After a labeled set was constructed, we typically divided it into 2 parts by random data subsampling: The first part, consisting of 90% of all datapoints was the training set that was used to train the SVM. The remaining 10% were used as an evaluation set. The evaluation set was used to compare the results of the SVMs with different parameters (Number of frames in the stacks, shift-correction range, number of principal components…) by running the classification of the SVM on the evaluation set and then comparing it to the labels established by the user.

4 Classification

4.1 Class prediction and confidence map

Once a set of SVMs has been trained, it is used for 'production', or to identify new stacks for cell identification and segmentation.

When a new stack is acquired, it is first scaled to the same dynamic range as the training stacks (basically the histogram of the entire stack is equalized over the maximum range of the training data type, which is usually uint16)

It is then transformed into the principal components base by multiplying the z-pixels by the principal coefficients from the feature extraction step described in section 3.3.

The SVM is used on the transformed data, and for each z-pixel a set of classification scores that correspond to each of the classes the SVMs were trained for is produced. Because all z-pixels are independent from each other and their classification does not depend on the classification of other pixels, parallelization of the prediction process is trivial. Unfortunately Matlab does not feature a GPU-accelerated SVM library, which can decrease processing time by two orders of magnitude (Catanzaro et al. 2008). Parallelization can still be performed at the CPU level. We used a Dell Precision T7910 with a 20-core Xeon ES-2650v3 processor, that allowed us to significantly decrease processing time, down to a bit more than a minute for an entire stack of 100 frames and 1392x1040 pixels per frames. Processing time can of course be reduced by defining regions of interest in the image and not classifying the rest of the image.

The SVMs raw scores are also used to establish a confidence map of the classification in the image. To produce this so-called confidence map, we apply the softmax function to the SVM classification results for each result, and pick the score of the best class as the classification "confidence". The softmax function is commonly used in deep neural networks and is used to normalize classification scores. It is defined as:

$$s_i(\boldsymbol{C}^p) = \frac{e^{c_i^p}}{\sum_k e^{c_k^p}},$$

With $\boldsymbol{C}^p$ being the vector of the SVM classifications scores $(c_1^p, \dots, c_m^p)$ of the m SVMs for each of the m classes in the training set, for a z-pixel p of coordinates (x,y). $s_i(\boldsymbol{C}^p)$ is the softmax function evaluated for class i on the classifications scores of z-pixel p. We call $\boldsymbol{S}^p = \left(s_1(\boldsymbol{C}^p),\ \dots\ s_m(\boldsymbol{C}^p)\right)$ the vector containing the m softmax evaluations of each class for a z-pixel p. The softmax function restricts the scores to the interval [0, 1], where the initial SVM classification scores could be any element of $\mathbb{R}$. The second interesting property of the softmax vector is that the sum of its elements $\sum_k s_k\,(\boldsymbol{C}^p)$ equals one. For those reasons the $\boldsymbol{S}^p$ vector can be interpreted as a probability distribution, and $s_i(\boldsymbol{C}^p)$ as the probability that pixel p is of class i. The $\boldsymbol{S}^p$ vector can also be seen as a sort of composition of the signature of z-pixel p in terms of the archetypal signatures for each of the classes. The confidence map M is

constructed for any of its pixel of coordinates (x,y) from the softmax vectors of the corresponding z-pixel p as the highest softmax score for that z-pixel:

$$M(x,y) = \|\boldsymbol{S}^p\|_\infty$$

The confidence maps can be seen in the results in section 4.2 and inform the experimenter or a downstream algorithm on the reliability of different parts the classified image. The class map M_i defined as $M_i(x,y) = s_i(\boldsymbol{C}^p)$ of a specific class i can be used to improve segmentation as we will see in section 5.3.

4.2 Results

The results presented here are classification results obtained on different organisms. In section 5 I present how these results can be used to segment cells. Unless otherwise specified, the results were obtained on different stacks than the ones used for training the SVMs

4.2.1 *Escherichia coli* – Mother Machine

The mother machine type of Z-stacks are the ones I worked on the most since I did most of my work on the toggle switch in this microfluidic device. The SVMs were trained to identify 6 classes:

- Inside: The inner part of the cells.
- Membrane: The outer part of the cell.
  While the distinction between those two parts of the cells is not strictly necessary and a single class for the entire cell gives equivalent results, this partition of the cell simplifies segmentation, and can be used for simple watershedding as demonstrated in section 5.1.
- Halo: The region of the mother machine chambers that is in between the cells and the PDMS wall.
- PDMS wall: The PDMS wall of the chambers.
- Chamber: Empty mother machine chambers.
- Empty: Empty parts of the image.

The training set was constructed with regions in the vicinity of the chambers because it was the only part of the images we were interested in. The classification is therefore not as good or relevant in other parts of the image. In section 5.1 we reduce the classified image to a region of interest (ROI) to avoid dealing with misclassified parts of the region.

In the results presented on Figure 4-1, we can see that different regions in the image are correctly labeled, except for a few mistakes that are caused by glitches in the image. Such minor errors can be ignored altogether or can be corrected a posteriori with morphological rules.

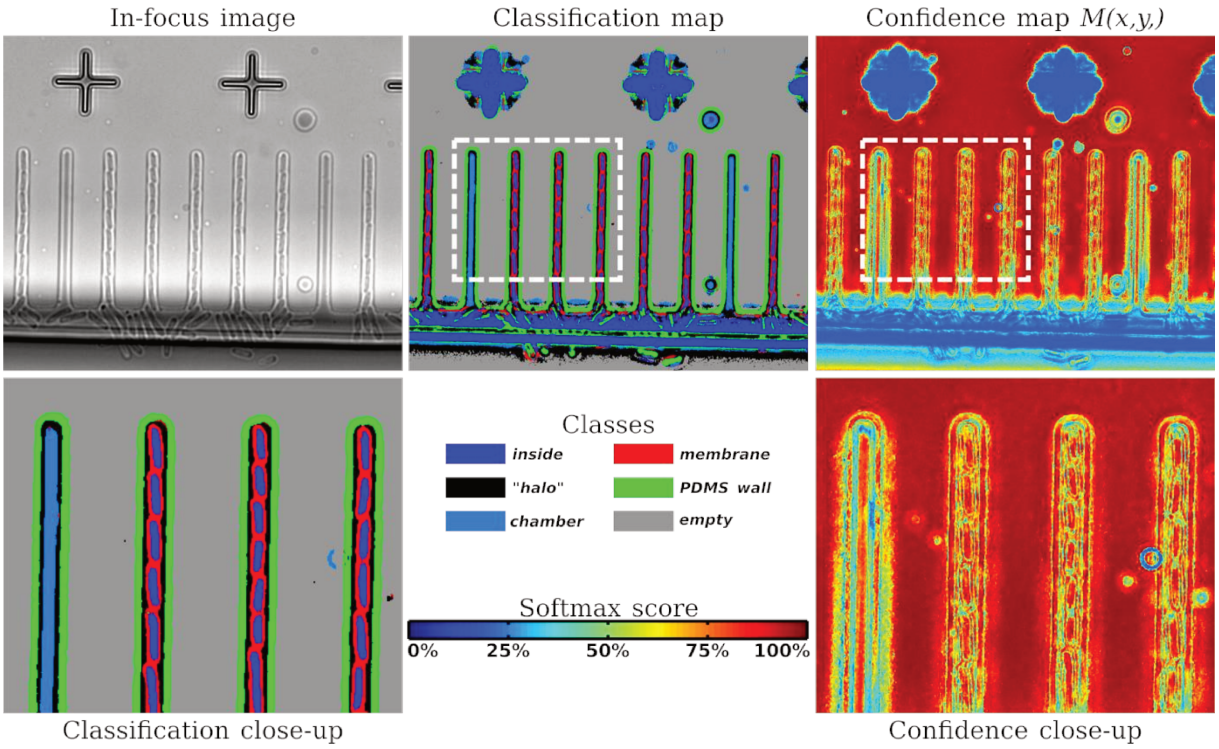

Figure 4-1 Identification result on mother-machine type of images. Stacks of 100 frames with 0.1um z-steps were acquired on Mustard with a 100X oil objective, in transmitted light. On the classification maps we can see that the procedure correctly labels almost all z-pixels in the mother machine chambers, but performs poorly on the rest of the image. This is because the training set was constructed only for the chambers part of the image since this is the only part we are interested in. On the confidence map, the softmax score of the classification is represented as a heat map. We can see that the confidence drops in the areas of the image the algorithm was not trained for. The confidence also drops at the boundary between different regions of the image.

An interesting aspect of the WTA-SVM technique used here is that it provides us, for each z-pixel, with scores for each of the 6 classes. As can be seen on the confidence map of Figure 4-1, the softmax classification score of the predicted class is lower at the boundary between classes: this is due to the transition from one class to the next as illustrated in Figure 4-2. This information can be exploited later for segmentation, as discussed in sections 5.2 and 5.3.

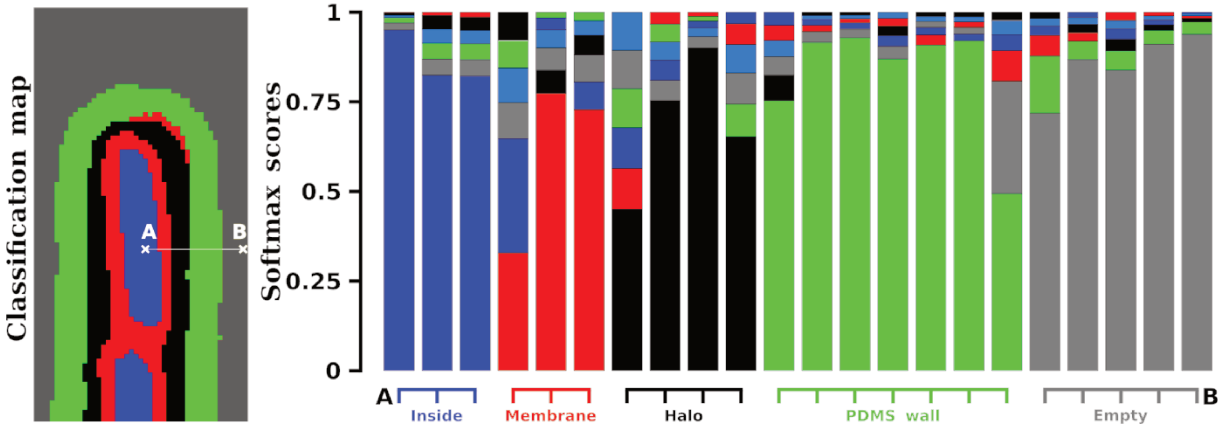

Figure 4-2 Scores per class for each z-pixel. The softmax scores for each class are represented for each z-pixel along a line between points A and B in the close-up of the classification map. We can see how the scores transition from one class to the next as the line goes through the different regions.

The effect of the number of frames per stack on performance was also studied. From the same stack and labeled z-pixels, different subsets of frames in the 100 that counted the stack were used, with different subsampling methods. We used between 3 and 99 frames per stack and performed feature extraction, training, and mislabeling evaluation on those subsets (see section 3). Once the SVMs were trained, we classified entire frames and recorded the total classification time. We discovered that, while higher numbers of frames would tend to increase performance, the gains were marginal after 10-20 frames per stack (see Figure 4-3A). We were able to obtain segmentable results with only 7 frames per stack (see Figure 4-3B), which could also be brought down by identifying the most informative frames in the stacks from feature extraction. This result can be particularly interesting in cases where time is an issue (for example in multi-position time-lapse acquisitions) or in cases where cells are sensitive to long exposure times.

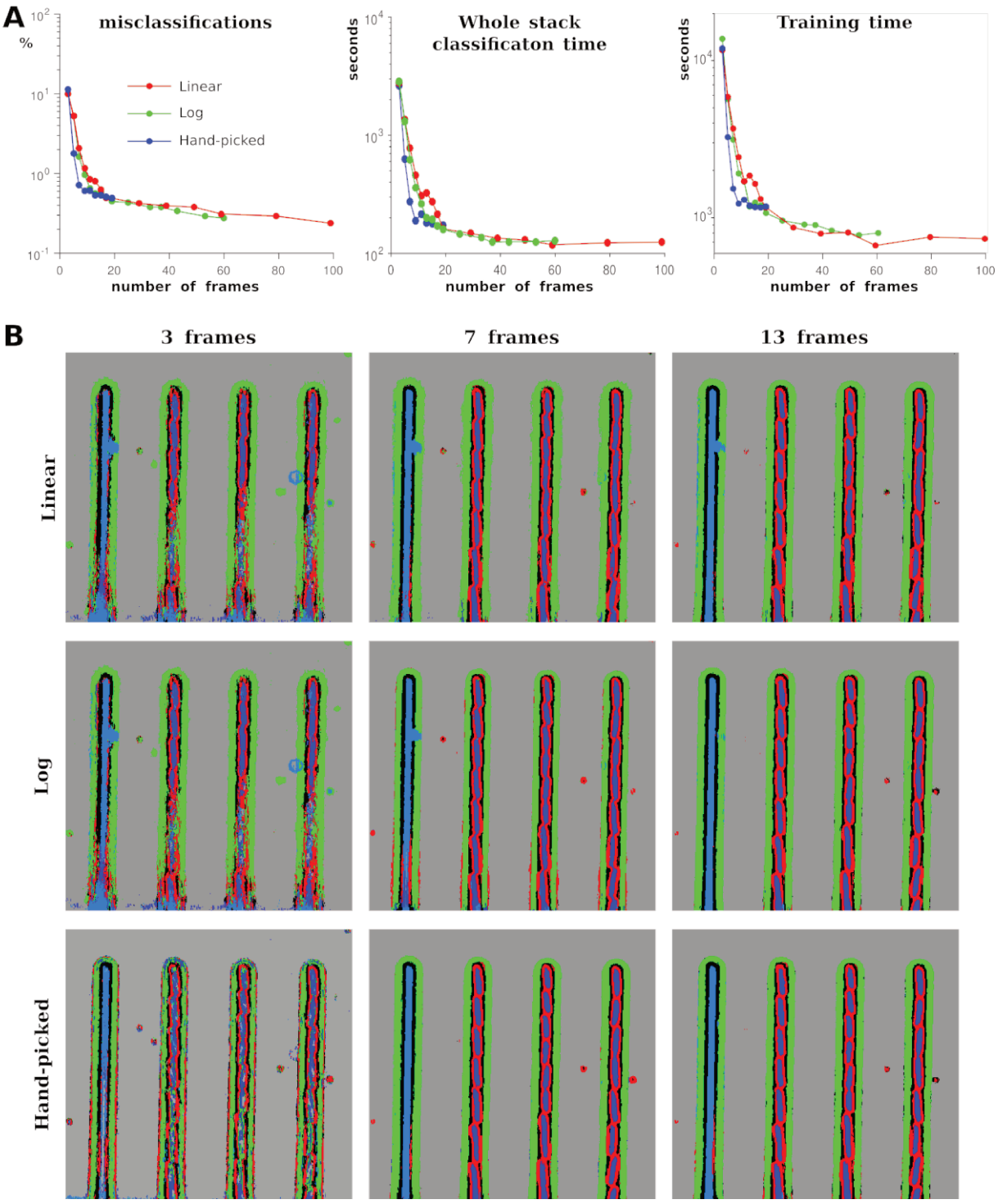

Figure 4-3 Performance for different numbers of frames per stack. The training procedure was performed for different sub-samplings of the original 100-frames stacks. Linear sub-sampling in which the selected frames are sampled linearly, i.e. with a constant z-step between them, was performed from 3 to 100 frames per stack. Log subsampling, in which the z-step increases logarithmically the further away the sampling is from the in-focus image, was performed from 3 to 60 frames per stack. Finally, a so-called hand-picked method was used, in which I chose the frames that seemed to be the most informative in the stack. A fourth method which I have not implemented yet would be to use the results from the feature extraction step to infer which frames are the most informative. A) Different performances were evaluated against the number of frames in the stack. The performance increases significantly between 3 and ~20 frames, and from there the gains in performance are marginal. The Hand-picked method gives the best results and reaches its plateau at already 7 frames-per-stack. Training was performed without parallelization, while whole-frame classification was parallelized on CPU over 20 cores. B) Classification result for the three methods and selected numbers of frames. At 7 frames per stack the results from the Hand-picked method are already good enough for the segmentation procedure described in section 5.1.

4.2.2 *Saccharomyces cerevisiae*

A number of people in the lab work on the budding yeast, Saccharomyces cerevisiae. While the size and shape of these cells make them slightly easier to segment than Escherichia coli cells, robust yeast cell segmentation is still an open problem.

Five classes were created for this training set:

- Inside: The inner part of the cells.
- Membrane: The outer part of the cells.
- Halo: White "halo" around the cells.
- PDMS wall: The PDMS wall of the device
- Empty: Empty parts of the image

In the results presented in Figure 4-4, we can see how the classification of yeast cell images produces little errors and makes the problem of yeast cell segmentation easier. Another interesting observation is that buds, the daughters of mother yeast cells, can be identified early on, and perhaps could even be connected directly to their mothers. The budding reproduction of yeast cells makes it difficult to reconstruct their lineage, and the possibility to automatically reconstruct lineages without using fluorescent markers would make the study of inherited behaviors much easier.

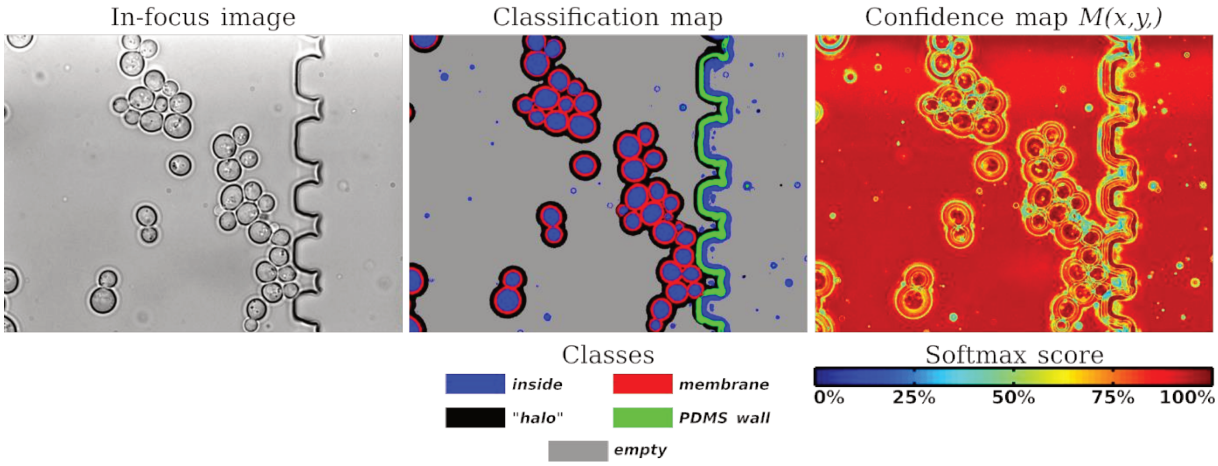

Figure 4-4 Identification result on yeast monolayer microfluidics. Stacks of 100 frames with 0.1um z-steps were acquired on Mustard with a 100X oil objective, in transmitted light. Yeast cells are identified almost perfectly. The training set was not constructed properly for the PDMS wall class, and parts of it are identified as cell inside or cell halo, but this misclassification can be corrected a posteriori by spatial rules like the ones presented in section 5.2.

4.2.3 *Saccharomyces cerevisiae & Escherichia coli*

Another interesting property of this approach is the possibility to identify different types of cells in an image. In this example yeast and bacteria were mixed and put under agar pads.

The five following classes were established for the training set:

- Inside_ecoli: The inner part of bacteria.
- Membrane_ecoli: The outer boundaries of bacteria.

- Inside_yeast: The inner part of yeast cells.
- Membrane_yeast: The outer boundaries of yeast cells.
- Empty: The empty parts of the images.

Because there were a reduced number of stacks in this dataset and the focus varies between stacks of this dataset, I could not train the images on some stacks and then evaluate it on some others. I had to construct the training set on the same stack I ran the classification on, but I separated the images in 4 sub-regions of equal size, three of which I used for training and the fourth I used for classification.

The stack also featured a reduced number of frames (35), which made the focus shifting method described in section 4.2.3 less efficient and classification quality worse overall. Confidence drops in at the interface between yeast cells and bacteria, but the quality of the classification is still sufficient for simple segmentation methods (see section 5.2 for a segmentation technique that re-labels misclassified bacteria once the yeast cells have been segmented.

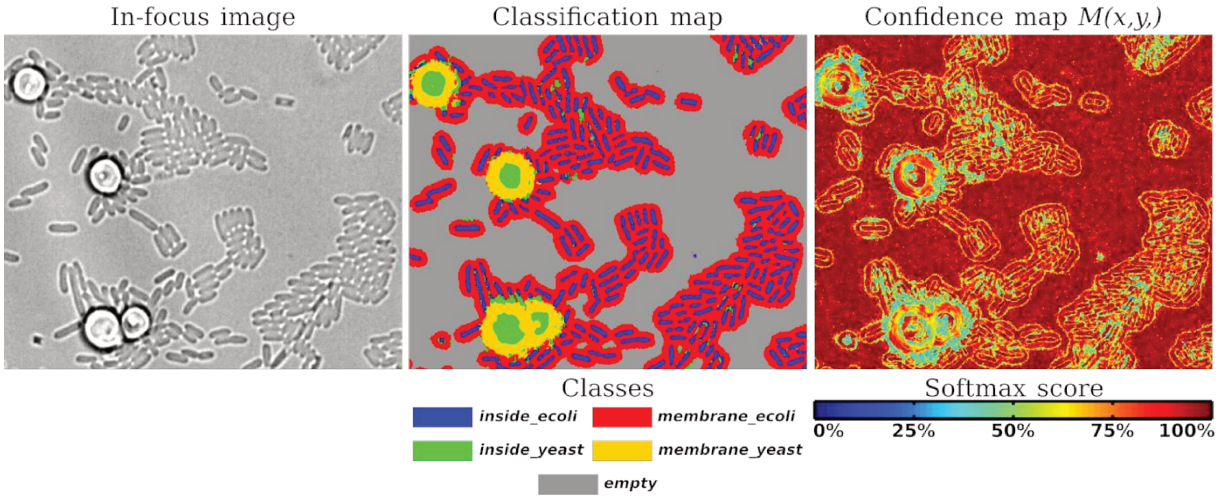

Figure 4-5 Identification result on monolayers of yeast and bacteria under agar pads. Stacks of 35 frames with 0.3um z-steps were acquired on St-Louis with a 60X oil objective, in transmitted light. Because the inside of yeast and bacteria are similar and the number of frames per stack in this dataset is lower than in others, parts of bacteria are misclassified as yeasts. This misclassification can be corrected a posteriori by spatial rules like the ones presented in section 5.2.

4.2.4 HeLa cells

Mammalian cells are notoriously difficult to segment. In this example, the boundaries of HeLa cells are identified and in section 4.2.4 those results are used to segment the cells with simple watershedding-based techniques.

Three classes are used here:

- Inside: The inner part of the cells.
- Membrane: The outer part of the cells.
- Empty: Empty parts of the image.

The best results are illustrated in Figure 4-6. In this figure the segmentation is nearly sufficient for simple watershedding as described in sections 5.1 and 5.2. However the cell

boundaries are not all completely closed, and in section 5.3 I present how the information from the confidence maps can be used to segment the cells.

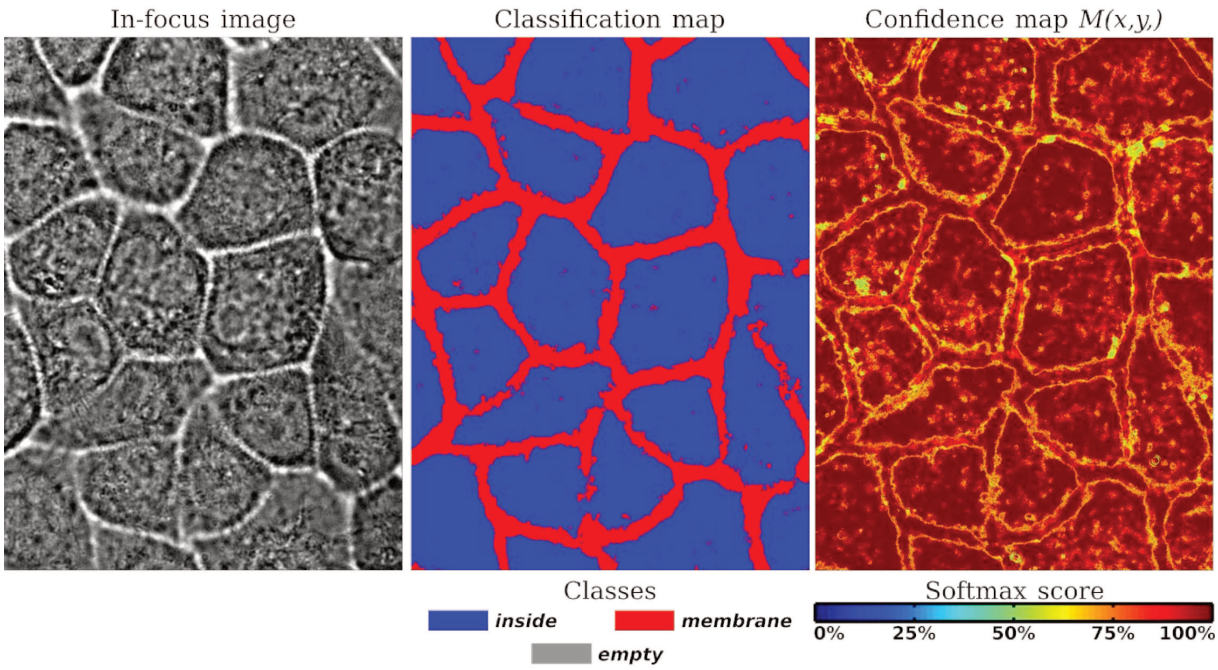

Figure 4-6 Identification results on HeLa cells. Stacks of 150 frames with 0.3um z-steps were acquired on Marple with a 60X oil objective, in transmitted light. The classification result is of excellent quality and the cells are almost segmented already. In section 5.3 I discuss a method that uses the confidence maps for closing the contours of the cells and segmenting them.

Mammalian cells have a tendency to clump together and the boundaries are not always as nicely delimited as in the in-focus image of Figure 4-6. In Figure 4-7 one of the worst results is presented: Boundaries are not as well-delimited, and at the interface between the empty region and the cells, the classification confidence drops. Although the technique presented in Section 5.3 is still able to segment the cells well, over or under- segmentation can occur.

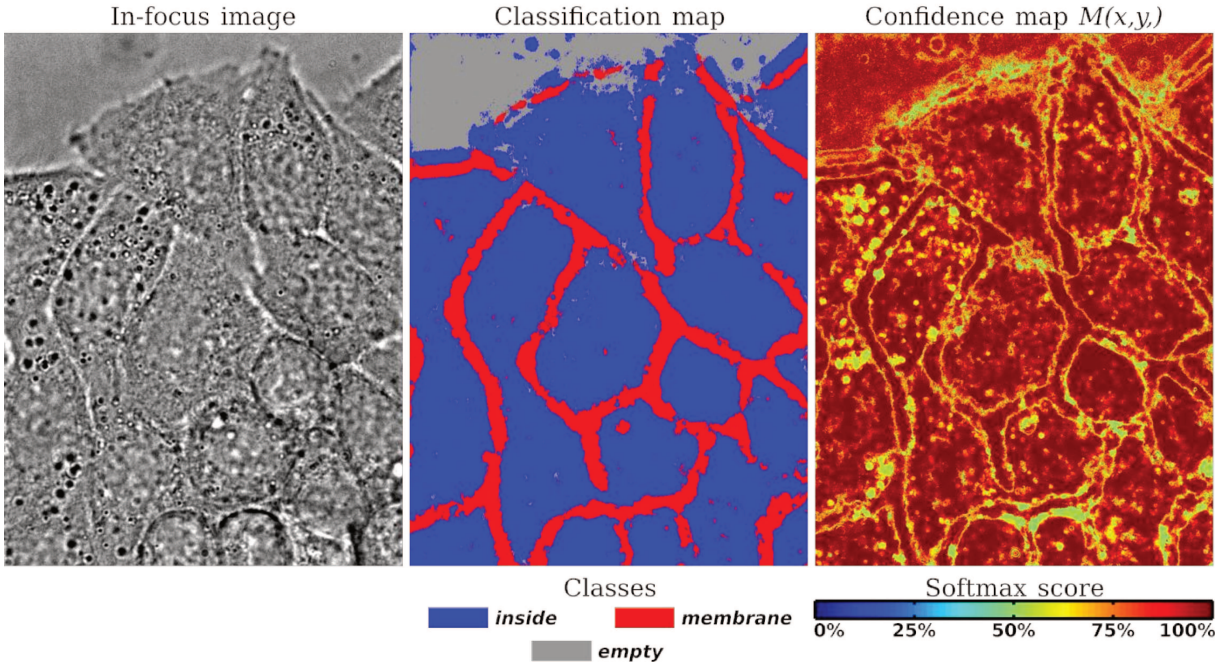

Figure 4-7 Identification results on HeLa cells. Stacks of 150 frames with 0.3um z-steps were acquired on Marple with a 60X oil objective, in transmitted light. Here the quality of segmentation is a bit lower, especially around the empty part of the image. In section 5.3 I discuss a method that uses the confidence maps for closing the contours of the cells and segmenting them.

5 Segmentation

I do not consider cell segmentation to be part of this algorithm, because almost all types of cell segmentation techniques can be used downstream of this classification/identification step. Traditional methods like watershedding or active contours can be used, or even deep learning approaches, although if one is to go through the trouble of implementing a deep learning algorithm, with a little extra effort the performance should be greatly improved by fusing the identification and segmentation step as I discuss in section 7.

However, here I will present simple implementations of cell segmentation to illustrate how easy cell segmentation becomes once the different parts of a microscopy image are identified, but also to illustrate how confidence maps and classification scores can be used to improve segmentation.

It should be noted that none of the segmentation procedures presented here use active contours or deformable models. Hence no assumption on the shape of the cells is made by the experimenter, which should improve the segmentation performance if chosen properly.

5.1 *Escherichia coli* – Mother Machine

This example illustrates the simplest form of segmentation that can be used to identify single cells in classified images. I will apply this procedure to the results described in section 4.2.1 because it is the most relevant to this thesis, although a similar approach can be used on the results presented in section 4.2.2.

A simplified form of 2-step watershedding is applied to the classified image as follows: The different watershedding objects/regions are seeded as the “cytoplasm” regions of the bacteria and are then spread into the membrane regions until they connect, thus delimiting the borders of each cell. The procedure as well as the results are described in Figure 5-1.

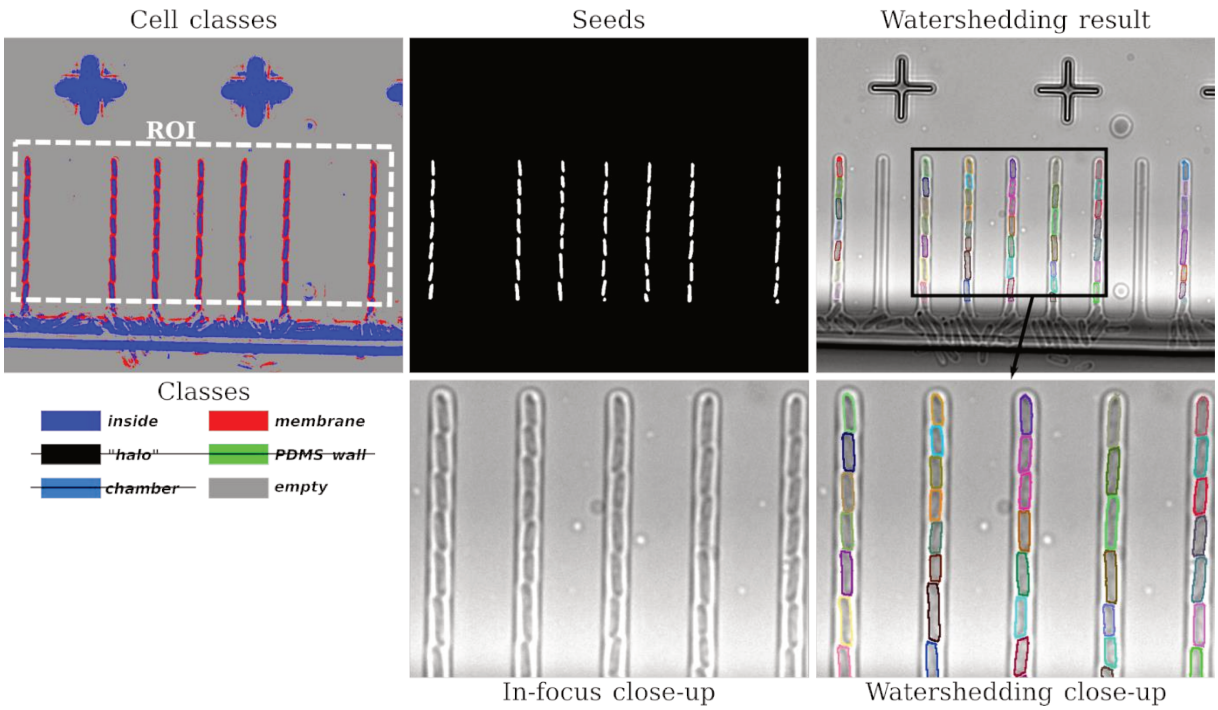

Figure 5-1 A Seed-based watershedding method is used to segment cells from the classification map. The halo, PDMS wall and chamber regions are removed and re-labeled as empty. A region of interest (ROI) is drawn to avoid the parts of the classification map that are mislabeled. Another procedure could be to identify the chambers first, from the PDMS wall class, and use those as ROIs. Only regions of the inside class that are within a membrane contour are used as seeds, and a morphological opening of 2-pixels radius is applied to clean the image. The seed-based watershedding algorithm is applied to expand the seeds back into the classes inside and membrane and identify single cells.

This simple procedure produced perfect segmentation on mother machine type of acquisitions, and excellent (>90%) results on monolayers of both bacteria and yeast cells.

5.2 *Saccharomyces cerevisiae & Escherichia coli*

In this example I will illustrate how spatial or morphological rules can be used to correct misclassification errors. The segmentation presented here is used on the results presented in section 4.2.3. The only morphological assumptions that is made here is that the "inside_yeast" class should be surrounded or almost entirely surrounded by a region of the "membrane_yeast" class. Based on this assumption, we filter out regions of both the "inside_yeast" class and "membrane_yeast" class that do not respect this rule, and use a watershedding procedure similar to the one described in section 5.1 to identify yeast cells.

Then, the scores of the "inside_yeast" and "membrane_yeast" classes are removed from the $\boldsymbol{C}^{p}$ classification vectors to form the new $\boldsymbol{C}^{p\prime}$. The softmax vectors $\boldsymbol{S}^{p\prime}$ are then re-computed for those new classification scores. The z-pixels that were classified as part of the two removed classes but did not end up being part of the final identified yeast cells regions are re-labeled as the next most-probable class in $\boldsymbol{S}^{p\prime}$. This way, z-pixels that were wrongly labeled as part of yeast cells are re-labeled as other classes, in most cases for the better. See Figure 5-2 for an illustration of the segmentation results.

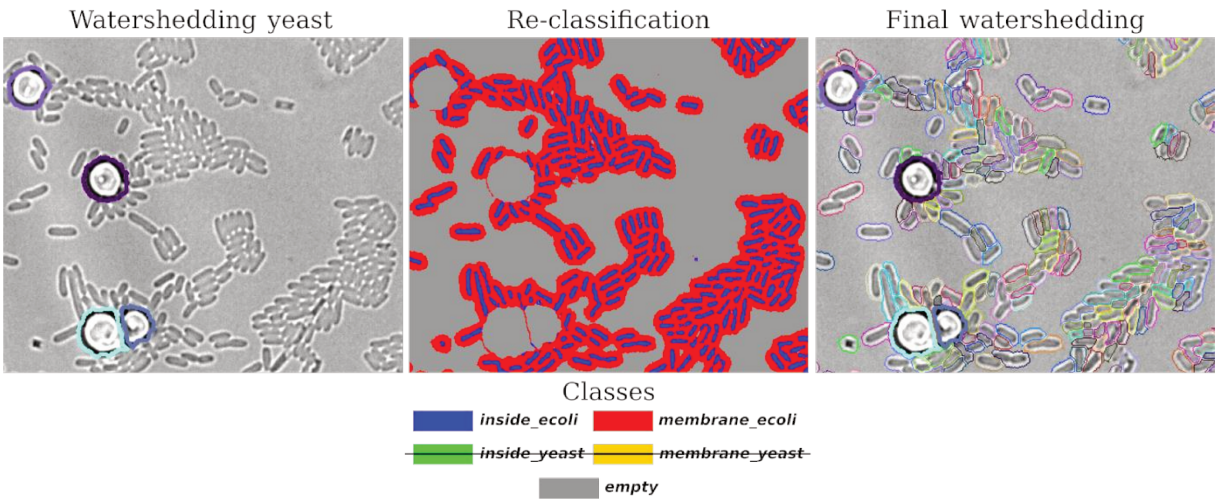

Figure 5-2 Identifying yeast and bacteria separately. The yeast cells are identified first in a procedure similar to the one described in section 5.1. Once yeast cells are identified, the yeast classes are removed and the mislabeled elements are re-classified as the next highest-scoring class. Watershedding is applied again to identify bacteria.

One of the advantages of this method, as already discussed in section 4.2.3, is that it can be used to segment images in which a mixture of cells are present, a result that has, to the best of my knowledge, so far eluded the field of single cell segmentation. This type of approaches could open the way to high-throughput quantitative analysis of ecosystems of cells.

5.3 HeLa cells

In this example I will illustrate the use of information from the class maps, and in particular the class map for the membrane class, to improve segmentation performance in the results presented in section 4.2.4. Because the membrane regions in the classified images do not completely encircle all the cells in the image, instead of using the z-pixels that are strictly identified as membrane by our WTA-SVM approach, we will use the class map for the membrane to identify which z-pixels are most likely to be the contour of the cells.

The first step in the process is to obtain a rough estimate of the cell centers by eroding the cell objects with a structuring element of variable size (see section 1.1). This step is actually crucial because the size of the structuring element is the only parameter that is set arbitrarily by the experimenter. Too big a structuring element will remove small cells, but if it is too small it may not separate cells that are connected on the classified image.

From those reduced cell regions, the seeds are expanded back into the confidence map until they connect. The segmentation is excellent for the first stack in Figure 5-3, with only minor errors such as removal of cells on the outer parts of the image. In Figure 5-4 however, although the result is well above what a typical single-image-based algorithm would do, either based on deformable models or deep learning, the segmentation is not perfect and is partially under-segmented. Deformable model methods applied to confidence maps might yield better results, but the limit of this SVM & Z-stacks approach might be reached. In section 7 in particular I discuss how a combination of this method with state-of-the-art deep learning methods could be used in conjunction with the Z-stack approach to yield better results.

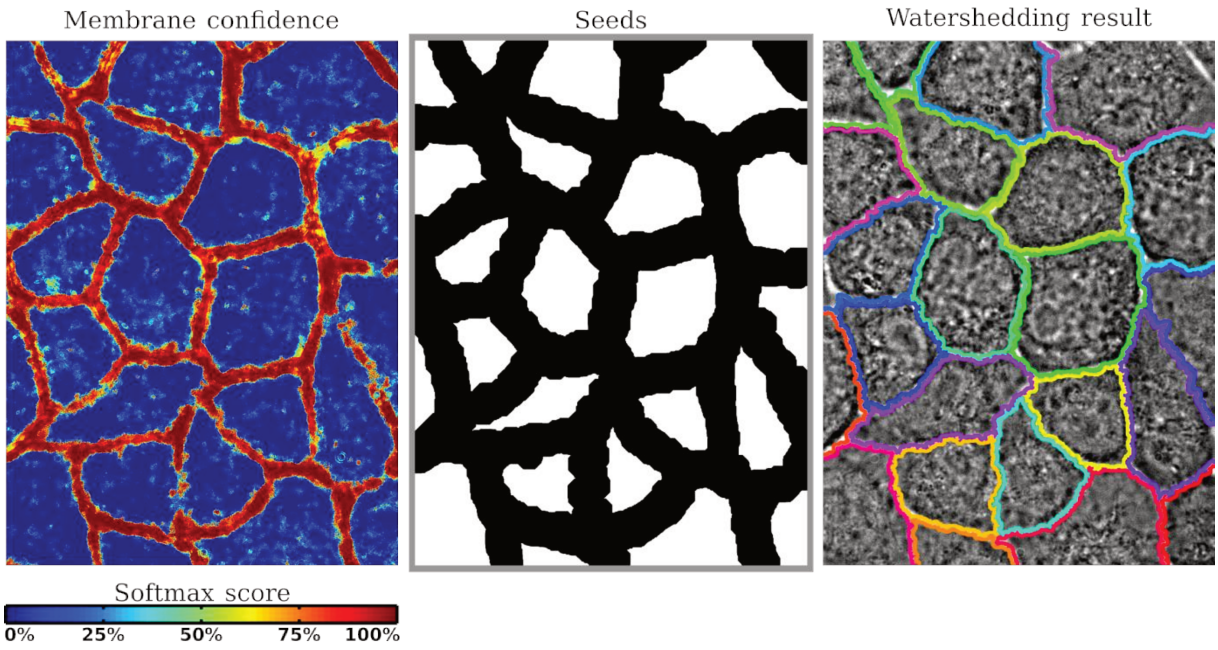

Figure 5-3 Watershedding on the confidence map. This method uses the confidence map, and not the classification map, to segment cells. Here the classification does not completely close the contour of the cells. Seeds are identified from the lowest-scoring elements of the Membrane confidence map and morphologically eroded to partition cells. The seeds are then expanded back into the membrane confidence map until they all connect. This method has the advantage of using information from the classification confidence instead of morphological rules.

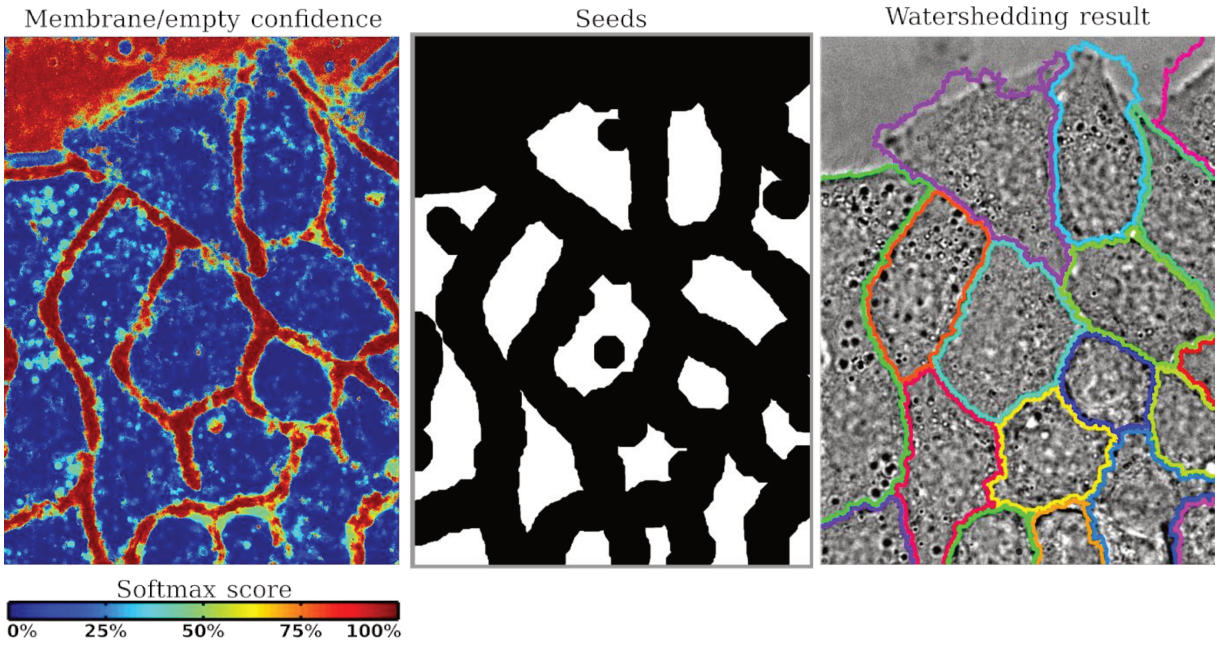

Figure 5-4 Watershedding on the confidence map. On this stack, the same procedure as in Figure 5-3 is applied. But seeds identification relies on arbitrary parameters, and the choice of the parameters can lead to under- or over-segmentation. Here the leftmost cell region is probably two different cells, but is not identified as such.

6 Towards 3D segmentation

To my knowledge, there have been no algorithms proposed for 3D segmentation of cells in Z-stacks of transmitted light images. While this problem is even more difficult than cell segmentation in a single focal plane, the possibility to segment colonies of cells freely growing in 3 dimensions would make the study of complex ecosystems of cells possible, or the evolution and growth of tissues, biofilms or embryo without requiring complex, and sometimes impossible, cloning procedures.

In this section I present a slight modification of the procedure described in previous sections that gave encouraging results in this direction. I used stacks of Staphylococcus aureus and mammalian red blood cells in solution. One particularity of those stacks is that cells are not all in the same focal plane, with red blood cells being in-focus about 7 microns above the bacteria. Because of this wide variety in focus I decided to try to segment the cells in 3D. Instead of using the entire stack to train and segment the cells, I used only 20 frames around the focal plan of the different parts of the training set and trained the SVMs on those local signatures. I then ran the classification on all possible moving stack of 20 frames along the Z axis through the entire 300-frame stack. For each center frame the local stack would be classified and the classification scores were kept.

In Figure 6-1, the in-focus frame, as well as the corresponding classification map, are represented for different local Z-stacks. The different parts of the image are correctly classified when they are in focus, but are incorrectly classified when out of focus. This error can probably be blamed on bad dataset construction, which is a tedious task in 3 dimensions, and also on the long computation times: The software has not been developed, and therefore not optimized, for this 3-dimensional approach, and the classification time for an entire stack is counted in hours. This puts a hard limit on the speed of development of this aspect of the algorithm. Furthermore, 3D convolutional neural networks might be more adapted to this type of application, as discussed in section 7.

The classes used in this training set were the following ones:

- Staph_inside: Inner part of Staphylococcus aureus.
- Staph_membrane: Outer part of Staphylococcus aureus.
- Red_blood_cell: Entire red blood cells.
- Empty: Empty parts of the image.

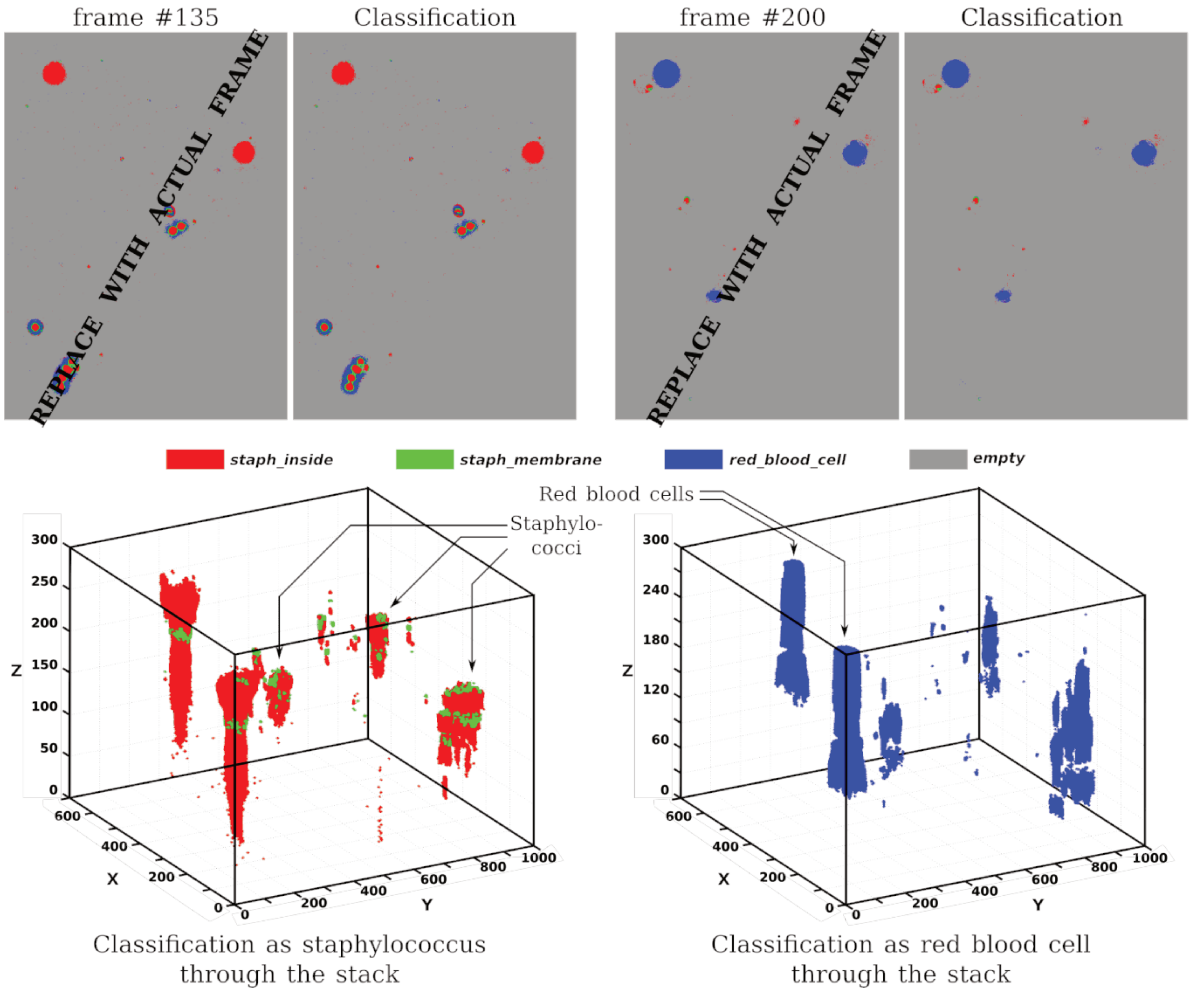

Figure 6-1 3D segmentation. Stacks of 300 frames with 0.1um z-steps were acquired on Zurich with a 60X oil objective, in transmitted light. Both Staphylococcus aureus and mammalian red blood cells are imaged together in solution. The different cells are all on different focal plans, and a training set consisting in local subsamples of the stack was used to train an SVM architecture that can identify different signatures along the X and Y axes, but also along Z. The Staphylococci and red blood cells are correctly identified in their respective frames, but are mislabeled in others.

7 Conclusion

In this chapter I presented a machine-learning-based algorithm for image analysis inspired by techniques in hyperspectral imaging. The algorithm identifies regions of the image as part of a certain class of objects, which simplifies the subsequent segmentation process. It can be used on a variety of organisms without requiring any changes to the program itself, but only the design of new labeled training set. Simple image segmentation techniques can be applied to the classified images and provide segmentation performances on par with state-of-the-art segmentation algorithm. The algorithm also provides a “classification confidence” metric that can be useful to assess the quality of the identification results and adjust the segmentation step accordingly. It can be adapted to a variety of microscopy setups and to a variety of acquisition parameters and still provide satisfactory results.

One of the advantages of this method is that SVMs typically require less samples for training than deep learning techniques (Liu et al. 2016) and, since they do not train on entire z-stacks

but on each z-pixel independently, it is not necessary to label entire stacks for training. In fact, we achieved results that were almost as good with only small regions of the stacks labeled (2-5% of one stack labeled would already provide segmentable identification results over hundreds of other stacks). Although z-stacks can rapidly represent a significant amount of data to store for time-lapse experiments, which could limit the applicability of the method, it is possible to perform the identification online when the stacks are acquired and not store the stacks but only the identification results. We developed a Matlab script for online, parallel identification that can be used with our more general software framework described in chapter 2.

Although simple segmentation algorithms applied to identified images were sufficient to achieve state-of-the-art segmentation performance, the performance could be further improved by the use of active contour or deformable model techniques. Another combination with existing techniques would be the use of identified images as the input for a deep learning segmentation method, like the U-Net architecture (Ronneberger et al. 2015). However, as suggested by recent advances in hyperspectral imaging (Zhang et al. 2016) it would also be possible to perform the identification step through a deep learning approach. To my knowledge, there are no published examples of segmentation from hyperspectral data using a deep-learning based framework yet, and only identification problems have been studied so far. However combining the identification function and the segmentation function into a single neural network based on the U-Net architecture but with 3x3xN convolutional layers, with N being the number of frames per stack, instead of the 3x3 convolutional layers used in the original U-Net paper (see Figure 7-1). We started investigating this possibility but the necessary tools for 3D convolution had not been developed yet. A few months ago, the two main software libraries in deep learning implemented new functions that make 3D-convolutions possible, making the development of such architectures an interesting extension to the work presented here.

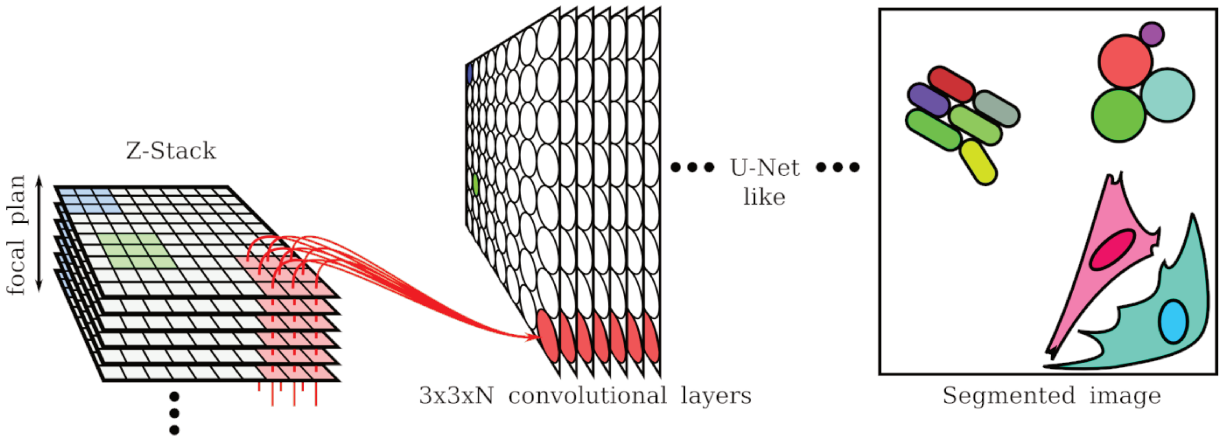

Figure 7-1 Possible architecture for a deep learning approach of Z-stack segmentation. Instead of 2-dimensional convolutional layers, 3-dimensional convolutional layers can be used as the first level in a U-Net like network (see section 1.2.3 and Figure 1-6) to identify single cells in an image.

Finally, robust, unsupervised and fast segmentation of any type of cells in timelapse fluorescence images is a quantum leap, not only for external control of gene expression at the single-cell level where online cell segmentation and tracking is crucial, but also more

generally for quantitative biology the possibility to generate single-cell data robustly and with minimal human inputs would allow for a finer level of analysis of gene expression in gene regulation networks at the population level but also, with sufficiently good data, at the lineage level.

Asimov, I., 1971. The Eureka Phenomenon. The Magazine of Fantasy and Science Fiction.

Boser, B.E., Guyon, I.M. & Vapnik, V.N., 1992. A Training Algorithm for Optimal Margin Classifiers. Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory, pp.144–152.

Bottou, L. & Bousquet, O., 2007. The Tradeoffs of Large Scale Learning. Nips, 20, pp.161–168.

Canny, J., 1986. A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8(6), pp.679–698.

Catanzaro, B., Sundaram, N. & Keutzer, K., 2008. Fast support vector machine training and classification on graphics processors. In Proceedings of the 25th international conference on Machine learning - *ICML '08*. New York, New York, USA: ACM Press, pp. 104–111.

Cesa-Bianchi, N., Gentile, C. & Zaniboni, L., 2006. Hierarchical classification. In Proceedings of the 23rd international conference on Machine learning - *ICML '06*. New York, New York, USA: ACM Press, pp. 177–184.

Chen, Y. et al., 2014. Deep Learning-Based Classification of Hyperspectral Data. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 7(6), pp.2094–2107.

Ciresan, D. et al., 2012. Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images. In Proceedings of the Neural Information Processing Systems Conference. Lake Tahoe, pp. 1–9.

Dalal, N. & Triggs, B., 2005. Histograms of Oriented Gradients for Human Detection. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition *(CVPR'05)*. IEEE, pp. 886–893.

Davis, L.S., 1975. A survey of edge detection techniques. Computer Graphics and Image Processing, 4(3), pp.248–270.

Dougherty, E.R. & Lotufo, R.A., 2003. Hands-on Morphological Image Processing, 1000 20th Street, Bellingham, WA 98227-0010 USA: SPIE.

Duan, K.-B. & Keerthi, S.S., 2005. Which Is the Best Multiclass SVM Method? An Empirical Study. In N. C. Oza et al., eds. Multiple Classifier Systems. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 278–285.

Gao, L. & Smith, R.T., 2015. Optical hyperspectral imaging in microscopy and spectroscopy - a review of data acquisition. Journal of biophotonics, 8(6), pp.441–56.

Gentile, C. & Warmuth, M.K., 1998. Linear hinge loss and average margin. In Nips. pp. 225–231.

Glasbey, C.A., 1993. An Analysis of Histogram-Based Thresholding Algorithms. CVGIP: Graphical Models and Image Processing, 55(6), pp.532–537.

Gowen, A. et al., 2007. Hyperspectral imaging – an emerging process analytical tool for food quality and safety control. Trends in Food Science & Technology, 18(12), pp.590–598.

Gualtieri, J.A. & Cromp, R.F., 1999. Support vector machines for hyperspectral remote sensing classification. In R. J. Mericsko, ed. SPIE. pp. 221–232.

Hotelling, H., 1933. Analysis of a complex of statistical variables into principal components. Journal of Educational Psychology, 24(6), pp.417–441.

Hughes, G., 1968. On the mean accuracy of statistical pattern recognizers. IEEE Transactions on Information Theory, 14(1), pp.55–63.

LeCun, Y., Bengio, Y. & Hinton, G., 2015. Deep learning. Nature, 521(7553), pp.436–444.

Liu, P. et al., 2016. SVM or deep learning? A comparative study on remote sensing image classification. Soft Computing, pp.1–13.

Long, J., Shelhamer, E. & Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, pp. 3431–3440.

Mathur, A. & Foody, G.M., 2008. Multiclass and Binary SVM Classification: Implications for Training and Classification Users. IEEE Geoscience and Remote Sensing Letters, 5(2), pp.241–245.

van der Meer, F.D. et al., 2012. Multi- and hyperspectral geologic remote sensing: A review. International Journal of Applied Earth Observation and Geoinformation, 14(1), pp.112–128.

Meijering, E., 2012. Cell Segmentation: 50 Years Down the Road. IEEE Signal Processing Magazine, 29(5), pp.140–145.

Meijering, E. et al., 2009. Tracking in cell and developmental biology. Seminars in Cell & Developmental Biology, 20(8), pp.894–902.

Melgani, F. & Bruzzone, L., 2004. Classification of hyperspectral remote sensing images with support vector machines. IEEE Transactions on Geoscience and Remote Sensing, 42(8), pp.1778–1790.

Mountrakis, G., Im, J. & Ogole, C., 2011. Support vector machines in remote sensing: A review. ISPRS Journal of Photogrammetry and Remote Sensing, 66(3), pp.247–259.

Nielsen, M., 2015. Neural Networks and Deep Learning, Determination Press.

Osowski, S. et al., 2009. Application of Support Vector Machine and Genetic Algorithm for Improved Blood Cell Recognition. IEEE Transactions on Instrumentation and Measurement, 58(7), pp.2159–2168.

Otsu, N., 1979. A Threshold Selection Method from Gray-Level Histograms. IEEE Transactions on Systems, Man, and Cybernetics, 9(1), pp.62–66.

Prewitt, J.M.S. & Mendelsohn, M.L., 1966. The analysis of cell images. Annals of the New York Academy of Sciences, 128(3), pp.1035–1053.

Raina, R., Madhavan, A. & Ng, A.Y., 2009. Large-scale deep unsupervised learning using graphics processors. In Proceedings of the 26th Annual International Conference on Machine Learning - *ICML '09*. New York, New York, USA: ACM Press, pp. 1–8.

Ramoser, H. et al., 2005. Leukocyte segmentation and classification in blood-smear images. In 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference. IEEE, pp. 3371–3374.

Ronneberger, O., Fischer, P. & Brox, T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. ArXiv.

Rosenblatt, F., 1957. The perceptron, a perceiving and recognizing automaton, Cornell Aeronautical Laboratory.

Rumelhart, D.E., Hinton, G.E. & Williams, R.J., 1986. Learning representations by back-propagating errors. Nature, 323(6088), pp.533–536.

Steinwart, I. & Christmann, A., 2008. Support Vector Machines, New York, NY: Springer New York.

Vapnik, V., 1963. Pattern recognition using generalized portrait method. Automation and remote control, 24, pp.774–780.

Vapnik, V. & Kotz, S., 1982. Estimation of dependences based on empirical data, New York: Springer-Verlag.

Vermaas, W.F.J. et al., 2008. In vivo hyperspectral confocal fluorescence imaging to determine pigment localization and distribution in cyanobacterial cells. Proceedings of the National Academy of Sciences of the United States of America, 105(10), pp.4050–5.

Yanowitz, S.D. & Bruckstein, A.M., 1988. A new method for image segmentation. In [1988 Proceedings] 9th International Conference on Pattern Recognition. IEEE Comput. Soc. Press, pp. 270–275.

Yuan, X. et al., 2006. Automatic Video Genre Categorization using Hierarchical SVM. In 2006 International Conference on Image Processing. IEEE, pp. 2905–2908.

Zhang, L., Zhang, L. & Kumar, V., 2016. Deep learning for Remote Sensing Data. IEEE Geoscience and Remote Sensing Magazine, (june), p.18.

Chapter V
Conclusion and perspectives

Table of Contents

1 Contributions 162
2 Limitations 163
3 Developments of the control platform 164
3.1 Active microfluidics 164
3.2 Optogenetics 164
3.3 Fluorescence 165
3.4 Image analysis 165
3.5 Cloning and genome engineering 166
4 Perspectives 166

1 Contributions

In this dissertation, I presented several tools for implementing a platform for single-cell control of gene expression in bacteria. I propose this platform to control a bistable, 2-inputs 2-outputs genetic circuit, the genetic toggle switch, to illustrate the possibilities offered by external control of gene expression, but also to study the dynamics of this fundamental network. A seducing aspect of the control results presented here is that a diffcult task like stabilizing the genetic toggle switch in its unstable area was performed with a simple control framework and minimal a priori knowledge of the controlled system. In general, the shift of the implementation of the regulatory process from in vivo to in silico allows for faster implementation-experimentation cycles, and more importantly allows for the development of control strategies that are significantly more complex than what can currently be designed for in vivo regulation.

The online single-cell aspect of the platform and acquired data presented here is also important. Single-cell data measures gene regulation networks dynamics at a finer level of details than population or flow cytometry measurements, and is crucial for the study of some aspects of gene expression. For example distinguishing intrinsic and extrinsic noise in the dynamic study of stochastic gene expression would be impossible without long-term single-cell definition. The study of inheritance in populations of cells would also be impossible without single-cell data, tracking, and lineage reconstruction in timelapse microscopy images. In this study, controlling a multistable system like the toggle switch would not have been possible without single-cell timelapse data: On one hand, if fluorescence had been extracted from timelapse data at the population level, like in (Fracassi et al. 2016; Fiore et al. 2013) , and not at the single-cell level, it would have been impossible to distinguish cells in the unstable area from a mixed population of cells in either of the two basins of attraction, since the fluorescence in those states would have been averaged. On the other hand, flow cytometry or similar methods used in other control platforms (Melendez et al. 2014; Milias-Argeitis et al. 2011; Milias-Argeitis et al. 2016) would not have provided the history of the measured cells and it would have been impossible to know whether cells were really maintained in the unstable area or simply transiting through it.

With this platform, we studied the possibility of real-time control on a synthetic bistable gene regulation network, the LacI-TetR genetic toggle switch. As illustrated in the state-of-the-art of this thesis, toggle switches, and multistable systems in general, are known to play a central role in cellular decision-making, and to be a key component of synthetic biology circuits. We showed two approaches to maintain this bistable genetic circuit close to its unstable equilibrium state: closed-loop control and periodic stimulation. We demonstrated that single-cell control of a bistable system can be achieved with relatively simple control frameworks and very little a priori knowledge of the system. We also demonstrate the possibility to perform multiple-input-multiple-output control studies of genetic networks, which would facilitate and accelerate the examination of internal network dynamics. Finally, the dynamic stabilization of an entire population of toggle switch cells in the unstable area of the system in open-loop experiments is a new observation that suggests that multistable genetic networks could be extracted from, and maintained outside of, their stable states without observing their

state in real-time. Another implication is that gene regulation networks can feature hidden stable states that could occur because of dynamic stabilization phenomena. Therefore, with the advent of the dynamic study of gene regulation networks, it should be possible to look for such effects that could not be discovered throughout the simple study of static connections between different elements of a gene regulation network.

One of the important aspects of single-cell control of gene expression is the ability to segment and track single cells. In the previous chapter of thesis I presented a new concept for automated cell segmentation. An untapped amount of optical information about the objects observed through a microscope is hidden in focal planes above and below the focal plane of the specimen. I developed a machine-learning algorithm that exploits this information to accurately identify regions of the image as part of different classes of objects. The procedure can be applied to mixtures of various types of objects like yeast cells, bacteria, mammalian cells, but also microfluidic structures. I also demonstrate that with this identification of the different regions of the image cell segmentation is facilitated and can be achieved with rudimentary image analysis tools. The development of machine learning-based tools for robust image analysis should lower the necessary efforts to control and investigate new gene regulation networks at the single-cell level in new organisms.

2 Limitations

The approach, of course, has its limitations. The use of chemical inducers like IPTG and aTC forced us to modify the host strain to ensure it would not interfere with the concentrations of inducers applied to the cells. The modifications increased delays in our system which limited the control performance. It also made the cells less more sensitive to environmental stresses, which also made control experiments more challenging since the cells had a tendency to filament during experiments. The use of chemical inducers should not be abandoned however, but used in conjunction with optogenetic induction mechanisms to expand the number of possible input knobs. Optogenetics and chemical inducers can also be used to examine the interplay between environmental changes and intracellular phenomena.

Also, the platform as it is implemented now only controls fluorescence level, and not actual proteins levels or the level of some biologically relevant element. It appears that in all applications of external control platforms so far, the state variable that is actually controlled is the one that is directly measurable, i.e. fluorescence. It is possible to estimate the levels of the system variables that are not directly measurable though, through state estimation techniques that have been part of control theory for a long time. In (Uhlendorf et al. 2012; Milias-Argeitis et al. 2011; Fiore et al. 2015; Milias-Argeitis et al. 2016) such state estimation filters were used, to not only estimate the state of the observed fluorescent probes that were being controlled, but also estimate the state of internal variables of their models. Those internal variables could also be controlled. In biological systems, most often the variable/process to control is not directly measurable and its observation is hidden behind intricate dynamics and noise. The ability to control hidden state variables in gene regulatory networks would be a

remarkable development in external cell control and would expand the area of its possible applications.

The application of model-predictive control has not been performed on our control problem. Although plugging this new controller into the platform would be facilitated by its modular structure, and we now have the knowledge necessary to establish a new model of the dynamics for controlling our toggle switch (since models used for model-predictive control are usually simpler than the one we have described here) we have not experimented with this approach yet, except in preliminary in silico studies. One of the possible extensions of the work presented here would be the analysis of control performance of the genetic toggle switch with this type of approach.

Several limitations are also inherent to the various elements of the platform. In the next section I discuss recent developments in various related domains that might trigger other progresses in the field of external control of gene expression.

3 Developments of the control platform

The field of in silico cybergenetics has emerged from recent developments in microfluiduics, molecular biology, systems biology, image analysis and in vivo measurement automation that made the assembly of external control platforms possible. Naturally, the range of applications of the field is going to expand with progresses made in those areas over the years to come. Some recent progresses have not been integrated to the platforms yet, or only partially, and here I am going to discuss some of those recent advances that could benefit our domain.

3.1 Active microfluidics

In our platform, and in all microfluidics-based platforms used for control so far, the valves and mixing apparatus were not fully integrated to the chip. It is possible however to integrate various valves and pumps to the microfluidic device (Unger et al. 2000; Li et al. 2005; Sundararajan et al. 2005). Those mechanisms have since been used to develop high-throughput labs-on-a-chip for chemistry and molecular biology, but they have also been used successfully for high-throughput studies of gene expression in live cells (Sorre et al. 2014; Dénervaud et al. 2013). Such systems could be used in control platforms to parallelize the study over different gene networks, different strains, or different independent stimulations.

3.2 Optogenetics

Optogenetics is a biological technique which involves the use of light-sensitive proteins to control intracellular processes, such as gene expression, protein recruitment, or enzymatic activity. The field is relatively recent, with the first system for controlling gene expression in yeast (Shimizu-Sato et al. 2002) appearing in the early 2000s. Over the following decade other optogenetic systems were developed in bacteria (Levskaya et al. 2005; Tabor et al. 2011; Kaberniuk et al. 2016) and mammalian cells (Strickland et al. 2008; Gunaydin et al. 2010; Toettcher et al. 2011a) to control either gene expression, signaling, or enzymatic activity.

Although optogenetic has been used in some control platforms (Milias-Argeitis et al. 2011; Melendez et al. 2014; Milias-Argeitis et al. 2016) its integration into microfluidic platforms is more difficult, although not impossible (Renault et al. 2015). Integrating optogenetic actuation into the light path of the microscope makes it even possible to illuminate the cells with chosen patterns (Levskaya et al. 2009; Yang et al. 2013), which would open the door to independent single-cell actuation of all cells within a population.

3.3 Fluorescence

Fluorescent probes design has gone through a rebirth in the last few years with the advent of super-resolution imaging (Mishin et al. 2015). One of the interesting developments for external control of gene expression is that new fluorescent probes have been developed that span new regions of the visible spectrum, therefore making it possible to observe higher numbers of fluorescent proteins at once. A promising result in this regard is the expansion of the space of discrimination between proteins into a new dimension: Time-modulation of light inputs, in addition to the traditional discrimination through the wavelengths of excitation and emission of the proteins, should make it possible in the near future to discriminate between a number of fluorescent proteins never reached before (Querard et al. 2015). Other remarkable advances in this direction are made to expand the usable spectrum for illumination and observation into the infra-red (Filonov et al. 2011). But infrared fluorescent proteins do not only expand the range of usable wavelength, they also permit deeper penetration of the light input into biological materials, and could be used in the future to image and control cells or cell populations inside colonies, biofilms or even tissues. The development of photoacoustic probe proteins goes even further in this direction, recently making the observation of gene expression up to 10 millimeters deep within tissues possible in live mice at high spatial resolution and in 3 dimensions (Yao et al. 2015).

The problems of fluorescent protein maturation delays and photobleaching limit the resolution of discernable dynamics in timelapse fluorescent imaging, and this can reduce the control performance. Fast-maturing fluorescent proteins have been developed over the years (Bevis & Glick 2002; Fisher & DeLisa 2008) but a recent shift in the field of fluorescent probe design that solves both maturation delays and bleaching issues uses so-called fluorogen-activating proteins (Schwartz et al. 2015; Plamont et al. 2016): Instead of waiting for the protein to fold completely for the chromophore to be activated, a synthetic chromophore is provided in the growth media. Fluorescence is activated by the genetically-encoded protein, which is simpler and smaller than fluorescent proteins and does not require chromophore maturation.

3.4 Image analysis

Image analysis was discussed extensively in chapter 4. New techniques for cell segmentation and tracking were developed in recent years, and the emergence of machine learning methods is a game-changer in the field, not only in cell segmentation but also in other sub-domains of image analysis (LeCun et al. 2015). Beyond the cell segmentation methods proposed in this thesis and in the U-net architecture (Ronneberger et al. 2015), machine learning could be used for robust tracking and lineage reconstruction, as such methods have been applied to similar problems in other areas of image analysis (Wang & Yeung 2013; Shen & Liu 2008). Robust cell segmentation, tracking and lineage reconstruction combined with the single-cell

optogenetic control procedure described earlier, would open the door to the study of epigenetic and transmission phenomena in gene regulation networks within populations of cells.

3.5 Cloning and genome engineering

In our study, the cellular chassis is considered part of the study. Indeed, now that a cellular chassis has been developed for long-term control in Escherichia coli with the lac and tet systems, other circuits can be constructed and transformed in this host cell for control experiments. The development of such chassis strains is a tedious process and can be accelerated by recently developed techniques in DNA manipulation and cloning. This also applies to the development of synthetic circuits for control, or to the integration of actuation and observation elements in the gene regulation networks to study. Although external control platforms require much less modifications of the genome or less complex synthetic networks, cloning and genome engineering are still necessary, and any progress made in this direction would benefit the study of those systems.

The recent revolutions of the ZFN, TALE and CRISPR/Cas9 systems (Gaj et al. 2013) are of course possible ways to speed up the control workflow. But not only can those systems be used for genome engineering, they can also be used as custom recombinases, transposases, and more importantly transcription factors (Kabadi & Gersbach 2014). The possibility to design custom-target transcription factors would make it possible to control the dynamics of arbitrary genes in gene regulation networks. Recently, a photoactivable version of the CRISPR/Cas9 system was developed for optogenetic genome editing (Nihongaki et al. 2015), and one can expect the design of custom optogenetic transcription factors to become possible in the near future.

New cloning techniques developed over the past decade, like the Golden Gate technique (Engler et al. 2008) and its standardized version the Modular Cloning technique (Weber et al. 2011), or the Gibson assembly (Gibson et al. 2009) made it faster and easier to assemble long circuits into plasmids, which in turn can speed up the workflow of control-based analysis of genetic networks. De novo synthesis also became one of the preferred method for developing new circuits (Kahl & Endy 2013) thanks to a drop in price over the past few years, and we can hope that these methods will eventually replace the tedious process of parts assembly in the long run.

4 Perspectives

In silico feedback control of biological processes in vivo has just emerged recently as I have detailed in the state-of-the-art of this dissertation, but it already sparked interest in the community (Del Vecchio et al. 2016) with possible applications speculated in medicine (Menolascina et al. 2012), optimal experimental design (Ruess et al. 2015; Ruess et al. 2013), metabolic engineering (Shiue & Prather 2012), or even space exploration (Menezes et al. 2015). It has to be noted though, that the work I presented here should be seen as a proof-of-concept implementation of a feedback control system for gene expression of multistable

systems in bacteria, and significant work remains to be done to actually apply feedback control systems for the generation of precise perturbations of any chosen biological system. In this final section however, I will detail possible long-term applications of in silico cybergenetics.

The recent success of systems biology illustrates the importance of investigating not only the structure and function, but also the dynamics of biological systems. In this respect, recent progresses in cloning, genome engineering and custom transcription factors fabrication should make it possible to control arbitrary genes in endogenous gene regulation networks to dissect their dynamics. Another interesting domain progressing quickly is the field of iterative, optimal experimental design. Given some initial information on a biological system, along with a defined objective (e.g. estimating the parameters of a model of the system, being able to perform the best predictions of a given situation etc.), it is possible to optimize the choice of experiments to be conducted to reach that objective. This method is particularly relevant to reverse-engineer complex genetic networks for which intuition is not sufficient to estimate the best course of action to extract a maximum of information out of all the expectable consequences of given perturbations. Control platforms could be used as automated tools for optimal experimental design to achieve on-the-fly, optimal, and possibly high-throughput scrutiny of endogenous networks.

Another closely related area of research that could benefit from external control of gene expression is circuit testing and characterization. By automating the process and parallelizing the platforms, library of circuits could be tested in different settings, with different dynamic perturbations. A significant obstacle for the construction of novel functions in synthetic biology is the integration of low-level modules together to obtain high-level signal processing functions in the biochemical network. Because the dynamics of the different modules are often poorly characterized, as we saw with the genetic toggle switch in this thesis, assembling them together to produce a desired function is often a difficult task. Parallelized control platforms could be used to characterize precisely those modules, and the inferred models could be used to predict the behavior of assembled circuits.

In a different domain, bioreactors used for industrial production of chemicals of interest are notoriously difficult to calibrate and batch-to-batch reproducibility and yield maximization are often imperfect (Polizzi & Kontoravdi 2015). External control of gene expression could help solve these problems (Milias-Argeitis et al. 2016) and lower the costs of production of the chemicals by minimizing chemical inducers and culture costs. Several problems remain to scale up the processes described in batch culture control of gene expression, especially input delivery, but this should be one of the most direct contributions of current gene expression control platforms to the industry.

One related domain where control platforms could also be applied in the near future is metabolic engineering. It has been shown that the yield of synthetic metabolic pathways can be optimized by integrating control loops between the metabolic layer and the genetic layer of the patway (Zhang et al. 2012; Oyarzùn & Stan 2012). Optogenetic tools for driving enzymatic activity already exist (Beyer et al. 2015) and a plethora of biosensors for various

metabolites have been developed over the years (Okumoto et al. 2012; Su et al. 2011; Polizzi & Kontoravdi 2015). The seminal work of (Toettcher et al. 2011b) in this regard proves that similar intracellular processes can be controlled over short timescales, and it should be possible to adapt control platforms to the synthesis of chemicals of interest like biofuels.

Finally the recent development of infrared fluorescent proteins (Filonov et al. 2011) and photoacoustic tomography (Yao et al. 2015) makes it possible to observe gene expression in live mammalians, and with the development of those techniques, one of the possible future domains of applications for external gene expression control platforms is real-time control of gene expression in live animals, with possible biomedical applications.

To summarize, in silico control of intracellular processes is still in its infancy, and we are still a long way from routine utilization of those methods to dissect genetic network dynamics, or to optimally produce chemicals of interest in bulk. On the other hand, this is the closest we have ever been to reaching those goals, and the field of in silico cybergenetics should play a major role in the years to come in the broader domains of synthetic biology and systems biology. Developing a platform for single-cell control of gene expression in bacteria, controlling a landmark bistable genetic network like the genetic toggle switch, and the surprising result of dynamic stabilization in a gene regulation network it entailed, will hopefully contribute to the development of the field into an established discipline.

Bevis, B.J. & Glick, B.S., 2002. Rapidly maturing variants of the Discosoma red fluorescent protein (DsRed). *Nature biotechnology*, 20(1), pp.83–7.

Beyer, H.M. et al., 2015. Optogenetic control of signaling in mammalian cells. *Biotechnology Journal*, 10(2), pp.273–283.

Dénervaud, N. et al., 2013. A chemostat array enables the spatio-temporal analysis of the yeast proteome. *Proceedings of the National Academy of Sciences of the United States of America*, 110(39), pp.15842–7.

Engler, C., Kandzia, R. & Marillonnet, S., 2008. A one pot, one step, precision cloning method with high throughput capability. *PLoS ONE*, 3(11).

Filonov, G.S. et al., 2011. Bright and stable near-infrared fluorescent protein for in vivo imaging. *Nature Biotechnology*, 29(8), pp.757–761.

Fiore, G. et al., 2013. An experimental approach to identify dynamical models of transcriptional regulation in living cells. *Chaos*, 23(2).

Fiore, G., Perrino, G. & Bernardo, M., 2015. *In-vivo real-time control of gene expression : a comparative analysis of feedback control strategies in yeast - Supporting Informations Appendix - GAL1 promoter mathematical model derivation*,

Fisher, A.C. & DeLisa, M.P., 2008. Laboratory evolution of fast-folding green fluorescent protein using secretory pathway quality control. *PLoS ONE*, 3(6), pp.1–7.

Fracassi, C. et al., 2016. Automatic Control of Gene Expression in Mammalian Cells. *ACS synthetic biology*, 5(4), pp.296–302.

Gaj, T., Gersbach, C.A. & Barbas, C.F., 2013. ZFN, TALEN, and CRISPR/Cas-based methods for genome engineering. *Trends in Biotechnology*, 31(7), pp.397–405.

Gibson, D.G. et al., 2009. Enzymatic assembly of DNA molecules up to several hundred kilobases. *Nature methods*, 6(5), pp.343–5.

Gunaydin, L. a et al., 2010. Ultrafast optogenetic control. *Nature Neuroscience*, 13(3), pp.387–392.

Kabadi, A.M. & Gersbach, C.A., 2014. Engineering synthetic TALE and CRISPR/Cas9 transcription factors for regulating gene expression. *Methods*, 69(2), pp.188–197.

Kaberniuk, A.A., Shemetov, A.A. & Verkhusha, V. V, 2016. A bacterial phytochrome-based optogenetic system controllable with near-infrared light. *Nature Methods*, 13(7), pp.591–597.

Kahl, L.J. & Endy, D., 2013. A survey of enabling technologies in synthetic biology. *Journal of Biological Engineering*, 7(1), p.13.

LeCun, Y., Bengio, Y. & Hinton, G., 2015. Deep learning. *Nature*, 521(7553), pp.436–444.

Levskaya, A. et al., 2009. Spatiotemporal control of cell signalling using a light-switchable protein interaction. *Nature*, 461(7266), pp.997–1001.

Levskaya, A. et al., 2005. Synthetic biology: Engineering Escherichia coli to see light. *Nature*, 438(7067), pp.441–442.

Li, N., Hsu, C.-H. & Folch, A., 2005. Parallel mixing of photolithographically defined nanoliter volumes using elastomeric microvalve arrays. *Electrophoresis*, 26(19), pp.3758–64.

Melendez, J. et al., 2014. Real-time optogenetic control of intracellular protein concentration in microbial cell cultures. *Integrative biology*, 6(3), pp.366–72.

Menezes, A.A. et al., 2015. Grand challenges in space synthetic biology. *Journal of The Royal Society Interface*, 12(113), p.20150803.

Menolascina, F., Siciliano, V. & Di Bernardo, D., 2012. Engineering and control of biological systems: A new way to tackle complex diseases. *FEBS Letters*, 586(15), pp.2122–2128.

Milias-Argeitis, A. et al., 2016. Automated optogenetic feedback control for precise and robust regulation of gene expression and cell growth. *Nature communications*, 7(May), p.12546.

Milias-Argeitis, A. et al., 2011. In silico feedback for in vivo regulation of a gene expression circuit. *Nature biotechnology*, 29(12), pp.1114–6.

Mishin, A.S. et al., 2015. Novel uses of fluorescent proteins. *Current Opinion in Chemical Biology*, 27, pp.1–9.

Nihongaki, Y. et al., 2015. Photoactivatable CRISPR-Cas9 for optogenetic genome editing. *Nat Biotechnol*, 33(7), pp.755–760.

Okumoto, S., Jones, A. & Frommer, W.B., 2012. Quantitative imaging with fluorescent biosensors. *Annual review of plant biology*, 63, pp.663–706.

Oyarzùn, D.A. & Stan, G.-B., 2012. Synthetic gene circuits for metabolic control: design trade-offs and constraints. *Journal of The Royal Society Interface*, 1400.

Plamont, M.-A. et al., 2016. Small fluorescence-activating and absorption-shifting tag for tunable protein imaging in vivo. *Proceedings of the National Academy of Sciences*, 113(3), pp.497–502.

Polizzi, K.M. & Kontoravdi, C., 2015. Genetically-encoded biosensors for monitoring cellular stress in bioprocessing. *Current Opinion in Biotechnology*, 31, pp.50–56.

Querard, J. et al., 2015. Photoswitching Kinetics and Phase-Sensitive Detection Add Discriminative Dimensions for Selective Fluorescence Imaging. *Angewandte Chemie International Edition*, 54(9), pp.2633–2637.

Renault, R. et al., 2015. Combining Microfluidics, Optogenetics and Calcium Imaging to Study Neuronal Communication In Vitro S. Martinoia, ed. *PLOS ONE*, 10(4), p.e0120680.

Ronneberger, O., Fischer, P. & Brox, T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. *ArXiv*.

Ruess, J. et al., 2015. Iterative experiment design guides the characterization of a light-

inducible gene expression circuit. *Proceedings of the National Academy of Sciences of the United States of America*, 112(26), pp.8148–8153.

Ruess, J., Milias-Argeitis, A. & Lygeros, J., 2013. Designing experiments to understand the variability in biochemical reaction networks. *Journal of the Royal Society, Interface / the Royal Society*, 10(88), p.20130588.

Schwartz, S.L. et al., 2015. Fluorogen-activating proteins provide tunable labeling densities for tracking Fc??RI independent of IgE. *ACS Chemical Biology*, 10(2), pp.539–546.

Shen, S. & Liu, Y., 2008. Efficient multiple faces tracking based on Relevance Vector Machine and Boosting learning. *Journal of Visual Communication and Image Representation*, 19(6), pp.382–391.

Shimizu-Sato, S. et al., 2002. A light-switchable gene promoter system. *Nature biotechnology*, 20(10), pp.1041–4.

Shiue, E. & Prather, K.L.J., 2012. Synthetic biology devices as tools for metabolic engineering. *Biochemical Engineering Journal*, 65, pp.82–89.

Sorre, B. et al., 2014. Encoding of Temporal Signals by the TGF-β Pathway and Implications for Embryonic Patterning. *Developmental Cell*, 30(3), pp.334–342.

Strickland, D., Moffat, K. & Sosnick, T.R., 2008. Light-activated DNA binding in a designed allosteric protein. *Proceedings of the National Academy of Sciences*, 105(31), pp.10709–10714.

Su, L. et al., 2011. Microbial biosensors: A review. *Biosensors and Bioelectronics*, 26(5), pp.1788–1799.

Sundararajan, N., Kim, D. & Berlin, A. a, 2005. Microfluidic operations using deformable polymer membranes fabricated by single layer soft lithography. *Lab on a chip*, 5(3), pp.350–4.

Tabor, J.J., Levskaya, A. & Voigt, C. a, 2011. Multichromatic control of gene expression in Escherichia coli. *Journal of molecular biology*, 405(2), pp.315–24.

Toettcher, J.E. et al., 2011a. *Light Control of Plasma Membrane Recruitment Using the Phy–PIF System*,

Toettcher, J.E. et al., 2011b. Light-based feedback for controlling intracellular signaling dynamics. *Nature methods*, 8(10), pp.837–9.

Uhlendorf, J. et al., 2012. Long-term model predictive control of gene expression at the population and single-cell levels. *Proceedings of the National Academy of Sciences of the United States of America*, 109(35), pp.14271–6.

Unger, M.A. et al., 2000. Monolithic Microfabricated Valves and Pumps by Multilayer Soft Lithography. *Science*, 288(5463), pp.113–116.

Del Vecchio, D., Dy, A.J. & Qian, Y., 2016. Control theory meets synthetic biology. *Journal of The Royal Society Interface*, 13(120).

Wang, N. & Yeung, D.-Y., 2013. Learning a Deep Compact Image Representation for Visual Tracking. In C. J. C. Burges et al., eds. *Advances in Neural Information Processing Systems 26*. Curran Associates, Inc., pp. 809–817.

Weber, E. et al., 2011. A modular cloning system for standardized assembly of multigene constructs. *PloS one*, 6(2), p.e16765.

Yang, X. et al., 2013. A light-inducible organelle-targeting system for dynamically activating and inactivating signaling in budding yeast. *Molecular Biology of the Cell*, 24(15), pp.2419–2430.

Yao, J. et al., 2015. Multiscale photoacoustic tomography using reversibly switchable bacterial phytochrome as a near-infrared photochromic probe. *Nature Methods*, 13(November), pp.1–9.

Zhang, F., Carothers, J.M. & Keasling, J.D., 2012. Design of a dynamic sensor-regulator system for production of chemicals and fuels derived from fatty acids. *Nature Biotechnology*, (February), pp.1–7.

Appendix A
List of plasmids and strains

Table of Contents

1 Plasmids 2
1.1 Backbones 2
1.2 Level 0 – Parts 3
1.3 Level 1 – Transcription units 4
1.4 Level 2 – Circuits 6
2 Strains 7

The MoClo backbones, parts, transcription units and cricuits are presented here, as well as the chassis strains that were developed in this PhD as well as those that were transformed with synthetic circuit plasmids for control. Not all strains were used in this PhD nor were all Level 2 MoClo plasmids transformed into specific strains.

1 Plasmids

1.1 Backbones

The backbones are used to assemble the different levels of the MoClo process into. They are organized such that they always assemble in the same order.

Name	Level	Short description	Res	Origin	copy nb.
pL0-P	0	Level 0 standard promoter backbone	Spec	pUC	500-700
pL0-U	0	Level 0 standard RBS backbone	Spec	pUC	500-700
pL0-SC	0	Level 0 standard CDS backbone	Spec	pUC	500-700
pL0-T	0	Level 0 standard terminator backbone	Spec	pUC	500-700
pL1-F1	1	Level 1 standard #1 transcription unit	Amp	pUC	500-700
pL1-F2	1	Level 1 standard #2 transcription unit	Amp	pUC	500-700
pL1-F3	1	Level 1 standard #3 transcription unit	Amp	pUC	500-700
pL1-F4	1	Level 1 standard #4 transcription unit	Amp	pUC	500-700
pL1-F5	1	Level 1 standard #5 transcription unit	Amp	pUC	500-700
pL1-F6	1	Level 1 standard #6 transcription unit	Amp	pUC	500-700
pL1-F7	1	Level 1 standard #7 transcription unit	Amp	pUC	500-700
pL1-F1a	1	Level 1 low-copy #1 transcription unit	Cm	ACYC	10-12
pL1-F2a	1	Level 1 low-copy #2 transcription unit	Cm	ACYC	10-12
pL1-F3a	1	Level 1 low-copy #3 transcription unit	Cm	ACYC	10-12
pL1-F4a	1	Level 1 low-copy #4 transcription unit	Cm	ACYC	10-12
pL1-F1c	1	Level 1 low-copy #1 transcription unit	Cm	CDF	15-20
pL1-F2c	1	Level 1 low-copy #2 transcription unit	Cm	CDF	15-20
pL1-F3c	1	Level 1 low-copy #3 transcription	Cm	CDF	15-20

pL1-F4c	1	unit Level 1 low-copy #4 transcription unit	Cm	CDF	15-20
pL2-1	2	Level 2 standard circuit backbone	Kan	pUC	500-700
pL2-1a	2	Level 2 low-copy circuit backbone	Cm	ACYC	10-12
pL2-1cc	2	Level 2 low-copy circuit backbone	Cm	CDF	15-20
pL2-1cs	2	Level 2 low-copy circuit backbone	Spec	CDF	15-20

Orange backbones were developed in-house to circumvent burden problems.

1.2 Level 0 – Parts

Level 0 parts are the basic building blocks of MoClo circuits. In our case a MoClo transcription unit is composed of 4 level 0 parts: Promoter (pL0-P plasmids), RBS/5' UTR (pL0-U plasmids), Gene/coding sequence (pL0-SC plasmids), and Terminator (pL0-T). The overhangs of each of those parts are organized such that they assemble in the right order.

Part no.	Designation (Geneious)	Designation (short)	Description
1	pL0-P LacO	pLac	lac promoter
2	pL0-P TetO	pTet	tet promoter
3	pL0-P J23119	j23119	consitutive promoter
4	pL0-T pFAB801	FAB801	Terminator
5	pL0-T pFAB822	FAB822	Terminator
6	pL0-T pFAB816	FAB816	Terminator
7	pL0-T pFAB815	FAB815	Terminator
8	pL0-SC mKate2	mKate2	mKate2 fluorescent protein
9	pL0-SC mEGFP	mEGFP	mEGFP fluorescent protein
10	pL0-SC EYFP	EYFP	EYFP fluorescent protein
11	pL0-SC LacI	LacI	LacI transcription factor
12	pL0-SC TetR	TetR	TetR transcription factor
13	pL0-U B0030	B0030	Ribosome binding site
14	pL0-U B0031	B0031	Ribosome binding site
15	pL0-U B0032	B0032	Ribosome binding site
16	pL0-U B0034	B0033	Ribosome binding site
17	pl0-SC mKate2::LacI SL	mKate2::LacI	mKate2-LacI fusion protein
18	pl0-SC mKate2::LacI LL		mKate2-LacI fusion protein
19	pl0-SC LacI::mKate2 SL		mKate2-LacI fusion protein
20	pl0-SC LacI::mKate2 LL		mKate2-LacI fusion protein
21	pl0-SC mEGFP::tetR SL	mEGFP::TetR	mEGFP-TetR fusion protein
22	pl0-SC mEGFP::tetR LL		mEGFP-TetR fusion protein
23	pl0-SC TetR::mEGFP SL		mEGFP-TetR fusion protein
24	pl0-SC TetR::mEGFP LL		mEGFP-TetR fusion protein
25	pl0-P J23110	j23110	consitutive promoter
26	pl0-P J23111	j23111	consitutive promoter
27	pl0-P J23112	j23112	consitutive promoter
28	pl0-P J23113	j23113	consitutive promoter
29	pl0-P J23114	j23114	consitutive promoter

30	pL0-U B0033	B0034	Ribosome binding site
31	pL0-U B0035	B0035	Ribosome binding site
32	pL0-U RBSswapAarI	RBS swapper AarI	RBS swapper
33	pL0-U RBSswapEsp3I	RBS swapper Esp3I	RBS swapper
34	pL0-SC_LacI-mKate2-OPE34	LacI_mKate2	mKate2-LacI operon
35	pL0-SC_TetR-mEGFP-OPE34	TetR_mEGFP	mEGFP-TetR operon
36	pL0-SC_LacI-mKate2-STPSTRT		mKate2-LacI operon
37	pL0-SC_LacI-mKate2-STPSTRT		mEGFP-TetR operon
38	pL0-LacI-mKate2_opeRBSswap		mKate2-LacI operon
39	pL0-TetR-mEGFP_opeRBSswap		mEGFP-TetR operon

1.3 Level 1 – Transcription units

Transcription units, made of different level 0 parts, express one or, in the case of operons, several genes downstream of a promoter. They are assembled together into level 2 circuits to form synthetic genetic networks. The level 1 assemblies are made into different backbones that are designed to assemble into a specific order: The overhangs on the backbones are organized such that a pL1-F1 transcription units will ligate to level 2 backbone on one end, and to a pL1-F2 unit on the other. A pL1-F2 unit assembles with a pL1-F1 and a pL1-F3 unit and so on until pL1-F7, which ligates back with the level 2 backbone, thus closing the assembly. Of course not all 7 level 1 units are necessary to construct a level 2 circuits, and one can assemble only one transcription unit into a level 2 backbone if they wish.

Name	Prom	RBS	CDS	Term	bckbn	Res	Ori	Description
	pL0-P	pL0-U	pL0-SC	pL0-T	pL1-FX			
1.1	pLac	B0034	mKate2	FAB801	pL1-F2	Amp	pUC	Single FP transcription unit
1.2	pTet	B0034	mKate2	FAB801	pL1-F2	Amp	pUC	Single FP transcription unit
1.3	j23119	B0034	mKate2	FAB801	pL1-F2	Amp	pUC	Single FP transcription unit
1.4	pLac	B0034	mEGFP	FAB822	pL1-F2	Amp	pUC	Single FP transcription unit
1.5	pTet	B0034	mEGFP	FAB822	pL1-F2	Amp	pUC	Single FP transcription unit
1.6	j23119	B0034	mEGFP	FAB822	pL1-F2	Amp	pUC	Single FP transcription unit
1.7	pLac	B0034	EYFP	FAB822	pL1-F2	Amp	pUC	Single FP transcription unit
1.8	pTet	B0034	EYFP	FAB822	pL1-F2	Amp	pUC	Single FP transcription unit
1.9	j23119	B0034	EYFP	FAB822	pL1-F2	Amp	pUC	Single FP transcription unit
1.10	pLac	B0034	mEGFP::TetR	FAB815	pL1-F2	Cm	ACYC	Single FP transcription unit
1.11	pTet	B0034	mKate2::LacI	FAB816	pL1-F2a	Cm	ACYC	Single FP transcription unit
1.12	j23114	B0034	mKate2::LacI	FAB816	pL1-F3a	Cm	ACYC	Single FP transcription unit
1.13	CmR resistance cassette flanked by FRT sites				pL1-F1	Amp	pUC	Resistance cassette for chromosomal integration
1.14	CmR resistance cassette flanked by FRT sites				pL1-F2	Amp	pUC	Resistance cassette for chromosomal integration

1.15		CmR resistance cassette flanked by FRT sites			pL1-F5	Amp	pUC	Resistance cassette for chromosomal integration
1.16		KanR resistance cassette flanked by FRT sites			pL1-F4	Amp	pUC	Resistance cassette for chromosomal integration
1.17		KanR resistance cassette flanked by FRT sites			pL1-F2	Amp	pUC	Resistance cassette for chromosomal integration
1.19		KanR resistance cassette flanked by FRT sites			pL1-F5	Amp	pUC	Resistance cassette for chromosomal integration
1.20	pLac	B0034	mEGFP::TetR	FAB815	pL1-F4a	Cm	ACYC	pLac-tetRToggle switch branch - Fusion proteins
1.21	pTet	B0034	mKate2::LacI	FAB816	pL1-F3a	Cm	ACYC	pTet-lacIToggle switch branch - Fusion proteins
1.22		intS 5' Homology region			pL1-F1	Amp	pUC	homology region for intS chromosomal integration
1.23		intS 3' Homology region #1			pL1-F3	Amp	pUC	homology region for intS chromosomal integration
1.24		intS 3' Homology region #2			pL1-F6	Amp	pUC	homology region for intS chromosomal integration
1.25		acrB 5' Homology region			pL1-F1	Amp	pUC	homology region for acrA-B knockout
1.26		acrA 3' Homology region #1			pL1-F3	Amp	pUC	homology region for acrA-B knockout
1.27		acrA 3' Homology region #2			pL1-F6	Amp	pUC	homology region for acrA-B knockout
1.28	pLac	B0030	mEGFP::TetR	FAB815	pL1-F4a	Cm	ACYC	pLac-tetRToggle switch branch - Fusion proteins
1.29	pLac	B0031	mEGFP::TetR	FAB815	pL1-F4a	Cm	ACYC	pLac-tetRToggle switch branch - Fusion proteins
1.30	pLac	B0032	mEGFP::TetR	FAB815	pL1-F4a	Cm	ACYC	pLac-tetRToggle switch branch - Fusion proteins
1.31	pLac	B0033	mEGFP::TetR	FAB815	pL1-F4a	Cm	ACYC	pLac-tetRToggle switch branch - Fusion proteins
1.32	pLac	B0035	mEGFP::TetR	FAB815	pL1-F4a	Cm	ACYC	pLac-tetRToggle switch branch - Fusion proteins
1.33	pTet	B0030	mKate2::LacI	FAB816	pL1-F3	Cm	ACYC	pTet-lacIToggle switch branch - Fusion proteins
1.34	pTet	B0031	mKate2::LacI	FAB816	pL1-F3a	Cm	ACYC	pTet-lacIToggle switch branch - Fusion proteins
1.35	pTet	B0032	mKate2::LacI	FAB816	pL1-F3a	Cm	ACYC	pTet-lacIToggle switch branch - Fusion proteins
1.36	pTet	B0033	mKate2::LacI	FAB816	pL1-F3a	Cm	ACYC	pTet-lacIToggle switch branch - Fusion proteins
1.37	pTet	B0035	mKate2::LacI	FAB816	pL1-F3a	Cm	ACYC	pTet-lacIToggle switch branch - Fusion proteins
1.38	pLac	RBS swapper AarI	mEGFP::TetR	FAB815	pL1-F3a	Cm	ACYC	plac-tetR "wildcard"Toggle switch branch - Fusion
1.39	pTet	RBS swapper Esp3I	mKate2::LacI	FAB816	pL1-F4a	Cm	ACYC	pTet-lacI "wildcard"Toggle switch branch - Fusion
1.40	pLac	B0030	TetR_mEGFP	FAB815	pL1-F1a	Cm	ACYC	pLac-tetRToggle switch branch - operon
1.41	pLac	B0031	TetR_mEGFP	FAB815	pL1-F1a	Cm	ACYC	pLac-tetRToggle switch branch - operon
1.42	pLac	B0032	TetR_mEGFP	FAB815	pL1-F1a	Cm	ACYC	pLac-tetRToggle switch branch - operon
1.43	pLac	B0033	TetR_mEGFP	FAB815	pL1-F1a	Cm	ACYC	pLac-tetRToggle switch branch - operon
1.44	pLac	B0034	TetR_mEGFP	FAB815	pL1-F1a	Cm	ACYC	pLac-tetRToggle switch branch - operon
1.45	pLac	B0035	TetR_mEGFP	FAB815	pL1-F1a	Cm	ACYC	pLac-tetRToggle switch branch - operon
1.46	pTet	B0030	LacI_mKate2	FAB816	pL1-	Cm	ACYC	pTet-lacIToggle switch branch - operon

					F2a			
1.47	pTet	B0031	LacI_mKate2	FAB816	pL1-F2a	Cm	ACYC	pTet-lacIToggle switch branch - operon
1.48	pTet	B0032	LacI_mKate2	FAB816	pL1-F2a	Cm	ACYC	pTet-lacIToggle switch branch - operon
1.49	pTet	B0033	LacI_mKate2	FAB816	pL1-F2a	Cm	ACYC	pTet-lacIToggle switch branch - operon
1.50	pTet	B0034	LacI_mKate2	FAB816	pL1-F2a	Cm	ACYC	pTet-lacIToggle switch branch - operon
1.51	pTet	B0035	LacI_mKate2	FAB816	pL1-F2a	Cm	ACYC	pTet-lacIToggle switch branch - operon
1.52	pLac	RBS swapper AarI	TetR_mEGFP	FAB815	pL1-F1a	Cm	ACYC	plac-tetR "wildcard"Toggle switch branch - operon
1.53	pTet	RBS swapper Esp3I	LacI_mKate2	FAB816	pL1-F2a	Cm	ACYC	pTet-lacI "wildcard"Toggle switch branch - operon

1.4 Level 2 – Circuits

Level 2 circuits are assembled from level 1 transcription units are the final product of Modular Cloning. Some of the plasmids presented here were then transformed into the chassis strains. The orange element represents toggle switches.

Name	Pos. 1	Pos. 2	3	4	5	6	7	Res	Origin	Description
2.1		1.1						Cm	ACYC	Single transcription unit
2.2		1.2						Cm	ACYC	Single transcription unit
2.3	1.22		1.21	1.20	1.19	1.24		Cm	ACYC	Toggle switch - fusion proteins
2.4	1.22	1.4		1.20	1.19	1.24		Cm	ACYC	Fusion proteins testing
2.5	1.22	1.2	1.21		1.19	1.24		Cm	ACYC	Fusion proteins testing
2.6	1.22	1.1			1.19	1.24		Cm	ACYC	Single transcription unit
2.7	1.22	1.2			1.19	1.24		Cm	ACYC	Single transcription unit
2.8	1.25	1.17	1.26					Cm	ACYC	acrA-B knockout
2.9	1.22	1.4			1.19	1.24		Cm	ACYC	Single transcription unit
2.10 to 2.45	1.40 to 1.45	1.46 to 1.51						Cm	ACYC	Toggle switches - operons
2.46	1.52	1.53						Cm	ACYC	Toggle switch - "wildcard"

The numbering of the circuits of the library of toggle switches is as follows:

	pTet-mKate2-LacI branches						
pLac-mEGFP-TetR branches		1.46	1.47	1.48	1.49	1.50	1.51
	1.40	2.10	2.11	2.12	2.13	2.14	2.15
	1.41	2.16	2.17	2.18	2.19	2.20	2.21
	1.42	2.22	2.23	2.24	2.25	2.26	2.27
	1.43	2.28	2.29	2.30	2.31	2.32	2.33
	1.44	2.34	2.35	2.36	2.37	2.38	2.39
	1.45	2.40	2.41	2.42	2.43	2.45	2.46

The greyed-out circuits could not be constructed. The orange circuit 2.31 is the selected circuit for the control experiments.

2 Strains

Name	From	Plasmid	Chromosome integr.	Gene deletions	Description
bPH_103	BW25113			fliA	fliA- strain from the Keio collection
bPH_104	bPH_103			fliA, lacY	fliA- lacY- chassis strain
bPH_121	bPH_103		Z1: LacI+, TetR+	fliA	fliA- lacI+ tetR+ chassis strain
bPH_122	bPH_104		Z1: LacI+, TetR+	fliA, lacY	fliA- lacY- lacI+ tetR+ chassis strai
bPH_123	bPH_121	2.1	Z1: LacI+, TetR+	fliA	Population control strain
bPH_124	bPH_121	2.2	Z1: LacI+, TetR+	fliA	Population control strain
bPH_125	bPH_122	2.1	Z1: LacI+, TetR+	fliA, lacY	Population control strain
bPH_126	bPH_122	2.2	Z1: LacI+, TetR+	fliA, lacY	Population control strain
bPH_127	bPH_104			fliA, lacY, acrA, acrB	fliA- lacY- acrAB- chassis strain
bPH_128	bPH_122		Z1	fliA, lacY, acrA, acrB	fliA- lacY- acrAB- lacI+ tetR+ chass strain
bPH_130	bPH_128	2.2	Z1	fliA, lacY, acrA, acrB	Population control strain
bPH_129	bPH_127	2.3		fliA, lacY, acrA, acrB	Fusion toggle switch strain
bPH_131	bPH_127	2.4		fliA, lacY, acrA, acrB	Fusion tests strain
bPH_132	bPH_127	2.5		fliA, lacY, acrA, acrB	Fusion tests strain

bPH_133	bPH_128		Z1, 2.6	fliA, lacY, acrA, acrB	Chromomal integration - single TU
bPH_134	bPH_128		Z1, 2.7	fliA, lacY, acrA, acrB	Chromomal integration - single TU
bPH_135	bPH_127		2.9	fliA, lacY, acrA, acrB	Chromomal integration - single TU
bPH_136	bPH_127		2.3	fliA, lacY, acrA, acrB	Chromomal integration - Fusion toggle
bPH_137	bPH_127	2.38		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_138	bPH_127	2.10		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_139	bPH_127	2.17		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_140	bPH_127	2.46		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_141	bPH_127	2.24		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_142	bPH_127	2.31		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_143	bPH_127	2.15		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_144	bPH_127	2.16		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_145	bPH_127	2.22		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_146	bPH_127	2.27		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_147	bPH_127	2.13		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_148	bPH_127	2.18		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_149	bPH_127	2.28		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_150	bPH_127	2.37		fliA, lacY, acrA, acrB	Toggle switch strain
bPH_151	bPH_127	2.43		fliA, lacY, acrA, acrB	Toggle switch strain