ABSTRACT

DE LUÍS BALAGUER, MARIA DELS ÀNGELS. Computational Learning Strategies for Assessing Modular Influences on Biological Phenotypes. (Under the direction of Dr. Cranos M. Williams.)

Biological organisms are highly complex systems composed of multiple levels of organization. Modeling biological systems is a crucial approach for describing processes from these organisms, and understanding characteristics of critical importance. Modeling approaches have traditionally targeted well confined biological processes located at low levels of organization; these include models that describe interactions among genes, metabolites, or biomolecules. New biological and engineering challenges, however, demand the comprehension of larger and more complex processes, and overall, the study of the interaction among processes at several levels of abstraction. Combining multiple models of these different levels of abstraction to generate multihierarchical models has been identified as a path to describe meaningful phenomena. This goal, however, raises several mathematical and computational challenges. The high complexity associated to biological models impedes the combination of large numbers of models. Mathematical and computational strategies are needed to provide more intuitive information of the models, simplify models, and relate models at several levels of abstraction and to fulfill the demands of the new challenges. Furthermore, I are sometimes not able to relate the low level models that have been developed over the years to meaningful higher level biological phenomena. This results in a need for better understanding how lower level biomolecular components impact higher level phenotypic characteristics.

I present in this document two computer based algorithms that aim to 1) provide functional information needed to understand the underlying structure of models, which can be a critical tool in comprehending the impact of biomolecular components on higher level phenomena, and 2) find a relationship between models of low level processes and more abstract meaningful phenomena, by identifying key groups of components to control or manipulate these phenomena. Particularly, the first algorithm decomposes dynamic models into functional units, using fuzzy clustering to identify primary relationships, secondary relationships, and changes of these relationships over time. The second algorithm leverages machine learning tools and experimental data to find a relationship between a model and a phenomena of interest, and it uses sensitivity analysis to find groups of components that can be used to manipulate the phenomena. This approach is used in combination with the decomposition algorithm to identify the functional modules from complex models that can impact those abstract phenomena.

I show the application of the presented algorithms to two different dynamic biological models, the EGF induced MAPK, and the C3 photosynthesis pathway. I validate the results of my
applications by comparing them with previous literature. By showing that the obtained results are similar to others that were experimentally acquired, I demonstrate the potential that these new computational methods can have on the biological perspective.
Computational Learning Strategies for Assessing Modular Influences on Biological Phenotypes

by
Maria dels Àngels de Luís Balaguer

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Electrical Engineering

Raleigh, North Carolina
2013

APPROVED BY:

Dr. Winser E. Alexander
Dr. William W. Edmonson
Dr. Edward Grant
Dr. Steve Campbell

Dr. Cranos M. Williams
Chair of Advisory Committee
DEDICATION

To my family, for their unconditional support and love, that undoubtedly made this possible.
BIOGRAPHY

Maria dels Àngels de Luís Balaguer was born to José Luís Gil and Maria Teresa Balaguer Martí in Valencia, Spain. She grew up in El Puig de Santa Maria, town located near the city of Valencia, where she attended Elementary and Middle School. She graduated from the I.E.S. Puçol High School, Puçol, before attending college at Universitat Politècnica de València, where she obtained her B.S. in Telecommunications Engineering in 2007. She entered the Ph.D. program at North Carolina State University in 2008, where she has conducted her research in Systems Biology.
ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Cranos M. Williams for his priceless advice, dedication, kindness, and wisdom. You made this happen; I will always remember.

I want to thank my committee members, Dr. Winser E. Alexander, Dr. William W. Edmonson, Dr. Edward Grant, and Dr. Steve Campbell, for their advise and guidance through my years of study and research.

I will not forget my labmates Jina Song and Megan Mathews, for the help and support that I have received from them.

I want to acknowledge Kenneth W. Jennings, my boyfriend, for believing in this research, for his valuable help by means of our discussions through my entire Ph.D. program, and for his patience through these last months.

With no doubt I want to thank my parents, José Luís Gil and Maria Teresa Balaguer Martí, for always being understanding, generous, and loving. Last, I want to thank my brother Marc, my sister Maite, and my cousin Asun, for making me feel like I never left our crowded home.
# TABLE OF CONTENTS

LIST OF TABLES ............................................................... vii

LIST OF FIGURES ........................................................... viii

Chapter 1 Introduction ....................................................... 1
  1.1 Overview ................................................................. 1
  1.2 Motivation ............................................................... 2
  1.3 Contributions .......................................................... 4
    1.3.1 Intellectual Merit .................................................. 4
    1.3.2 Broader Impacts ................................................... 4
  1.4 Organization ............................................................ 6

Chapter 2 Tools for Modeling and Analysis of Biological Systems: Background 7
  2.1 Phenotypes Manipulation ............................................. 7
  2.2 Towards Multihierarchical Modeling ................................ 12
  2.3 Modeling Biological Systems ........................................ 13
    2.3.1 EGF Induced MAPK .................................................. 13
    2.3.2 C3 Photosynthesis Model ......................................... 14
    2.3.3 Experimental Data .................................................. 16
  2.4 Modularization of Biological Systems ............................... 16
    2.4.1 Modules in Static Data ............................................ 16
    2.4.2 Model Decomposition of Dynamic Systems ....................... 19
  2.5 Black-Box Modeling .................................................... 21
    2.5.1 Hidden Markov Models ............................................. 22
  2.6 Sensitivity Analysis .................................................... 24
    2.6.1 Sensitivity Analysis Methods ..................................... 24
  2.7 Chapter Summary ........................................................ 26

Chapter 3 Computational Strategies for the Transition towards Multiscale Modeling 27
  3.1 A Computational Framework for Model Decomposition .......... 27
    3.1.1 Introduction ....................................................... 27
    3.1.2 NOTATION .......................................................... 29
    3.1.3 General Steps of a Heuristic Simulation Based Algorithm for Model Decomposition ............................................. 29
    3.1.4 Our Algorithm ..................................................... 31
    3.1.5 VALIDATION OF THE RESULTS: ................................... 37
  3.2 A Computational Framework for Identification of Components of a Model Relevant to a Phenotype ................................. 40
    3.2.1 Introduction ....................................................... 40
    3.2.2 An Algorithm to Relate Model and Phenotype: A SA/Machine Learning Approach ......................................................... 42
  3.3 Chapter Summary ........................................................ 51
LIST OF TABLES

Table 4.1 Description of the data point in the Boreas te-09 photosynthetic response data [22]. ................................................................. 72
Table 4.2 Components with highest impact factors. ................................. 77
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Layers of abstraction of a biological organism</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Black box to model the relationship between a model at the pathway level and a phenotype of an organism</td>
<td>10</td>
</tr>
<tr>
<td>2.3</td>
<td>EGF receptor-induced MAPK [74] model. Modified picture.</td>
<td>14</td>
</tr>
<tr>
<td>2.4</td>
<td>C3 Photosynthesis model from by Laisk et al. in [45]. The original picture from [45] has been modified, to include only the dark reactions</td>
<td>15</td>
</tr>
<tr>
<td>2.5</td>
<td>Clustering process</td>
<td>17</td>
</tr>
<tr>
<td>2.6</td>
<td>Model decomposition review</td>
<td>20</td>
</tr>
<tr>
<td>3.1</td>
<td>Toy example. a) Original data. It feigns to be collected from a model with ( n = 3 ) variables, where ( M = 3 ) different ICs have been simulated, and the integration time has been divided in ( T = 4 ) steps. b) Activity vectors that would be generated in [27]. c) Activity vectors that my algorithm would generate.</td>
<td>33</td>
</tr>
<tr>
<td>3.2</td>
<td>Steps of the algorithm</td>
<td>44</td>
</tr>
<tr>
<td>3.3</td>
<td>Representation of a toy pathway and a phenotype whose relationship needs to be modeled. Measurements are taken from one component of the model and from the phenotype. These measurements are used to estimate the parameters of the HMM (the black box).</td>
<td>45</td>
</tr>
<tr>
<td>3.4</td>
<td>Toy example to show how the level of discretization of the measurements leads to particular sizes of the state transition probability matrix, and the observation probability matrix.</td>
<td>46</td>
</tr>
<tr>
<td>3.5</td>
<td>Estimation of the HMM’s parameters by minimizing the error between the measured and the estimated phenotype values, given a sequence of a component’s concentration values.</td>
<td>46</td>
</tr>
<tr>
<td>4.1</td>
<td>Simulation of the EGF receptor-induced MAPK, by Schoebel et al. [74] run in XPP. ICs of the trajectories provided by BioModels [50].</td>
<td>56</td>
</tr>
<tr>
<td>4.2</td>
<td>Modules and submodules of the EGF model. Some of the components that appear in several modules have been highlighted, as well as some reactions that are intermodulars. Each of the components highlighted has been circled with a different color.</td>
<td>57</td>
</tr>
<tr>
<td>4.3</td>
<td>Number of clusters returned by the Silhouette index at each time point for the EGF induced MAPK model. The dashed horizontal bar indicates the average (rounded) of these values over time.</td>
<td>59</td>
</tr>
<tr>
<td>4.4</td>
<td>Primary and secondary memberships of the components in the EGF model for ( \rho = \mu(d_k) ). Columns represent components, and rows represent clusters.</td>
<td>60</td>
</tr>
<tr>
<td>4.5</td>
<td>Decomposition of the EGF induced MAPK [74] in primary and secondary relationships. Decomposition corresponding to the clustering in Fig. 4.4.</td>
<td>61</td>
</tr>
</tbody>
</table>
Figure 4.6 Transient of the relationship between components 22 and 33, located in modules 1 and 6 respectively. Index $T_{22,33}$ indicates if components 22 (Grb2) and 33 ((EGF- EGFR*)2-GAP-Shc*-Grb2) were primarily clustered together, secondarily clustered together, or not clustered together at each sampling time. ................................................................. 63

Figure 4.7 Transient of the relationship between components 22 and 27, located in modules 1 and 2 respectively. Index $T_{22,27}$ indicates if components 22 (Grb2) and 27 ((EGF- EGFR*)2-GAP-Grb2-Sos-Ras-GDP) were primarily clustered together, secondarily clustered together, or not clustered together at each sampling time. ................................................................. 63

Figure 4.8 Transient of the relationship between components 34 and 43, located in modules 6 and 2 respectively. Index $T_{34,43}$ indicates if components 34 ((EGF-EGFR*)2-GAP-Shc*-Grb2) and 43 (Ras-GTP*) were primarily clustered together, secondarily clustered together, or not clustered together at each sampling time. ................................................................. 64

Figure 4.9 Simulation of the C3 Photosynthesis model by Laisk et al. in [45][46] run in XPP. ICs of the trajectories provided in BioModels [50]. ......................................................... 66

Figure 4.10 Number of clusters returned by the Silhouette index at each time point for the C3 photosynthesis model. The dashed horizontal bar indicates the average (rounded) of these values over time. ......................................................... 67

Figure 4.11 Primary and secondary memberships of the components in the C3 photosynthesis model for $\rho = \frac{1}{2}\mu(d_k)$. Columns represent components, and rows represent clusters. ................................................................. 68

Figure 4.12 Decomposition of the photosynthesis model. Picture taken from [89] (reduced). The numbers in the figure identify each reaction. The metabolites in different compartments, i.e. cytosol and stroma, are represented separately in the model. The shaded boxes indicate the my decomposition. Decomposition corresponding to the clustering in Fig. 4.11. ......................................................... 69

Figure 4.13 Measured photosynthetic rate, predicted photosynthetic rate, and measured CO$_2$. Amplitudes presented to scale of the discretization level (not real values in [millimoles]/[meter$^{-2}$][second$^{-1}$] and [parts per million]). ......................................................... 73

Figure 4.14 Predicted photosynthetic rate and CO$_2$ trajectory for typical ICs of the model. Amplitudes presented to scale of the discretization level (not real values in [millimoles]/[meter$^{-2}$][second$^{-1}$] and [parts per million]). ......................................................... 74

Figure 4.15 Photosynthesis model. Picture taken from [89] (reduced). The impact factors of each of the components on the photosynthetic rate are represented by numbers in brackets. The most impacting factors component have been circled. The metabolites in different compartments, i.e. cytosol and stroma, are represented separately in the model. ......................................................... 76

Figure 4.16 Illustration of some of the reactions catalyzed by the enzymes identified in the literature as being highly impacting on the photosynthetic rate. ......................................................... 78

Figure A.1 Primary and secondary memberships of the components in the EGF model for $m = 5$. Columns represent components, and rows represent clusters. ................................................................. 95
Figure A.2 Primary and secondary memberships of the components in the EGF model for $m = 9$. Columns represent components, and rows represent clusters. .......................................................... 96
Figure A.3 Primary and secondary memberships of the components in the C3 photosynthesis model for $m = 3$. Columns represent components, and rows represent clusters. .......................................................... 97
Figure A.4 Primary and secondary memberships of the components in the C3 photosynthesis model for $m = 9$. Columns represent components, and rows represent clusters. .......................................................... 99
Chapter 1

Introduction

1.1 Overview

The endeavor to understand living organisms has motivated research for centuries. Living organisms, however, are tremendously complex systems and their complete comprehension is far from being fulfilled. Biological organisms are composed of multiple levels of organization, and numerous processes occur at each level. At the lowest levels of operation, millions of nucleotides combine to form genes, and in turn, they form DNA. DNA molecules, proteins, and other compounds participate in subcellular mechanisms, and contribute to cell formation. Billions of cells interact to form tissues, which are also regulated by biological processes that result in organisms. This level of complexity exceeds the comprehension ability of current research solutions. Classical methods to study biological processes have been restricted to analyze confined systems, i.e., well defined processes that occur at one particular level of biological organization. Current techniques are experimentally based, which results in very time consuming methodologies. A complete understanding of the operation of these numerous, complex systems and how they interact seems, for now, unattainable.

The difficult problems facing biological researchers, however, demand comprehension of larger and more complex processes and the study of the interaction among processes at several levels of operation. For example, the interests in fighting cancer, in identifying non-food sources of biofuels, and in improving the production of crops, require the understanding of not only a single process of the organisms involved, but also of larger views of such organisms. Large numbers of studies on particular processes have been conducted over recent decades. There is now a need for new mathematical and computational strategies to relate these studies at several levels of abstraction. Technological advances are also propelling analysis of more complex systems, since techniques such as microarrays are enabling the possibility for collecting enormous amounts of data. Mathematical and engineering techniques are needed to effectively analyze
all the collected data.

The focus of the work presented in this document is the development of a framework that combines existing and novel techniques aiming to computationally support the analysis of biological processes, and shift some of the focus from experiments to computation of current biological research methods. We do this by developing two computer-based algorithms. The first algorithm is a tool that supports the biological analysis procedure by elucidating the underlying structure and functional units of such biological processes—where functional unit refers to a group of components in a biological process that are involved in performing a common biological task. The second algorithm uses knowledge of functional units to better identify relationships between underlying biological components and higher-level biological characteristics of critical importance. By combining these two tools, the framework presented reveals new prospects towards the study of biological systems across different levels of abstraction.

1.2 Motivation

Systems biology is an interdisciplinary research area that brings together biology, math, and engineering. Biological processes are studied from a system point of view, by collecting data, building mathematical models based on this data, and making predictions or designing controls based on these models. Mathematical models of low-level biological processes from numerous organisms such as Homo sapiens, Drosophila melanogaster, or Arabidopsis thaliana, have been developed over the years, and often collected in databases [64][1][61]. The ultimate goal of systems biology, however, is typically the understanding of some characteristic of the organism at the highest level of abstraction. The motivation for studying biological processes occurring at low levels of abstraction, thus, comes from the interest in understanding phenomena at higher levels of abstraction. Numerous examples exist: the improvement of crop production and the difficulty of certain plants to adapt to sudden draught conditions caused by climate change have motivated further study of the photosynthesis pathway [89]; diseases such as cancer have driven studies of pathways that impact the cell cycle [12]; the reduction of lignin to improve efficiency of biofuel production has promoted the study of lignin biosynthesis [85]; and the improvement of the efficacy of personalized medicine has identified proteomics and the development of models in this area as a key tool to generate the necessary predictive approaches [84]. Each of these pathways and related abstract phenomena lay in different levels of biological organization. Mathematical models of biological processes generated over the years are unable to explain meaningful high-level phenomena in some cases. Thus, a better understanding of how lower level biomolecular components impact higher level phenotypic characteristics is needed. We are at a stage in the research where we can generate large complex models of interactions that do not present any intuitive understanding of phenomena that we are interested in.
Furthermore, these complex phenomena need to be described not only with confined models, but also with multiscale models. Some attempts to develop multiscale models exist in the literature, such as those in [58][75][83]. Although these models have been successful in coupling different scales for the particular problems that they address, many challenges have been identified for attaining general multilevel oriented methodologies. For example, the complexity of current mathematical models has been identified as a major obstacle to achieving multilevel modeling [21]. Advances in technology have lead to the generation of larger and more complex models. Mathematical models alone, however, do not always provide the insight needed to fundamentally understand those processes. With this, the challenge becomes interpreting models by identifying functional groups using strategic decomposition techniques. Decomposition techniques are tools that break down large complex models to facilitate interpretation and analysis. Approaches that decompose biological models by finding primary, or primary and higher order relationships among its components have been developed with this aim [47][41][38][62][72][27][3].

Among these methods, those that have considered primary and secondary relationships across components [47][41][38][62] have been applied to static models. Identifying relationships across components in dynamic models is particularly challenging due to the high dimensionality of biological models of practical interest. Thus, approaches based on dynamic models have typically been developed to find primary relationships only [72][27][3], leading to solutions that found non-overlapping functional groups. In biochemical systems, however, components can appear in multiple reactions, and may be located in multiple functional modules. Relationships developed based on mutually exclusive decomposition methods can lead to misleading results when the models that they are applied to contain a high percentage of repeated components in distinct reactions. I present in this dissertation an approach that decomposes dynamic models into functional units, identifying primary relationships, secondary relationships, and changes of these relationships over time using fuzzy clustering. This approach is intended to provide functional information needed to understand the underlying structure of models, which can be a critical tool in comprehending the impact of biomolecular components on higher-level phenomena.

We also outline in this work an approach that finds a relationship between models of low level processes and more abstract meaningful phenomena, by identifying key groups of components to control or manipulate these phenomena. This novel approach leverages machine learning tools and experimental data in combination with a sensitivity analysis to achieve this goal. Specifically, Hidden Markov Models are used as a black box modeling tool, allowing the system to pattern the relationship between both levels of abstraction by means of some experimental data. This approach is used in combination with the decomposition algorithm to identify the functional modules from complex models that can impact those abstract phenomena. This framework has the potential to be extended and applied to link models at several levels of
biological organization, highlighting path for future work in multihierarchical models.

1.3 Contributions

1.3.1 Intellectual Merit

The research presented in this dissertation is intended to serve as a critical step in the implementation of large scale Plant Systems Engineering. The framework presented encompasses two methods, that:

1. Strategically partition dynamic systems of Ordinary Differential Equations (ODEs) into hierarchical functional units.

   We develop a decomposition method that identifies primary and secondary relationships between components of biochemical pathways, as opposed to previous decomposition methods from the literature that focused on primary relationships. We are also able to assess how relationships between components or groups of components change over time.

2. Identify key biomolecular components that impact quantifiable biochemical, physiological, and behavioral traits in living organisms.

   The sensitivity analysis/black box algorithm that I present is a novel approach that, for first time, aims to relate models of biological processes with phenomena placed at different levels of abstraction in a “non model-specific” way.

3. Identify key functional units in a model that impact quantifiable biochemical, physiological, and behavioral traits in living organisms.

   The decomposition algorithm serves as a preliminary step in the process of identifying groups of biological components that impact an abstract phenomenon of interest. Finding functional groups can lead to the discovery of groups of components that can be used to control, for example, a trait of a living organism, by identifying a single component known to be impacting.

1.3.2 Broader Impacts

The impacts of my research can have indirect implications in the following way:

1. There are tremendous amounts of models in the literature. This information should serve as a base for future studies related to the same organisms, and for research that aims to impact more abstract phenomena based on existing low-level studies. There
are, however, disconnects among current models of biological processes, even when they concern the same organism. Biological models can also become very complicated. As a result of the complexity of models and the disconnect among them, the interpretation and practical use of models or groups of models can become difficult for those who did not directly participate in their generation. My decomposition method provides functional information that can be crucial for model comprehension. This tool can ease the process of studying existing biological models for their use in broader analyses and applications.

2. The paradigm that I present can contribute to save experimental time in studies where components that impact an abstract phenomenon need to be found. In the design of new drugs, for instance, knocking down a target component may not suffice to achieve a pursued effect, due to the redundancy that exists in biological systems. Experiments with multiple components may need to be performed to identify all the components that have an impact on the effect under study. This can become a very long process, with studies of a single pathway or process stretching on for years, or even decades. My algorithms could be used to identify all the components that need to be targeted without performing individual experimental studies of each of them.

3. Design and synthesis of multiscale biological systems may become a reality in the near future. Reduction algorithms have been identified as a key tool towards multihierarchical modeling. The presented methods can serve as a first step towards multihierarchical modeling, by identifying which components the reduction algorithms should target (in cases where the pursued multihierarchical model aims to describe a given trait of interest). My framework also has the potential to be extended to link models at several levels of biological organization. This research can have an impact on applications related to this perspective, such as biofuel generation, climate change, and food needs.

4. The application of both of my algorithms to a plant model is shown in this dissertation. The uses of these algorithms, however, are not limited to plant systems. My algorithms can be applied to human and general high organisms systems. The algorithms could also be extended for their use in environmental, population, and health models.

5. The historic separation of disciplines evolved such that biology is limited to almost purely empirical methodologies, reserving computational and mathematical techniques for other fields, such as physics, mathematics, or engineering. Interdisciplinary areas, such as systems biology, recognize the need for more collaborative disciplines. In past decades, biologists have had the need to understand and merge more mathematical solutions into their empirical perspective. Yet this has not become evident in our educational systems. The algorithms that I present could be used as part of a toolkit to train biologists in
a context that embraces a blend of experimental and mathematical/computational solutions. The use of these types of methods in the early stages of higher education could lead future generations of biologists to be more aware of the potential of available tools.

1.4 Organization

This dissertation focuses on the explanation of the algorithms that I developed, the context in which they are applied, and the impacts of their application. Chapter 2 presents different angles of the problems that I address. It then provides an outline of modeling biological systems, along with a sketch of the EGF induced MAPK model and the C3 photosynthesis model, which I use as my applications. This is followed by a review of methods from the literature related to each of my algorithms. This review aims to provide some technical background on each topic, as well as to highlight the deficiencies of the cited techniques in the context of my problem. Chapter 3 describes my work. It first explains the decomposition algorithm, followed by the sensitivity analysis/black-box algorithm. In Chapter 4, I apply the decomposition algorithm to the EGF Induced MAPK model, which demonstrates the usefulness of finding, not only primary memberships of components in functional units, but also secondary memberships. This application shows that secondary memberships reveal connections among components that analysis of primary memberships alone cannot uncover. I then apply both of the algorithms to the C3 photosynthesis pathway. This application shows the potential of my framework by identifying the most important components and enzymes in this pathway to increase the rate of photosynthesis. I prove that I can reach similar results to those achieved by previous experimental methods through a more timely efficient computational method. In Chapter 5 I summarize the conclusions of my work, and present possible ways of extending this research.
Chapter 2

Tools for Modeling and Analysis of Biological Systems: Background

2.1 Phenotypes Manipulation

Living organisms can be seen as highly complex systems of systems composed of multiple levels of abstraction and numerous biological processes occurring at each level. At the lowest levels, millions of nucleotides get together to form genes, and in turn, they form DNA. Other low levels of abstraction include functional proteins, which are large strings of aminoacids that serve as the catalyst for key aspects of metabolism, regulation, and signaling, and the pathway level, where enzymes, proteins, and compounds participate in reactions to execute specific metabolic, signaling and regulatory functions. Biological processes (or systems) build up, from the lowest to the highest levels of the organism, forming dependences (see Fig. 2.1). Systems biology seeks to understand these processes by developing mathematical models that describe and explain them. The ultimate goal of systems biology is typically the understanding of some characteristic of the organism at the highest level, i.e., a phenotype of the organism. For this understanding, models of the processes occurring at the gene, pathway, or cellular level are typically developed and studied. It is however very difficult to relate these lower level components to the higher level phenotype. Examples of phenotypes that led to the study of biological systems are some diseases such as cancer, that has motivated the study of pathways that impact the cell cycle [12], the improvement of crop production and the difficulty of certain plants to adapt to sudden draught conditions brought by the climate change, that raised interest in further studying the photosynthesis pathway [89], the reduction of lignin to improve efficiency of biofuel production, that has promoted the study of lignin biosynthesis [85], and the improvement of the efficacy of personalized medicine, which has identified proteomics and the development of models in this area as a key tool to generate the necessary predictive approaches [84].
The problem that we face with current models is that they often describe systems that are placed at low levels of operations (i.e., models of biomolecular interactions, genetic or proteomic interactions, etc.), while we are typically interested in knowing how these models can explain more abstract phenomena (i.e., high level characteristics of the organism, such as a cancer tumor, improvement of crop production, or adaptation to draught conditions). Meanwhile, some layers of abstraction among them exist that are not mathematically modeled. For example, the photosynthetic metabolism is located at the pathway level in Fig. 2.1, but the plant growth under draught conditions is located at the phenotype level in Fig. 2.1, while all the layers circled in the figure lie between the model and the phenotype, and they are not modeled. Effective ways of mechanistically modeling and connecting these multiple levels of biological abstraction do not currently exist.

Classical solutions to analyze a phenotype by means of a low level process are experimentally based. They consist of performing measurements to relate components of the process and the phenotype, often with the goal of constructing a mathematical model leveraging the collected data. While mathematical models between each model from the organism and the phenotype would be undoubtedly powerful, this option can be extremely expensive, time consuming, and not doable in some cases. Also, all the layers of abstraction that are not modeled between the biological process in question and the phenotype can pose a problem for developing a mechanistic model that connects them, since there is a disconnect between them. A particular example
is the study of the photosynthesis pathway and the efforts to understand how this pathway can improve the photosynthetic rate. Photosynthesis is a biological process that impacts the environment and humans in multiple ways. This has resulted in a tremendous interest in studying photosynthesis over the years. The capacity of increasing the photosynthetic rate of some plants, for example, could result in an increase of crop production [56]. Improving the efficiency of plants to perform photosynthesis given a limited amount of resources could help them to better resist the effects of climate change [89][55]. These are some examples of challenges that have lead scientists to try to identify enzymes that play a critical role in photosynthetic activity. Studies such as that of Tamoi et al. [82], generate transgenic plants to clarify the contribution of certain enzymes—FBPase and/or SBPase, in this case—to photosynthesis. These experiments target the enzymes individually, and thus, a different transgenic plant needs to be generated for the study of each enzyme. These studies, would benefit from computational methods that could accelerate the process.

Another example is the relationship between the PI3K signaling pathway and cancer. It is known that inappropriate co-option of PI3K signaling is one of the most frequent occurrences in human cancer [19]. This has motivated multiple studies over the years to generate inhibitors of the PI3K pathway to treat cancers. It is not yet clear however, which types of cancers will benefit most from these treatments [19]. Several therapeutics target different components to test their efficacy in distinct cancer types in preclinical studies. This type of investigation—that need to target, experimentally, one component at a time—would also benefit from computational methods to ease and accelerate the process, or from alternative solutions that involved less experimental information.\(^1\)

An alternative to these costly mathematical models, that require large amounts of experiments, would be the generation of non mechanistic models that have lower needs for resources. A non mechanistic model would not provide a description of the abstract phenomena in terms of the components of the low level biological system. It would, however, provide a viable way of predicting changes in phenotype when modifications are made at lower levels of biological abstraction.

In this work, I have developed a framework that addresses this problem. My algorithms generate non mechanistic models to connect low level biological processes and phenotypes. We use these non mechanistic models to identify key components in low level biological processes that impact a phenotype of interest. To my knowledge, this problem has not been addressed in the past.

To solve this problem, I need to:

1. Divide the pathway into groups of components that participate collectively in the estab-

\(^1\)Although I do not work on the application of the PI3K signaling pathway and cancer, this example is provided through all this document to illustrate a different type of application that my method can target.
lishment of any phenotype. Components from the same group, should accordingly be jointly needed or dispensable for the manipulation of a given phenotype. This division of the pathway corresponds to a partition into groups of components that are functionally related, since components that share a common function in the pathway should have a common role in the establishment of any phenotype.

2. Identify which of the previously determined groups of components influence in a greater or lesser factor a given phenotype. This identification corresponds to a sensitivity analysis (SA) of the phenotype in relation to the components of the model. SA methods rely on a mathematical description of the output of interest (phenotype) in terms of the inputs (model components) to perform the analysis. The model and the phenotype however, lay in different levels of abstraction and their relationship is not assumed to be known. I thus, need to perform two tasks at the same time, modeling the relationship between the system and the phenotype, and performing a SA based on this relation. Establishing a black box relationship between components and phenotype then performing SA would be well suited for this task.

The framework that I present encompasses two computational strategies that address the two tasks that need to be accomplished. The first one is a model decomposition technique that hierarchically divides a dynamic biological model into functional units of the system, revealing the underlying structure of the system. It is accepted that biological systems have a modular
structure [27][3]. I claim that the identification of these modules uncovers information about which groups of components impact higher level phenotypes. Components belonging to the same functional group in a biological system share a common function in the overall process, for we define functional units as groups of components that share a common physiological task. I hypothesize that they have similar impact factors on a phenotype of interest, i.e., certain functional modules, as a whole, will be more necessary to control certain phenotypes than some others, and more dispensable than others to control a different phenotype. Finding functional groups thus, can lead us to find groups of components that can be used to control a phenotype, by identifying a single component known to be impacting. Components in biological systems can interact with multiple reactions simultaneously, and consequently, participate in several functional groups. My algorithm aims to not only divide the model in functional units, but also to find the components that are related to diverse functional groups. A decomposition algorithm without this property would be able to place each component in only one functional group, leading to a potential failure in identifying all the components that impact a phenotype of interest. I generate my decomposition algorithm by finding relationships among components. Unlike previous decomposition algorithms, I do not only find primary relationships among components, which would lead to find non overlapping functional groups, resulting in the location of each component in a unique group. Instead, I find a hierarchy of relationships among components, focusing mainly on primary and secondary relationships.

The second algorithm of the framework is the identification of components of a model that can be used to control or manipulate a higher level phenotype. This algorithm links the disconnect between the model and the phenotype by generating a black box model of their relationship (See Fig. 2.2). This model is used to identify the variation of the phenotype due to a variation of any component in the model, trough a SA algorithm. This leads to the determination of which are the components that impact the phenotype in a greater or lesser factor. This novel algorithm can contribute to save experimental time in studies where components that impact an abstract phenomena need to be found. In the design of new drugs, for instance, knocking down a target component may not suffice to achieve a pursued effect, due to the redundancy that exist in biological systems. Then, experiments with multiple components may need to be performed, which can result into a very long process. These experiments pursue to identify all the components that have an effect on the expected outcome –phenotype of interest. My algorithm could be used to identify all the components that need to be targeted without performing experimental studies of all of them. My work overcomes the challenges of modeling distinct layers of multihierarchical organisms by introducing black box models in the modeling perspective, in combination with the classical mathematical descriptions.
2.2 Towards Multihierarchical Modeling

We argue that models have traditionally not been developed multihierarchically, but they have rather been constrained to describe one or more processes occurring at the same biological level, such as the cellular or the pathway level. I am now thus, facing the challenge of interpreting the models that have been generated during decades, and trying to link processes and models from multiple layers.

The scientific community has recently started to highlight the importance of moving towards finding multihierarchical solutions, for this is the only path to a more complete understanding of organisms. Maus et al. [58] identifies multi-levelness to be an important and general principle of systems biology that needs to be addressed in the models. Dada & Mendes [21] review the state of the art of multiscale modeling from several disciplines, and analyze the potential of these approaches in biological models. They identify model reduction as a strategy needed to simplify large complex models, which arise from multi-scale approaches. They review all the attempts to develop a multiscale model that can be found in the literature. The distinct models have been developed using different techniques, such as Agent-Based Modeling or finite element modeling. Some models present as major drawbacks, high complexity and inability to have mathematical descriptions that can be simulated at once. They argue, in agreement with [5], that “there is a need for automated methods to help in analysing a complex model defined at a particular spatial and temporal scale in order to compute the parameters of a simpler model that captures the model behaviour relevant to the scales above”. This crucial statement identifies these type of techniques as the bridge between current modeling techniques and a systematic multiscale modeling.

We claim that my framework can be an important tool to link several models from distinct levels of abstraction. I do not propose a reduction method, but instead, my method analyzes and identifies which modules of functional units and groups of components need to be discarded or highlighted by reduction techniques. As stated by Dada & Mendes in [21], “it is only by combining multi-scale modeling strategies with advances in computational technology that the aims of systems biology can fully be achieved”, and my proposed computational framework is a big step in this direction.

The rest of this chapter is divided in sections according to the different algorithms of my work. It first presents some characteristics of biological models, and this is followed by an introduction to the two models that I have used in this work to test my algorithms. We then discuss the format of the experimental data that can be commonly found in the literature, proceeded by the state of the art of decomposition methods, the background related to black-box models, and to SA approaches.
2.3 Modeling Biological Systems

Modeling biological systems consists of describing a biological process (system) under consideration. This is typically done mechanistically, through mathematics. Modeling natural systems is a necessary step to understand the processes and phenomena around us. It is a step to go from observations of biological processes, to a level of abstraction where we can operate quantitatively. Formulating mathematical models of biological systems, allows us to study and understand the models, and potentially, to predict and control them.

Classical methodologies of biological modeling include S-systems, Michaelis-Menten kinetics, or Mass Action representations. For an enzymatic reaction as follows,

\[
E + S \rightleftharpoons ES \rightarrow E + P
\]

where E is the enzyme, S is the "substrate", ES is an enzyme-substrate complex, and P is the product, Mass Action models predict the dynamics of \(E\), \(S\), \(ES\), and \(P\) in terms of the intermediate \([ES]\) and the rates \(k_i\). Michaelis-Menten kinetics predicts the dynamics of the product, \(\frac{d[P]}{dt}\), in terms of \([S]\) and the rates \(k_i\), while S-system models are approximations of the Michaelis-Menten kinetic rate laws with power-law functions.

The common factor among the shown modeling representations is that they predict several metabolites dynamics in the system through Ordinary Differential Equations (ODEs). The biological processes presented in this document are modeled with ODEs derived from one or more types of representations.

2.3.1 EGF Induced MAPK

The EGF receptor-induced MAP kinase cascade has been widely studied, and has been used to test several system decomposition or reduction algorithms [72][27][3][18]. I use this model to test my decomposition algorithm, for it is possible to validate my results with previous decomposition results.

The Epidermal Growth Factor (EGF) receptor is a mammalian hormone involved in embryonic and postnatal development, and in progression of tumors [74]. Schoeberl et al. developed a popular model of the EGF receptor-induced MAPK, presented in [74] and available in the BioModels database [50], with a total of 100 variables and 125 reaction rates. Mechanistically, this model describes the reactions from the binding of the EGF molecules to their receptors, to the phosphorylation of MEK, and the catalyzation of the dual phosphorylation of ERK. It includes the binding of the EGF, the formation of signaling complexes, the activation of Ras,
and the activation of the MAP-kinase cascade. This system also models the EGF receptor internalization.

2.3.2 C3 Photosynthesis Model

The C3 photosynthesis is the most widely studied process in plant biology. Knowledge about this process allows us to optimize resources to increase photosynthesis, and thus impact food and biofuel challenges. This idea has motivated a large number of previous studies. Models of the C3 photosynthesis can be found in [89][45][46][67][91], among others. Studies of how some components from the model affect the photosynthesis have also been performed [89][82][80][67].
The amount of information available about this model makes it a great candidate to test my computational strategies, for my results can be validated by comparison with previous analysis.

The model developed by Laisk et al. in [45] and [46], and available in the BioModels database [50] (reduced version), has been used as my testing model. I particularly worked with the version in BioModels—in the reduced version, the light dependent reactions are not implemented. It consists of 42 state variables, each representing the concentration of a component in the system. The model includes the Calvin Cycle, as well as the Starch synthesis, and Sucrose synthesis. My algorithm decomposes this model into functional units, and it identifies which are the most important components in the model to manipulate or control the photosynthesis. This identification of components shows, in turn, that the decomposition in functional units can also help to elucidate which are the most important components—by association with the most impacting functional units.

Figure 2.4: C3 Photosynthesis model from by Laisk et al. in [45]. The original picture from [45] has been modified, to include only the dark reactions.

Results from the studies in [89] [82] have been used in my work to validate my own conclusions. Zhu et al. [89] analyses the distribution of resources between enzymes of photosynthetic carbon metabolism, concluding which enzymes have a greater factor of impact on the photosynthesis. Tamoi et al. [82] narrows the analysis down to the impact of two important enzymes. These two analysis are in agreement with other current literature [67], and I will show that my
results are also in line with them.

### 2.3.3 Experimental Data

When a phenotype of an organism is investigated, a typical procedure for biologists starts with collecting measurements from the phenotype in relation to some important component of the organism, or to some drug. The measurements are often collected as time series, and may be repeated for various conditions. For example, if the growth of a plant needs to be studied, measurements of the plant size with respect to some internal component levels—for example, Sodium chloride (NaCl), Sodium ion (Na\(^+\)), or Carbon dioxide (CO\(_2\))—may be taken [31]. Other examples could be the study of the growth of a tumor, where measurements of the size of the tumor, and measurements of some related gene or some applied drug are taken [39]. This type of measurements are typically taken as an initial pool of information to perform preliminary analysis in the study of a phenotype, or for model validation.

### 2.4 Modularization of Biological Systems

Model decomposition has been addressed from different perspectives in the literature. It has been used to analyze signaling networks [72], gene co-expression networks [37][47], social networks, and computer networks [63]. Approaches to implement module identification can be separated into two main groups. The first one performs module decomposition of static models, i.e., the available information is static data. The second approach deals with module decomposition of dynamic systems.

#### 2.4.1 Modules in Static Data

Several approaches can perform identification of modules in static data. These approaches include machine learning (clustering) and correlation networks.

**Clustering Techniques**

Machine learning is concerned with extracting information from data, and developing algorithms that can automatically extract categorical patterns from data. Machine learning methods can be supervised or unsupervised. Supervised learning refers to those techniques that use a data set with a known pattern to adjust the classification system. In contrast, in unsupervised classification or clustering, “similar objects” are grouped into clusters. In this case, the classes or groups are unknown.

Figure 2.5 illustrates the steps in the process of clustering [86]. These steps are:
1. **Feature Selection:** Some problems are greatly simplified by reducing their dimensionality before starting the classification. Feature selection identifies which are the most important features or inputs of a problem. Feature reduction can be implemented by discarding the least important features.

2. **Similarity Measures and Clustering Algorithms:** Defining “similarities between objects” is a key step of the process of clustering. This step is known as the selection of similarity measures. This is followed by “selecting or designing the clustering algorithm to be applied”.

3. **Cluster Validation:** In this step, a testing criteria will be applied to measure the validity of the obtained clusters.

![Clustering process diagram](image)

**Figure 2.5:** Clustering process

Numerous studies have developed different clustering algorithms. They can be hierarchical, or nonhierarchical, depending on whether the resulting groups are organized in a hierarchical structure, or they divide the domain in a one layer partition. Linkage methods are an example of hierarchical algorithms, whereas K-Means, and Self-Organizing Maps are examples of non-hierarchical algorithms (note that Self-Organizing Maps are ANN trained with unsupervised techniques). They can also be classified as crisp, if the objects belong to one and only one group, or fuzzy, if the objects can be classified in more than one group with certain degree.
Fuzzy algorithms will be particularly useful when the boundaries between groups are not clearly defined. An example of this could be the classification of components that are involved in several biochemical pathways at the same time. One particular fuzzy algorithm of interest is Fuzzy C-Means (FCM) [26][7], which is probably the most popular fuzzy clustering algorithm [36]. Details of FCM follow.

- **Fuzzy C-Means:**

  Given a number $N$ of data points that I wish to group in $C$ different clusters based on their similarity or proximity, FCM aims to minimize the distances between the center of each cluster and the cluster member. FCM, however, allows each data point to be located in more than one cluster, with different degrees of membership.

  Let us define the membership of the data point $i$, $x_i$, in cluster $j$ with cluster center $c_j$, as:

  \[
  u_{ij} = \frac{1}{\sum_{k=1}^{C} \left( \frac{||x_i - c_j||}{||x_i - c_k||} \right)^{2-m}}^{m-1}, \quad 1 \leq m \leq \infty,
  \]

  where $m$ is the fuzzifier coefficient, which determines the level of fuzziness of the algorithm. The fuzziness of the clustering increases with increasing values of $m$.

  Let us also define the center clusters, $c_j$, as:

  \[
  c_j = \frac{\sum_{i=1}^{N} u_{ij}^m \cdot x_i}{\sum_{i=1}^{N} u_{ij}^m}, \quad 1 \leq m \leq \infty,
  \]

  Given these definitions, FCM aims to minimize the following objective function:

  \[
  J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m ||x_i - c_j||^2, \quad 1 \leq m \leq \infty
  \]

  FCM iteratively optimizes the objective function, by updating, in each iteration, the membership and cluster center values.

**Correlation Networks**

A correlation network is a system of interrelated nodes and connections, built of correlations between quantitative measurements. Correlation networks are increasingly used in bioinformatics and biological applications [37][47]. Network analysis has been successful in gene co-expression networks, protein-protein interaction networks, cell-cell interaction networks, and the world wide web and social interaction networks [88]. Correlation networks can be used,
among others, to find clusters of interconnected nodes, and to build modules based on that. Gene co-expression networks are probably the largest application of correlation network analysis [47]. In a gene co-expression network, genes are represented by nodes. There is a connection from node to node if the corresponding genes have highly correlated gene expression profiles. Clustering algorithms typically leverage gene co-expression networks methods, and are used to group highly correlated genes. Correlation networks however, are typically based on static datasets that do not fully reflect the complete perturbation space of a given biological system. Other model decomposition methods that have been developed to modularize signaling or metabolic pathways considering dynamic models are presented below.

2.4.2 Model Decomposition of Dynamic Systems

Model decomposition aims to divide large models into smaller interacting subunits. Studies to analyze the modularity of dynamic systems, and in particular biological systems, can yield a more complete understanding of the system. These include clustering techniques for dynamic data, community detection algorithms, spectral graph theory, and simulation based approaches.

Clustering Techniques

Clustering techniques have been developed, not just for static data, but also for time series of data [71] [32]. T. W. Liao provides in [52] a survey of different clustering methods that aim to work with time varying data, rather than static data. The methods that are presented in this survey are similar to methods for static data. The similarity measure that they use, and/or the preprocessing of the data is what separates them from the clustering methods for static data. Some methods [71] work directly with raw data, and thus, they have to use appropriate measures of similarity that reflects the character of the time series. Some others [32] work indirectly with features extracted from the raw data, or indirectly with models built from the raw data. The interested reader can refer to [52] for more information.

Community Detection Algorithms And Spectral Graph Theory Based Algorithms

Community detection algorithms such as that presented in [63] base their approach on graph theory. These methods have become popular, but they ignore the dynamics of the system. Saez-Rodriguez et al. present in [72] a method based on community structure detection that considers the dynamics to make the partitions. This method defines modules by minimizing the retroactive interconnections. Components in the biochemical network that are not coupled bidirectionally to any other element have no entries in the adjacency matrix. Therefore, these components can be placed in any module. An optimization method is used afterwards to optimize the modularity of the system.
Anderson et al. [3] develop a method based on spectral graph theory. The method is suitable for nonlinear systems and requires no prior information about the system, although it requires a linearization of the system to perform the decomposition. These two methods achieve hard classification of the systems, but their approaches do not address the identification of meaningful secondary relationships among components.

**Heuristic-Simulation Based Algorithms**

The methods presented in [27][44][87] are heuristic-simulation based algorithms. They find relationships among the components of a model leveraging data extracted from simulations of the model. As explained in [27], the exact analysis of large nonlinear ODE models is very difficult and in most cases impossible, which leads to the use of heuristic methods. The method described in [87], however, finds relationships among proteins from mRNA and protein time series, rather than being applied to signaling or metabolic pathways. The method summarized in [44] and the one developed by Ederer et al. [27] perform the decomposition based on the correlation or distance between the activities of substances, i.e., two substances are more likely to belong to the same functional unit if they are often active at the same time. Particularly, the comparison of the activity of the components is performed along typical trajectories of the system. The idea used in [27] is similar to reconstructability analysis [90] from general systems.
Reconstructability analysis, however, is computationally too expensive, and it is not applicable to large problems—and hence to many biological systems of practical importance.

Heuristic-simulation based algorithms have the potential to identify secondary relationships, and even the transients of these relationships, since the entire approach focuses on the dynamics of the components. The method in [27][44], however, as the previous ones, does not address this problem. Particularly, 1) the structure of the algorithm does not allow the capture of transient relationships, or the variation of the decomposition in time, 2) the algorithm, in the form presented, does not capture the secondary relationships of the components for a given time, and 3) the method is difficult to generalize, since it requires some prior knowledge of the system regarding the nature of some of the components. In my work, I leverage heuristic-simulation approaches to develop my decomposition algorithm, for heuristic-simulation methodologies allowed us to find secondary relationships among components. My algorithm decomposes biochemical pathways into functional units, identifying primary relationships, secondary relationships, and changes of these relationships over time.

2.5 Black-Box Modeling

Black-box models are mathematical constructs that do not require physical or mechanistic insight [77]. They approximate a model from a collection of data, rather than from a mathematical description. This type of modeling is concerned with relating observed inputs and outputs of a dynamical system, such that the inferred function that relates them is parametrized in a flexible way [54]. Techniques to perform nonlinear black-box modeling include regression based methods [28], function expansions and basis functions [77], and wavelets networks [51]. They also include learning machine methods, such as Hidden Markov Models (HMM) [6], Artificial Neural Networks (ANN) [40], and support vector machines (SVM) [81].

Among all the cited approaches, ANNs are one of the most popular ones, due to the fact that they have been very effective in a large number of nonlinear estimation problems [77]. SVMs have more recently received a lot of attention. The main reason for this is that SVMs typically outperform ANNs in terms of classification accuracy [4][9]. SVMs, however, involve a higher computational complexity than ANNs [14]. The high complexity required to train and execute SVMs inhibits them from being applied to large problems [4]. HMM have also been reported to outperform ANN in a number of studies [17][35]. Conroy et al. [17] use HMM to classify three well-known chromosome data sets and compare these approaches with the use of ANN. They show that HMMs are a particularly robust approach to identification, representing a substantial improvement of the classification accuracy chromosomes compared with ANN-based methods. Hatzipantelis et al. [35] also show that HMMs produced the best recognition
performance compared to ANNs, although their application is not biological.

HMMs are specially useful for temporal applications. They allow us to recover a sequence of states of a system from a sequence of the system’s output. The system being modeled in an HMM is assumed to be a Markov process, where the states are hidden. On the contrary, the output is observable, and dependent on the states. The transition from one state to the next is a Markov process and each state has a probability distribution over the possible output. Knowing the output sequence then, gives information about the states that lead to such output. As opposed to other machine learning tools thus, HMMs do not require multiple input feature vectors to improve their performance. It suffices one single feature or observation time vector to predict the output time vector. This is advantageous for biological applications, since minimizing the experimental data needed to make a prediction is desired. Furthermore, my target models produce temporal data since they are dynamic. This makes HMMs, along with the aforementioned advantage, the most appropriate tool for my algorithm. Further details on HMMs and a short example are provided in the following subsection to offer a conceptual understanding of this approach.

2.5.1 Hidden Markov Models

Markov Models

A Markov model is a stochastic process, \( x(t) \), that satisfies the following property:

\[
P(x(t_n) \leq x_n \mid x(t) \forall t \leq t_{n-1}) = P(x(t_n) \leq x_n \mid x(t_{n-1}))
\]

which is the first order Markov property, i.e., the current state of the process depends only on the previous state.

Mathematical Description of Hidden Markov Models

An HMM is a Markov model in which the states are hidden. The following are the elements that characterize an HMM [79][68]:

- \( T \): length of the observation sequence
- \( N \): number of distinct states in the model
- \( M \): number of observation symbols
- \( Q = q_0, q_1, ..., q_{T-1} \): actual state at each time \( t \), for \( t \in [0...T-1] \)
- \( S = \{s_0, s_1, ..., s_{N-1}\} \): distinct states symbols of the Markov process
• $V = \{v_0, v_1, ..., v_{M-1}\}$: set of observation symbols

• $A$: state transition probability matrix

• $B$: observation probability matrix

• $\pi$: initial state distribution vector

• $O = (O_0, O_1, ..., O_{T-1})$: observation sequence

An HMM is typically denoted by $\lambda = (A, B, \pi)$, where

$$A_{N \times N} = \{a_{ij}\}, \quad a_{ij} = P[q_t = s_j \mid q_{t-1} = s_i] \quad (2.6)$$

$$B_{N \times M} = \{b_{ik}\}, \quad b_{ik} = b_i(v_k) = P[O_t = v_k \mid q_t = s_i] \quad (2.7)$$

$$\pi_{1 \times N} = \{\pi_i\}, \quad \pi_i = P[q_1 = s_i] \quad (2.8)$$

and $A$, $B$, and $\pi$ are row stochastic. $A$ denotes the probability of transitioning from each state to any other state, $B$ represents the probability of observing a symbol given a particular state, and $\pi$ is the distribution of probabilities of which state was the initial state.

As follow, I give an example, presented by Stamp [79], that illustrates the concept of an HMM.

**Example [79]**

Assume we want to determine which was the average annual temperature during a series of years, in some region of the Earth, in a distant past. We are only interested in a binary discretization of the temperature in *cold* or *hot*. Since we do not have access to any recorded data of the temperature, we want to determine the temperature by looking at an indirect index that can be collected in the current time. The index resolves that there is a correlation between tree size—small, medium, large—and temperature—cold, hot—, given by:

$$\begin{pmatrix} 0.1 & 0.4 & 0.5 \\ 0.7 & 0.2 & 0.1 \end{pmatrix} \quad (2.9)$$

and we have also evidence that the probability of a hot year followed by another hot year is 0.7 and the probability that a cold year is followed by another cold year is 0.6, which can be summarized as:

$$\begin{pmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{pmatrix} \quad (2.10)$$
For this example, the temperature $H$ or $C$ is the state, matrix (2.9) is the observation probability matrix, and matrix (2.10) is the state transition probability matrix. The transition from one state to the next is a Markov process of order one, and the states of the system are hidden, since we cannot observe the temperature directly. Instead, we observe the trees sizes—observations—from which we infer the states of the system. In addition, we could have an initial state distribution, indicating the probability that the first year of the series was hot or cold:

$$
\pi = \begin{bmatrix} 0.6 & 0.4 \end{bmatrix}
$$

(2.11)

This model corresponds to a Hidden Markov Model.

### 2.6 Sensitivity Analysis

Sensitivity analysis (SA) aims to quantify the impact of each input or input parameter on the output of a system. This is measured by computing the change in the output due to a perturbation of each of the inputs or input parameters. SA helps elucidate the dependences of the output with respect to the inputs. This makes SA an important tool in several steps of the mathematical modeling process—some examples are its use in testing the robustness of the system, in reducing uncertainty, in model verification, or in model reduction [13]. Sensitivity analysis is a useful tool in mathematical modeling of a variety of areas, such as economics [65], physics [48], biology [57] or engineering [11]. Of particular interest to us, is the application of sensitivity analysis to the process of modeling biochemical systems [57][16][25].

#### 2.6.1 Sensitivity Analysis Methods

SA methods can be classified in local, screening, and global methods, as defined in [23][13]. Local methods perform the analysis around a typical point in the input parameter space. An example of a local method is the one-at-a-time (OAT) approach [20], which is probably the most widely used SA approach [23]. Screening methods are applied to complex systems where the number of parameters would result in a computationally expensive analysis. They narrow down the number of parameters by discarding the least impacting ones. After this, a more computationally expensive method can be applied. The Morris method is an example of a screening method [60]. Global methods do not limit the analysis to the region around a single point, and instead, they perform the perturbations in the entire input parameter space. These algorithms are the most computationally expensive, and also the ones that provide a most complete result. The Fourier Amplitude Sensitivity Test (FAST algorithm) [10] and the Sobol’s method [78] are examples of global methods.
Given an initial system of ODEs,

$$\dot{x} = f(x(t), p), \quad x(0) = x_0$$

(2.12)

$$y(t) = g(x(t)).$$

our problem consists of finding the largest impacting inputs $x_i$ on a given output of interest $y$. OAT methods represent the prevalent approach when analyzing biochemical systems modeled with ODEs [23]. For the problem just presented, an OAT method would perturb each input $x_i$ and measure the changes in the output $y$. The quantification of this change is typically named sensitivity coefficient, and can be defined as

$$S_i = \frac{g(x_1, ..., x_{i-1}, x_i + p\% \cdot x_i, x_{i+1}, ..., x_N) - g(x_1, ..., x_{i-1}, x_i, x_{i+1}, ..., x_N)}{p\% \cdot x_i}$$

(2.13)

or as the derivative

$$S_i = \frac{\partial g(x)}{\partial x_i}$$

(2.14)

OAT methods are typically applied locally, i.e., $f$ and $g$ are calculated at a nominal operation point $\tilde{x}$. In this case, these methods are called local one-at-a-time (LOAT). LOAT methods are computationally inexpensive, and provide knowledge about the dependences of the output with one input at a time around an operational point of interest. This information often provides the knowledge needed to apply model complex computationally intensive procedures. When knowledge about interactions among inputs, or about the simultaneous effect of several inputs on the output is desired, LOAT methods alone will not suffice.

Metabolic Control Analysis (MCA) [42] is a similar approach to that of LOAT methods, that has been extensively used in models of metabolic pathways [80][67]. In this case, the sensitivity coefficient, known as flux control coefficient, is defined as:

$$C_{iJ} = \frac{\partial J_a}{\partial E_i} E_i$$

(2.15)

and it measures the impact of an enzyme $E_i$ on a flux of interest $J_a$. Although purely, this is not a general SA method, it is an approach that has been developed for metabolic networks, and performs a similar analysis to that of the LOAT methods. This highlights again that in metabolic pathways, one-at-a-time sensitivity analysis techniques are useful for assessing control characteristics in metabolic pathways.

The methods introduced here —both, SA and MCA— rely on a mathematical description of the output of interest in terms of the inputs to perform the analysis, i.e., they need a function that relates $y$ and $x_i$, or $J_a$ and $E_i$ to compute the coefficients. In my case, as it has already been stated, the output of interest is a phenotype that lies in a higher level of abstraction.
than that of the model and the inputs. The relationship between them, not known a priori, is modeled in my framework using a black box. My SA thus, will incorporate a black box model to find the perturbation of the phenotype due to a perturbation of each of the components.

2.7 Chapter Summary

Biological organisms are composed of numerous processes that are placed at different levels of abstraction. Systems biology seeks to understand these processes by developing mathematical models that describe them. The ultimate goal of systems biology is typically the understanding of some characteristic of the organism at the highest level, i.e., a phenotype of the organism. The problem that we face with current models is that they often describe systems that are placed at low level models, while they intend to explain a high level characteristic of the organism. Meanwhile, some layers of abstraction among them exist that are not mathematically modeled. Effective ways of mechanistically modeling and connecting these multiple levels of biological abstraction does not currently exist.

Classical solutions to analyze a phenotype by means of a low level process are experimentally based, often with the goal of constructing a mathematical model leveraging the experiments. These solutions however, are typically expensive, and not doable in some cases.

We briefly outlined in this chapter an alternative to these costly mathematical models. This alternate solution is based on the generation of non mechanistic models that have lower needs for resources and experiments. Particularly, this solution divides a low level model in functional groups, and then identifies which functional groups impact the phenotype of interest. This is done by means of a black box that models the relationship between the model and the phenotype, and a SA method. This solution constitutes my work.

The outline of the method in this chapter was followed by a review of the background related to each step of the method. Several techniques, along with their issues associated to address the problem that I seek to solve, were presented.

In the following chapters, I expand my discussion on these topics. In chapter 3, I present my work on decomposition and analysis algorithms. In chapter 4, I apply the algorithms to the EGF Induced MAPK model, and the C3 Photosynthesis pathway. In chapter 5, I summarize the conclusions of my work, and present possible ways of extending my work.
Chapter 3

Computational Strategies for the Transition towards Multiscale Modeling

3.1 A Computational Framework for Model Decomposition

3.1.1 Introduction

The decomposition of large complex models in functional subunits is critical for understanding the relationships between components and their contribution to emergent characteristics of the system. Attaining this level of decomposition or modularization is fundamental for understanding large complex models of biological systems. Understanding the functions that relate components in biological pathways allows us to better assess combinatorial effects to new medicines, identify avenues to improve responses in plants to environmental stress, and generate useful modifications that can improve nutrition content in agricultural foods. Mathematical models provide useful tools for describing biological components and their interactions in significant detail. This level of detail, along with the increase in model complexity, however, has lead to models that are often difficult to fundamentally understand. With this, the challenge now turns to interpreting models by identifying functional groups using strategic decomposition techniques. Decomposition techniques are tools that break down large complex models, and facilitate its interpretation and analysis. Approaches that decompose biological models by finding primary, or primary and higher order relationships among its components have been developed with this aim [47][41][38][62][72][27][3]. Particularly, understanding the primary and secondary relationships among biological components can provide significant insight into practical applications for identification of cis-regulatory elements [70], the study of secondary metabolites [29],
the detection of coregulatory relationships among genes [47][41][38][62], and can serve as the precursor for model reduction algorithms [18]. In this paper, the terms primary and secondary relationships will be confined to define “links among components with the same primary function”, and “links among components that do not have the same primary function, but still exhibit influence on the activity of one another”, respectively.

Decomposition techniques that use static datasets, e.g. microarrays, have been developed to identify primary and secondary relationships across components [47][41][38][62]. Identifying these relationships across components in dynamic models, however, is particularly challenging due to the high dimensionality of biological models of practical interest. Approaches based on dynamic models have typically been developed to find primary relationships only. Community structure detection algorithms [72], heuristic methods based on typical trajectories [27], or methods based on systems control theory [3] are some examples. All of them perform functional decompositions of dynamic biological processes by identifying primary correlations or relationships between groups of components. These relationships are identified based on mutually exclusive functional modules. In biochemical systems, however, components can appear in multiple reactions, and thus, may be located in multiple functional modules [33]. Relationships developed based on mutually exclusive decomposition methods then, can lead to misleading results when they are applied to models that contain a high percentage of repeated components in distinct reactions. Furthermore, these decomposition methods may result in modules that lead to hypotheses that contradict known biological functions [27][18].

In this section, I present an approach for decomposing dynamic models into functional units, identifying primary relationships, secondary relationships, and changes of these relationships over time, using fuzzy clustering. I apply my algorithm to two models of biochemical pathways: 1) the EGF induced MAPK [74] pathway, and 2) the C3 photosynthesis model [45][46].

The contributions of this work can be summarized in the following points:

1. I develop a method that identifies primary and secondary relationships between components of biochemical pathways, as opposed to previous decomposition methods from the literature [72][27][3] that focused on primary relationships.

2. I are able to assess how relationships between components or groups of components change over time. Biological systems are dynamic systems, and as such, the strength of relationships between components likely changes over time. Because of this, static connections or relationships often generated by more traditional approaches may not be the most appropriate to fully elucidate relationships between biological components. The identification of relationships that change over time, and the contribution of the secondary relationships to the final decomposition allowed us to reveal relationships among components that previous techniques did not find.

The biochemical models that I address in this algorithm are described by systems of ODEs
of the form:
\[ \dot{x} = f(x, p), \quad x \in \mathbb{R}^n, \quad p \in \mathbb{R}^q \] (3.1)
where the elements of \( x, x_k \), represent the biological components involved in the pathway, and
the vector \( p \) represents the \( q \) kinetic parameters of the pathway. The vector function \( f(x) \in \mathbb{R}^n \)
characterizes the change in component concentration over time. I define \( \xi(t, x(0)) \in \mathbb{R}^n \) as the
solution of (3.1) for some time \( t \), and with the initial condition (IC) \( x(t = 0) = x(0) \in \mathbb{R}^n \).
\( \xi(t, x(0)) \), also called the trajectories of the system, describe the variation in time of each of
the component concentrations.

The focus of my work is to develop an algorithm capable of revealing hierarchical relationships between
the components of any biochemical system, given relations embedded in trajectories \( \xi(t, x(0)) \). Particularly, I extract characteristics from a collection of \( M \) trajectories \( \xi(t, x(0)_{(h)}) \),
corresponding to \( M \) different ICs \( x(0)_{(h)}, h \in [1, M] \). The algorithm uses these trajectories
to decompose the model in functional modules \( C_j(t) \). These functional modules capture both
intra-module primary relationships and cross module secondary relationships between components. I describe the individual components of the algorithm in the sections below.

3.1.2 NOTATION

The following notation will be used in the next sections:

- \( n \) is the number of variables.
- \( M \) is the number of trajectories.
- \( T \) is the number of time points.
- \( \zeta \) is the number of clusters in a clustering.

3.1.3 General Steps of a Heuristic Simulation Based Algorithm for Model Decomposition

A heuristic simulation based algorithm for model decomposition, such as that presented by
Ederer \( et al. \) in [27], is composed of the following steps:

Generate typical trajectories

Generate \( M \) trajectories of the model that cover a representative area of the state space, starting
the model at \( M \) different ICs. The trajectories are discrete in time, and of length \( T \).
Normalize the data

Normalize the values of the trajectories to values between 0 and 1, and the derivatives between -1 to 1:

\[ f_k^* = \frac{f_k}{f_m^k}, \quad x_k^* = \frac{x_k}{x_m^k}, \quad \forall k \in [1, n], \quad (3.2) \]

where:

\[ f_m^k = \max_{h \in [1, M], 0 \leq t \leq T-1} |f_i(\xi^{(h)}(t))|, \quad (3.3) \]

\[ x_m^k = \max_{h \in [1, M], 0 \leq t \leq T-1} \xi^{(h)}(t). \quad (3.4) \]

Discretize the data in a finite number of levels

This is a critical step for the method presented in [27]. Ederer et al. compute what they define as “activities”, and classify each component as active or inactive at each sampling time. The definition of activity in Ederer et al. depends on the nature of the component. For instance: 1) a fast component will be active if its concentration is a large value, whereas 2) a slow component will be active if its derivative is a large value. For these two examples thus, the activity at time \( t \) is defined as:

\[ a_f^k(t) = \begin{cases} 
1 & \text{if } x_k^*(t) > c \\
0 & \text{otherwise} 
\end{cases} \quad (3.5) \]

\[ a_s^k(t) = \begin{cases} 
1 & \text{if } f_k^*(t) > c \\
0 & \text{otherwise} 
\end{cases} \quad (3.6) \]

where \( c \) is a small positive number. Then the activity vector \( a_k \) of the \( k \)th component is a vector containing all of the activities for each simulation, and it is defined as:

\[ a_k = [a_k^{(1)}(0), \ldots, a_k^{(1)}(T-1), \ldots, a_k^{(M)}(0), \ldots, a_k^{(M)}(T-1)] \quad (3.7) \]

Perform cluster analysis

Clustering of the components is performed, where the activities defined in (3.7) are used as the feature vectors for each component. Some possible measures of distance are the crosscorrelation among components, or the L-1 distance among component activities.

Generate functional units

Decompose the components of the model into functional units based on the result given by the clustering algorithm.
3.1.4 Our Algorithm

Below I outline a novel extension of the typical heuristic-simulation methods identified in the literature [27][44]. Our proposed method 1) decomposes the system into primary functional subunits while identifying secondary relationships across subunits, 2) captures the variation in time of these relationships, and 3) provides an approach that is applicable to any biochemical pathway described by a system of ODEs.

Proposed algorithm:

**GENERATE TYPICAL TRAJECTORIES**

I make use of the fact that by exciting one and only one element at a time, it is possible to distinguish which components respond to that excitement. I hypothesize that exciting the system in this heuristic way creates trajectories that are separable in high dimensions, enabling categorical grouping by clustering algorithms. Also, this behavior should be manifested in any system of ODEs, which leads to a good general solution for choosing the ICs. Let the number of trajectories $M$ equal the number of variables in the system $n$, i.e., $M = n$, and let $x^{max}$ be the maximal concentrations of a reference simulation. The ICs $x^{(h)}(0)$, $h \in [1, M]$, $k \in [1, n]$ are defined as:

$$ x^{(h)}_k(0) = \begin{cases} x^{max}_k & \text{for } k = h, \\ 0 & \text{else} \end{cases} \quad (3.8) $$

where (3.8) defines the IC of the $h^{th}$ trajectory of the $k^{th}$ component. This set of $M$ ICs leads to $M$ different trajectories $\xi^{(h)}(t) = \xi(t, x^{(h)}(0))$, with $0 \leq t \leq T - 1$, $x^{(h)}(0) = (x^{(h)}_1(0), ..., x^{(h)}_n(0)) \in \mathbb{R}^n$.

Previous literature [27] suggested this solution in some examples, but they did not propose this as a general solution. My results, however, indicated that other approaches for producing sample trajectories, such as using randomized initial conditions, produced results with poor performance when clustering in high dimensions.

**NORMALIZE DATA**

In this step, I normalize the data, as specified in (3.2)-(3.4).

**COMPUTE ACTIVITIES**

I propose the generation of activity vectors, where a component is active if the absolute value of its derivative is larger than a threshold; a component thus, is active if it is varying. This is in contrast to [27], where activities were also used, but whose definition of activity was not consistent for all the components (see (3.5), (3.6)). This eliminates the need for a priori
knowledge of the characteristics of the components in the system. I quantize the activity of one component at time $t$ over $m$ discrete levels to incorporate further diversity in the resulting trajectories. For $m = 3$, the activity at time $t$ of component $i$ would be computed as:

$$a_k(t) = \begin{cases} 
1 & \text{if } |f_k^*(t)| > |f_k^μ|, \\
0.5 & \text{if } 0 < |f_k^*(t)| < |f_k^μ|, \\
0 & \text{if } |f_k^*(t)| = 0.
\end{cases}$$

(3.9)

where

$$|f_k^μ| = \text{mean}_{h \in [1,M], 0 \leq t \leq T-1} |f_k(ξ^{(h)}(t))|$$

Activities at each time can, in principle, be defined in any number of levels $m$. My experience however, enforces what Ederer et al. already stated [27]: a low number of levels performs best. Ederer et al. suggested a discretization level of $m = 2$, but my analysis indicates that this number may vary for different applications. Higher values of $m$ incorporate the effects of small variations of the data, resulting in an increase of noise. In some applications however, a discretization in 2 levels may not give enough information to find secondary relationships. This phenomena can be easily identified, since the clustering algorithms that will be used in the following steps return empty clusters in this situation. Increasing $m$ is appropriate in this scenario.

I construct activity vectors in a way that is conducive to elucidating primary and secondary relationships between components. As opposed to other proposed methods [27], I do not aggregate all of the information for a given component over all time and ICs as shown in (3.7). I hypothesize that the aggregation of this information hinders insight into potential relationships or grouping that change over time. Instead, I construct activity vectors based on the dynamics of a component at a given time $t_i$ over all ICs. For example, Fig. 1a shows sample trajectories of 3 components for 3 different initial conditions and sampled at 4 discrete times $t_i$, $i = 1...4$. The approach presented in [27] generates a single activity vector for a component $x_k$ (Fig. 1b). Alternatively, my approach generates 4 different activity vectors for a given component, $x_k$, one for each sampled time $t_i$, $i = 1...4$ (Fig. 1c). Each of these activity vectors contains information about how $x_k$, at a given time $t_i$, responds to all ICs. These activity vectors provide the necessary information to 1) construct time specific functional decomposition of pathway components and 2) construct primary and secondary relationships between components based on this time specific decomposition.
FUZZY CLUSTER ANALYSIS

In this step, I implement a fuzzy clustering approach that allows us to build a dissimilarity matrix, which reflects transitions in primary and secondary relationships over time. I use this dissimilarity matrix to construct functional modules that highlight global primary and secondary relationships across components. I describe my approach below.

1. Perform an initial fuzzy cluster analysis at $N$ time points:

I use my resulting activity vectors, in combination with a fuzzy clustering approach to generate time dependent fuzzy clusters, from which I extracted tiered relationships between components in the system. In the previous step, I generated tiered relationships between components in the system. In the previous step, I generated activity vectors from each component, $x_k$, $k = 1, \ldots, n$, for each sample time $t$. I use these activity vectors as
the input feature vectors to the clustering algorithm to assess similarities in the dynamic activity across components. Thus at each $t_i$, I had a collection of $n$ feature vectors whose $M$ elements corresponded to the activities obtained from the $M$ sample trajectories or realizations. I use the $L_1$ norm, $\| \cdot \|_1$, (or Manhattan Distance) as the metric for measuring the distance or dissimilarity between all feature vectors collected at time $t_i$ [2]. The use of fuzzy clustering algorithms highlights the fact that each component can have varying degrees of similarity between functionally diverse groups or clusters.

At each time $t_i$, the fuzzy clustering algorithm produces a cluster set $C(t_i) = \{C_j(t_i)\}$, $j = 1,\ldots,\zeta$, where $\zeta$ is the total number of clusters in the cluster set. Each cluster $C_j(t_i)$ in the cluster set $C(t_i)$ is defined as

$$C_j(t_i) = \{x_1^{d_{ij}}, \ldots, x_n^{d_{ij}}\},$$

$$j \in [1,\ldots,\zeta], i \in [0,\ldots,T-1].$$

Equation (3.10) shows that the individual clusters of each cluster set, $C(t_i)$, capture the degree of membership, $d_{ik}^j$, of each component, $x_k$, $k = 1,\ldots,n$, to that specific cluster $C_j(t_i)$. The membership value $d_{ik}^j$ can vary from 0 (no membership) to 1 (full membership). The use of fuzzy clustering to capture tiered relationships provides a stark difference from Ederer’s work, whose initial clustering and final results were based on approaches that only produced mutually exclusive clusters.

Specifically, I use Fuzzy C-mean (FCM) clustering (with a fuzzifier $f = 2$) in my approach, although other algorithms can be used. FCM has been shown in the literature to perform well in other biological applications, specifically the clustering of microarray data [24][30]. These applications have demonstrated the robustness of FCM to difficult and noisy clustering problems. FCM, however, is sensitive to the data points chosen to initialize the algorithm, i.e., the initial centroids. I preprocess the clusters using K-means to address this problem. I run K-means for 6 different ICs, and the best partition—the partition that minimizes the objective function of the algorithm—is chosen as the initial partition of FCM. This process is repeated at each time point $t_i$, leading to an improvement of the stability of the clustering process.

2. Build a matrix of dissimilarities:

I use the time dependent cluster sets $C(t_i)$, $i = 0,\ldots,T-1$, to generate a dissimilarity matrix, $D$, that captures primary and secondary relationships between the components in the system. This is done by generating a scoring metric that quantifies primary and
secondary relationships between components in the system. I use this scoring metric to build a similarity matrix $S$, which is then converted to a dissimilarity matrix $D$ using the approach outlined in [88].

I start by first defining the primary members of each cluster $C_j(t_i), C_j(t_i) \in \mathcal{C}(t_i)$. A component $x_k$ is a primary member of cluster $C_j(t_i)$, if $C_j(t_i)$ is the cluster where $x_k$ has the maximum degree of membership $d_{kj}$ over all $j$. I use this definition of primary membership of each cluster $C_j(t_i)$ to define pairwise primary and secondary relationships between each components $x_k$ and $x_l$. A component $x_k$ is primarily related to component $x_l$ at time $t_i$ if both components are primary members of cluster $C_j(t_i)$, i.e., $C_j(t_i)$ contains the maximum membership values $d_{kj}$ and $d_{lj}$ for components $x_k$ and $x_l$, respectively. A component $x_k$ is secondarily related to component $x_l$ if the second largest membership value of $x_k$, again over $j$, $d_{kj}^2$, occurs in the cluster where $x_l$ is a primary member. I calculate the primary score, $P_{score}$, and the secondary score, $S_{score}$, as the frequency in which component $x_k$ is primarily or secondarily related, respectively, to component $x_l$ over all time $t_i, i = 0, ..., T - 1$. The similarity matrix, $S$, is constructed using a linear combination of these primary and secondary scores where the $(k,l)$ element of the similarity matrix is defined as

$$s_{kl} = \frac{P_{score}^{(k,l)}}{T} + \alpha \frac{S_{score}^{(k,l)}}{T}, \quad \alpha < 1.$$  

(3.11)

Here, $\alpha$ is a weighting factor that allows us to reduce the influence of the secondary relationships. I have fixed the value of $\alpha$ to $\alpha = 0.5$, as I experimentally determined that this value performed well in my applications. I confine my analysis to primary and secondary relationships so as to not include spurious or noisy connections. This allows us to highlight relationships across components that reside above the noise floor.

Matrix $S$ contains all the information of the similarities among components. It captures primary and secondary relationships that I identified over all time $t$. I cluster this data to achieve global relationships that take into consideration primary and secondary relationships across components. I follow a known gene coexpression analysis method [47][37][88] to do this. Clustering algorithms however, typically use the dissimilarities to cluster them. Matrix $S$ thus, is transformed into a dissimilarity matrix $D$. I follow the approach presented in [88] to convert matrix $S$ into an adjacency matrix, $A$, and then into the dissimilarity matrix $D$. The details of this transformation follow:

- **Define an adjacency function and an adjacency matrix:**

  In this step, I aim to convert the measure of the concordance between components (encoded in $S$) into a measure of node connectivity (encoded in a new matrix $A$).
We apply an adjacency function to each element of the matrix $S$. The adjacency function is a monotonically increasing function that maps the interval $[0, 1]$ into $[0, 1]$ [88]. I use the power adjacency function, which leads to a soft thresholding, or a weighted network:

$$a_{kl} = \text{power}(s_{kl}, \beta) = |s_{kl}|^{\beta}$$

where $\beta = 2$. The $(k,l)_{th}$ entry of matrix $A$, $a_{kl}$, encodes the connection strength between component $k$ and $l$.

- **Define a measure of dissimilarity:**

In this final step, I use a topological overlap measure [69] to convert the adjacency matrix into a dissimilarity matrix [69]. This is a measure of dissimilarity based on the topological overlap matrix (TOM), which is defined as:

$$TOM = \omega = [w_{kl}]$$

$$w_{kl} = \frac{l_{kl} + a_{kl}}{\min(k_k, k_l) + 1 - a_{kl}}$$

where

$$l_{kl} = \sum_{u=1}^{N} a_{ku}a_{ul} \quad \text{and} \quad k_k = \sum_{u=1}^{N} a_{ku}.$$  

The dissimilarity matrix, $D$, is finally defined as:

$$d_{kl} = 1 - w_{kl}$$

For further details of this transformation, see [88].

3. **Perform cluster analysis of $D$:**

In this step, I obtain global relationships across components using the primary and secondary relationships contained in $D$. I do this by identifying an appropriate clustering approach that can identify functional groups, while still conserving the tiered characteristics of these relationships. Applying a clustering algorithm that only identifies mutually exclusive clusters would ignore the tiered relationships that we know exist between the components of the system. Thus, a fuzzy clustering approach, similar to that used at each time $t_i$, would better identify the varying degrees of relationships contained in $D$. The fuzzy clustering algorithm used previously, however, identified fuzzy clusters based on feature vectors. In contrast, the matrix $D$ contains relational information that must be handled in a slightly different way.
We use the Relational Fuzzy C-Means algorithm (RFCM) [34] to cluster the matrix $D$. The RFCM algorithm results in a final cluster set $C_F = \{C_j\}$, $j = 1, ..., \zeta$, where

$$C_j = \{x_{d_j}^1, ..., x_{d_j}^n\}.$$  

(3.16)

Similar to before, each cluster $C_j$ in the final cluster set $C_F$ contains the degree of membership, $d_{k,j}$, of each component, $x_k$. This final result allows us to identify functional groups of components, where each component in the system can be ‘members’ of multiple functional groups with varying degree. I utilize an adjustable threshold $\rho$ to reduce the number of spurious members shown in any given cluster $C_j$. Here, $\rho$ is a component dependent threshold whose value is set to the mean of all membership values of a component $x_k$, or $\rho = \mu(d_k)$. I found that this value highlighted the components whose secondary memberships were notably higher than the tertiary and higher order memberships. This value, however, can be heuristically adjusted, particularly in applications where secondary relationships are subtle. It is important to note that the level of the threshold does not affect the results, but only adjusts the amount of high order relationships that will be shown.

3.1.5 VALIDATION OF THE RESULTS:

CLUSTERING VALIDATION

Clustering validation is used to identify the most appropriate number of functional groups or clusters, $\zeta$, that can be extracted from the activity vectors. The clustering techniques used in my method, k-means and FCM, require a priori knowledge about the number of clusters contained in the data set, i.e., they perform a partition for a given number of clusters. In cases where the user is unaware of the number of clusters—functional groups—contained in the model, clustering validation provides a useful mechanism for identifying the number of clusters that maximize or minimize some information criterion.

An alternative way of calculating the most appropriate number of clusters would be leveraging the FCM’s objective function. This is done by performing multiple implementations of FCM for different numbers of clusters, then evaluating the objective function and choosing the number of clusters that minimizes this objective function. This alternative, however, can lead to inadequate results. The objective function of FCM seeks to minimize the cluster size. Performing FCM for different numbers of clusters results in a decrease of the objective function as the number of clusters increases. Thus, it is better suited to use alternate clustering validation approaches to find the appropriate number of clusters.

I implement the clustering validation in two steps. I first perform a validation of the number
of clusters at each time point. I then compute the final number of clusters \( \zeta \) combining the information of the validation applied at each time point. These steps are computed in my method prior the fuzzy cluster analysis described in Section 3.1.4. After this validation, I use \( \zeta \) to perform clustering at each time point, again, as explained in Section 3.1.4. The steps follow:

1. **Validation of the number of clusters at each time point:**

   In this step, the validation index is calculated at each time point, resulting in the optimal number of clusters at each \( t_i \), which I will call \( \zeta_{i^{\text{opt}}} \). Our approach requires a validation index that operates with an absence of information about the true classes, since I do not assume to have *a priori* information about the clusters. These type of indices are known as internal validation indices. I choose to use the Silhouette index, since it almost always outperforms other internal indices [8]. The Silhouette index works under the assumption that the result of a clustering algorithm should return clusters that are compact and apart from each other. This leads to two validation criteria: 1) a measure of the cluster size, i.e., how far apart the elements inside a cluster are, and 2) a measure of the clusters distances, i.e., how far apart the elements from distinct clusters are.

Indices that only seek to minimize the cluster size typically find that partitions with high numbers of clusters optimize this criterion, independently of the distribution of the data. These indices behave similarly to the objective functions of K-means and FCM, for they also seek to minimize the cluster size. Thus, the two criteria of the Silhouette index present an advantage over validation indices that measure only the cluster size.

The Silhouette index [8] is defined as:

\[
S = \frac{1}{\zeta} \sum_{j=1}^{\zeta} \left[ \frac{1}{n_j} \sum_{x \in C_j} S(x) \right]
\]

(3.17)

where \( x \) is a point in cluster \( C_j \), \( n_j \) is the number of points in \( C_j \), and \( S(x) \) is the silhouette width of the point \( x \), defined as the ratio:

\[
S(x) = \frac{b(x) - a(x)}{\max[a(x), b(x)]},
\]

(3.18)

where \( a(x) \) is the average distance between \( x \) and the rest of the points in \( C_j \), and \( b(x) \) is
the minimum among the average distances between $x$ and the points in the other clusters:

$$a(x) = \frac{1}{n_j - 1} \sum_{y \in C_j, y \neq x} d(x, y),$$

(3.19)

$$b(x) = \min_{h=1,\ldots,K, h \neq j} \left[ \frac{1}{n_h} \sum_{y \in C_h} d(x, y) \right].$$

(3.20)

The global Silhouette index defined in (3.17) is the average, over all clusters, of the silhouette width of their points [8].

We use this index to find the optimal number of clusters at each $t_i$, $\zeta_{i}^{\text{opt}}$, for my data set—where optimal refers to the number of clusters that maximizes (3.17). This is done by performing FCM at each time $t_i$, for $i = 0 \ldots T - 1$ for increasing values of $\zeta$, and choosing the number of clusters $\zeta_{i}^{\text{opt}}$ that results in the largest value of (3.17).

2. Calculation of the final number of clusters:

The variations over time of the data, along with the randomness associated with the ICs of the clustering algorithms at each $t_i$, result in variations of $\zeta_{i}^{\text{opt}}$ over time, i.e., $\zeta_{i}^{\text{opt}}$ and $\zeta_{j}^{\text{opt}}$ need not achieve the same value, for $i, j \in [0, T - 1]$, $i \neq j$. The final result, however, should provide a fixed number of functional groups that combines information of the intermediate clusterings at each $t_i$. Combining clustering results that had distinct number of clusters may lead to results that are not biologically relevant. As a result, I use a single value, $\zeta_{\text{avg}}$, which was generated by averaging $\zeta_{i}^{\text{opt}}$ over all $i$, to generate the clusters from which primary and secondary relationships were extracted. I hypothesize that this average exhibits the most representative behavior of the data. This adds a critical level of validation that is not included in previous proposed methods [27].

VALIDATION OF THE FINAL RESULT

I establish the biological relevance of my final results by testing it on an example whose modularity has been previously studied—the EGF induced MAPK [74]—and on an example that has been deeply investigated—the C3 photosynthesis model [45][46]. I use previous literature [27][18][45][46] to compare my results, and to certify the quality and biological relevance of my results.
3.2 A Computational Framework for Identification of Components of a Model Relevant to a Phenotype

3.2.1 Introduction

Models of many low level biological processes from numerous organisms such as Homo sapiens, Drosophila melanogaster, or Arabidopsis thaliana, have been developed over the years, and often collected in databases [64][1][61]. The understanding of phenotypes however, is typically the final goal of researchers and investors. Important phenotypes include increase of biomass, elimination of human diseases, and production of biofuels. The question now is how these collections of models can be used to study and explain phenotypes of the organisms that the models belongs to.

As it was explained in Section 2.1, there are layers of abstraction between the low level processes and the phenotypes that are not mathematically modeled, and thus, there is a disconnect between them. As a result, we are typically reduced to blindly relating phenotypes to individual components using experimental quantification. This consists of performing measurements to relate components of the process and the phenotype, often with the goal of constructing a mathematical model leveraging the collected data. This option however, can be extremely expensive, time consuming, and undoable in some cases. In the design of new drugs, for instance, knocking down a target component may not suffice to achieve a pursued effect, due to the redundancy that exist in biological systems. Then, experiments with multiple components may need to be performed to identify all the components that have an effect on the expected outcome or phenotype of interest. The PI3K signaling pathway and cancer is an example of a relationship between a model and a phenotype that scientists are interested in knowing, and whose investigation is taking years.

An alternative to the experimental solutions and to the mathematical models would be the generation of black box models that have lower needs for resources. A black box model would not provide a description of the abstract phenomena in terms of the components of the low level biological system. It would, however, be able to predict the phenotype as a function of the components of the model. Black boxes have less needs for mechanistic knowledge. This would result in a method that could be applied when a mechanistic model is not doable, and that could save time in cases where mathematical models are pursued. A black box modeling approach with the described goal, would need to meet two criteria: 1) capability to work with time series data, to be be able to process the experimental measurements of components and phenotypes, and 2) low needs for experimental data.

I present in this section a method that addresses this problem. The specific purpose of the method is to find the components that significantly impact some higher level phe-
**notype.** In relation to the photosynthesis pathway and the photosynthetic rate example, my algorithm could find the groups of enzymes that need to be increased to improve the efficiency of the photosynthetic rate, based on a set of temporal measurements of the photosynthetic rate versus CO$_2$—which is a component in the pathway, known to be a good controller of the photosynthetic rate. I will show the application of my method to this example in the next chapter.

For the PI3K signaling pathway and cancer example, my method would leverage measurements from one component of the pathway of interest and from the phenomena under study (inhibition of some symptom, size of a tumor, inhibition of a pathway, etc) to provide the group of key components that need to be targeted to achieve the pursued phenomena—without requiring experimental data from all of the components (see [53][15] for typical experimental data between tumors and some component).

The algorithm achieves its goal in two steps. It first needs to generate a non mechanistic model to connect the low level biological process and the phenotype. As explained in Section 2.5, HMMs are a good fit for this task. They are specially useful for temporal applications, and, as opposed to other machine learning tools, HMMs are not required to work with multiple input feature vectors to improve their performance. Input feature vectors are, in terms of HMMs, observation vectors. In my case, these observation vectors correspond to the component concentration measurements over time. Low needs for input feature vectors thus, translates in my case into low needs for number of components to be measured. This makes of HMMs an advantageous approach versus classical mechanistic models, where *mechanistic* refers to models that describe system in terms of their integral parts.

In the second step, the algorithm needs to leverage the black box model to identify the components of the model that are relevant to the phenotype. Sensitivity Analysis (SA) algorithms are a suitable tool to address the challenge that I seek to solve. SA aims to quantify the impact of each input or input parameter on the output of a system. This is measured by computing the change in the output due to a perturbation of each of the inputs. Among all types of SA approaches, LOAT methods will be used in my algorithm, for this is an effective approach when analyzing biochemical systems. Recall from Chapter 2 that SA methods need to know the relationship between the inputs and the output of a system to perform the analysis. The black box model generated in the first step provides the relationship between different levels of abstraction needed for the SA algorithm to operate.

This novel algorithm can contribute to save experimental time in studies where components that impact an abstract phenomena need to be found. My algorithm could be used to identify all the components that need to be targeted without adding experimental time, and also to find components that influence a pathway and a phenotype without going through the long process of building a mechanistic model. For cases where developing a mechanistic model of a
phenotype in terms of biological processes is not doable, my algorithm could be used to make preliminary studies of the phenotype in terms of the components. As far as I know, this is the first time that an algorithm—non specific to a particular biological process—can relate several levels of abstraction in a biological model to find critical components to control an abstract phenomena.

The contributions of this algorithm are:

1. It finds key components in an organism to control a phenotype of interest.
2. It combines SA and black-box modeling, resulting in a technique that can assess phenotype sensitivity across different levels of abstraction.
3. It can save experimental time in studies where components that impact an abstract phenomena need to be found.

The next section presents the SA/black-box method that I developed, while in the next chapter, I will apply it to the well known photosynthesis pathway. I identify groups of components from the model that impact a phenotype of interest—which is the photosynthetic rate. I compare my results with previous experimental and/or mathematical analyses of the model. I show that my algorithm successfully identifies, in an automated way, the components that impact the photosynthesis.

3.2.2 An Algorithm to Relate Model and Phenotype: A SA/Machine Learning Approach

Introduction To The Algorithm

Given a dynamic biological system at the pathway level and a phenotype of interest, my algorithm aims to find which components from the system impact the phenotype. Two main tasks need to be addressed to achieve this goal. The first one is to fill the disconnect that exists between the pathway model and the phenotype where this disconnect is produced by the fact that they reside in different levels of abstraction. For this, I develop a black box model that patterns the relationship between the pathway and the phenotype. This model is built based on data measurements of the phenotype versus some component of the pathway. The second task that needs to be done is a study of the impact of each component of the pathway on the phenotype. I use the generated model to find out how changes on the pathway components affect the phenotype. SA methods address this type of problems. Specifically, they aim to quantify the change of a system’s output due to a perturbation of each of the inputs. In my case, the SA is applied to the system that models the relationship between the pathway and the phenotype. The inputs of this system are the components of the model, while the phenotype
is the output. The SA method is applied in this context, leading to the key components of the pathway that impact the phenotype of interest. The format of the data used to build the black box model, which is the starting point of the algorithm, is described in the following section.

The Data

The structure of the data that my algorithm requires, responds to a format that aims to 1) minimize the amount of experimental measurements required, and to 2) be in line with the data that biologist typically collect in studying phenotypes [53][15][22].

When a phenotype of an organism is investigated, a typical procedure for biologists starts with collecting measurements from the phenotype in relation to some important component of the organism. The measurements are often collected as time series, as well as they may be repeated for various conditions. For example, if the growth of a plant needs to be studied, measurements of the plant size with respect to the CO\textsubscript{2} assimilation levels may be taken [66]. Other examples could be the study of the growth of a tumor, where measurements of the size of the tumor, and measurements of some component that is suspected to play a role in cancer development are taken [53]. I leverage this format of data to build the connexion between the model and the phenotype. These types of measurements are typically taken as an initial pool of information to perform preliminary analysis in the study of a phenotype. I expect thus, that the data may be readily available when my algorithm needs to be applied as a leading approach for model reduction in relation to a phenotype, or when it needs to be applied to perform a preliminary study to find components that impact a phenotype.

The Algorithm

1. **Generate The Black Box Model**

   The first step of the algorithm consists of building a black box model representing the relationship between the pathway and the phenotype. This covers the disconnect existing between them. I rely on experimentally collected data to build the black box. The data is of the form explained in the previous section. As it has been already presented, HMMs are the black box modeling tool that best fits the type of data that is available.

   Building an HMM consists of estimating its constituent parameters, such that the resulting HMM models my system of interest. These parameters are the entries of the state transition probability matrix, $A$, and the entries of the observation probability matrix, $B$ (see Section 2.5.1), and in my case, the system is the relationship between the model and the phenotype. The estimation of the parameters is done, as mentioned, leveraging data that has been collected from the system to be modeled. In terms of the HMM, this data contains a sequence of the states of the system, and a sequence of the output of the sys-
tem. This scenario is pictured in Fig. 3.3 by means of a toy example of a pathway and a phenotype. The goal of this step thus, is to estimate the probability of transitioning from one state to another one (state transition probability matrix, $A$), and the probability of observing a symbol given a particular state (observation probability matrix, $B$), provided a sequence of states from a state machine, and a sequence of observations corresponding to those states—outputs of the machine.

To achieve this task I need to 1) estimate the number of parameters that are necessary to fit a set of data, i.e., estimate the sizes of $A$ and $B$, and 2) estimate the values of such parameters. These steps are further explained as follows.

(a) **Estimate the number of parameters needed to fit the set of data:**

I start this step by discretizing the amplitude of the data. The number of the dis-
cretization levels of the observation vector (measurements of the internal component) dictates the number of observation symbols that we have (each discretized value corresponds to a symbol), while the number of the discretization levels of the state vector (phenotypical measurements) dictates the number of states (each discretized value corresponds to one state). In Fig. 3.4, I show a toy example to evidence this process. This example shows a pathway and a phenotype. Measurements from one component of the pathway and from the phenotype are taken at 5 time points. The value of the measurements are discretized in a finite number of levels, i.e., the amplitude range of the measured values is divided in several levels, and all the measurements take discrete values according to these levels. In the figure, the phenotypical measurements have been divided in 3 levels, while the component measurements have been divided in 2 levels. This leads to a total of 3 possible states (0,1,2) of the system to be modeled, and 2 possible observation symbols (0,1) of the same system. The HMM, then, requires a $3 \times 3$ state transition probability matrix, $A$, and a $3 \times 2$ observation probability matrix, $B$, to model this system.

(b) Estimate the parameters of the HMM to fit the set of data:

Here, I want to estimate the values of $A$ and $B$, such that given the vector of measured component concentrations (observation sequence, $O$), the HMM can predict a vector that matches the vector of measured phenotype (state sequence, $Q$). In Fig. 3.3, we can see how these two time series are used to fit the parameters of the
Figure 3.4: Toy example to show how the level of discretization of the measurements leads to particular sizes of the state transition probability matrix, and the observation probability matrix.

Figure 3.5: Estimation of the HMM’s parameters by minimizing the error between the measured and the estimated phenotype values, given a sequence of a component’s concentration values.

HMM to this data. In Fig. 3.5 we can see that the HMM takes the observations (the internal component measurements) as its inputs, and it predicts the phenotype. The real measurements of the phenotype are used to adjust the parameters of the HMM, such that they minimize the error between the predicted vector of phenotypes and the real measurements. I implement the HMM in Matlab, leveraging the maximum likelihood estimate to calculate the state transition probability matrix and the
observation probability matrix based on some given data. Details of the maximum likelihood estimate of the HMM’s matrices follow.

**Maximum-Likelihood Estimates of the Parameters of the HMM:**

Given a state sequence, \( Q = q_0, q_1, ..., q_{T-1} \), where each \( q_t, \forall t \in [0, T-1] \), can take values in \( S = \{s_0, s_1, ..., s_{N-1}\} \), and an observation sequence, \( O = (O_0, O_1, ..., O_{T-1}) \), where each \( O_t, \forall t \in [0, T-1] \), can take values in \( V = \{v_0, v_1, ..., v_{M-1}\} \), we want to find the model, \( \hat{\lambda} = (\hat{A}, \hat{B}, \hat{\pi}) \), that maximizes the probability of producing \( Q \) and \( O \).

Define \( \alpha_{ij} \) as the number of times that state \( j \) follows state \( i \) in the sequence \( Q \):

\[
\alpha_{ij} = \sum_{t=0}^{T-1} [q_t = s_j \land q_{t-1} = s_i]
\]

Then, the maximum-likelihood estimates of the transition probabilities \( a_{ij} \), are:

\[
\hat{a}_{ij} = \frac{\alpha_{ij}}{\sum_j \alpha_{ij}}.
\]

Define \( \beta_{ik} \) as the number of times that state \( i \) is paired with observation \( k \) in the sequences \( Q \) and \( O \):

\[
\beta_{ik} = \sum_{t=0}^{T-1} [q_t = s_i \land O_t = v_k]
\]

Then, the maximum-likelihood estimates of the observation probabilities \( b_{ik} \), are:

\[
\hat{b}_{ik} = \hat{b}_k(v_k) = \frac{\beta_{ik}}{\sum_k \beta_{ik}}.
\]

The maximum-likelihood estimates of the initial state probabilities, \( \pi_i \), are defined as \( 1 \) if state \( i \) is the initial state in the sequence, and \( 0 \) otherwise:

\[
\hat{\pi}_i = \begin{cases} 
1 & \text{if } q_0 = s_i \\
0 & \text{if } q_0 \neq s_i 
\end{cases}
\]

The previous explanation has been presented for a given number of discretization levels for the state vector and the observation vector. I have not addressed yet how to choose the number of levels. The appropriate number of the discretization levels will depend on the number of data points that are available, and the level of detail of the prediction needed. A finer discretization of the states will allow more level of detail in the future.
predicted hidden states. The number of parameters that we need to fit however, rises with the number of states and observation symbols, and so does the amount of data needed. In my algorithm, I determine the number of the discretization levels experimentally. I do this by building multiple HMM configurations—we vary the number of discretization levels of the data, which leads to HMMs with different number of states and observation symbols—, and by choosing the configuration that minimizes the error of the prediction.

The process of fitting the parameters of the HMM to the data allows that the HMM, in the future, will be able to predict the values of the states—hidden states, in this case—, given a new vector of observations. This property allows us to use the HMM as a model—a black box model, since it does not provide a mechanistic description of the system—that will relate the states (components of the pathway) with the observations (phenotype). In the following sections, I will denote the black box model generated as H.

2. **Predict the phenotype under typical conditions of the model**

In the previous step, I built a black box model, H, to relate a phenotype of interest, and a biological process that I want to use to manipulate the phenotype. I built H based on experimental measurements of a component that participates in the biological process, and the phenotype of interest. In this step, I want to predict the phenotype that would be produced for typical values of a biological process, given a mathematical model of the biological process in question, and H. The mathematical models that I work with are described by systems of ODEs of the form:

\[
\dot{x} = f(x, p), \quad x \in \mathbb{R}^n, \quad p \in \mathbb{R}^q
\]

where the elements of \( x \), \( x_k \), represent the biological components involved in the pathway, and the vector \( p \) represents the \( q \) kinetic parameters of the pathway. The vector function \( f(x) \in \mathbb{R}^n \) characterizes the change in component concentration over time. I define \( \xi(t, x(0)) \in \mathbb{R}^n \) as the solution of (3.1) for some time \( t \), and with the initial condition (IC) \( x(t = 0) = x(0) \in \mathbb{R}^n \). \( \xi(t, x(0)) \), also called the trajectories of the system, describe the variation in time of each of the component concentrations.

I am interested in using H and a mathematical model of the biological process such as (3.26) to predict typical values of the phenotype—sequence of phenotypical values for a time period—for the biological process in question. The goal of having a vector of typical values of the phenotype will be to use it as a reference vector for the following steps of the algorithm, where changes to the model will be performed, and new predictions of the phenotypical sequence will be compared to the reference one. Let us continue by providing
the definition of some notation:

- **Typical conditions** of the models, \(x(0)^{typ}\): initial conditions (ICs) that the components of the model have been experimentally tested to be typical for the described biological process.
- **Typical trajectories**, \(\xi^{typ}(t, x(0)^{typ})\): change of concentration of each of the model’s components over time when the starting values of the concentrations are the typical ICs.
- **Observation component**, \(x_{obs}\): component from the model that coincides with the component whose experimental measurements were used in the previous step to generate \(H\).
- **Typical observation trajectory**, \(\xi^{typ}_{obs}(t, x(0)^{typ})\): typical trajectory of the the component that I use as my observation, \(x_{obs}\).

We use the typical observation trajectory, \(\xi^{typ}_{obs}(t, x(0)^{typ})\), and the trained HMM, \(H\), to predict the phenotypical vector. In terms of HMMs, \(\xi^{typ}_{obs}(t, x(0)^{typ})\) acts as the observation sequence, and the predicted vector of phenotypes is the predicted vector of states—hidden states. I denote the phenotypical vector that has been predicted for typical initial conditions as the **typical phenotypical vector**, \(P^{typ}\).

As before, I perform this step in Matlab. Matlab predefined HMM’s functions use the Viterbi algorithm to, given a sequence, \(\xi^{typ}_{obs}(t, x(0)^{typ})\), calculate the most likely path, \(P^{typ}\), through the HMM specified by the transition probability matrix, and emission probability matrix. More details of this process follow.

**Viterbi Algorithm to Estimate the Hidden State Sequence:**

Given a model, \(\lambda = H = (A, B, \pi)\), and an observation sequence, \(O = \xi^{typ}_{obs}(t, x(0)^{typ})\), we want to find a maximum probability state sequence, \(Q = P^{typ}\), for the Markov process. The Viterbi algorithm can be used to recursively solve this problem in the following way:

Define \(\delta_i(t)\) as the probability of the most probable sequence ending in state \(i\) at time \(t\):

\[
\delta_i(t) = \max_{q_0,...,q_{t-1}} P(q_0, q_1, ..., q_{t-1}, q_t = i, O_0, O_1, ..., O_t | \lambda)
\]  

(3.27)

and \(\pi_i\) as the initial probability of state \(i\).

Then \(\delta_i(t)\) can be calculated recursively as:

\[
\delta_i(t) = \max_{1 \leq j \leq N} [\delta_j(t-1)a_{ij}]b_j((O_t)),
\]

(3.28)
where

$$\delta_i(1) = \pi_i b_i(O_0), \quad \text{for} \quad 1 \leq i \leq N$$ \hspace{1cm} (3.29)

The maximum probability state sequence, $Q$, will be given by the highest probability final state, followed by the highest probability path:

$$Q = \max_{1 \leq i \leq N} [\delta_i(T)].$$ \hspace{1cm} (3.30)

3. **Predict change in phenotype under perturbed conditions of the model**

In this step, I want to perturb the IC of one component at a time to predict the new phenotypical vector after this perturbation. This will allow us to, in the next steps, measure the change of the phenotype with respect to the reference phenotypical vector, $P^{typ}$. For this, I proceed as follows:

The ICs of the model components are perturbed one at a time. I add a percentage $p\%$ of their typical value to their IC:

$$If \quad x_k(0) = \alpha_k, \quad then \quad x(0)_k^{perturbed} = \alpha_k + p\% \cdot \alpha_k$$ \hspace{1cm} (3.31)

The appropriate value of $p$ may vary for different applications as a function of 1) the magnitude of the variation of the phenotype as a consequence of the components perturbations, and 2) the number of levels of the discretization. The value of $p$ thus, can be determined heuristically for each application. In classical SA applications, perturbations can be of the order of $0.1\% - 1\%$. Thus, these are pertinent values to use in my method. In some applications, however, these values may be too small to reflect changes in the phenotype. In these cases, it would be appropriate to heuristically increase the value of the perturbation, until meaningful changes of the phenotype are detected.

After the perturbation of a component is performed, the system is solved to obtain the new trajectories. For the new trajectories, $x_{obs}$ has a corresponding “perturbed behavior”, or a “$k_{th}$ perturbed observation trajectory” $x_{obs}^{perturbed}(t, x(0)_k^{perturbed})$, which is stored. Index $k$ refers to the $k_{th}$ component that has been perturbed, for $k \in [1, n]$. The process is repeated until all of the components have been perturbed one at a time.

The phenotype is again predicted by means of the black box model $H$, and each of the perturbed observation trajectories $x_{obs}^{perturbed}(t, x(0)_k^{perturbed})$, $\forall k \in [1, n]$. This results in $n$ different predicted “perturbed phenotypical vectors” $P^{pert}$, for $k \in [1, n]$.

4. **Compute the sensitivity coefficients with respect to the inputs of the model**
Here, I aim to find the influence of model components on the output of interest (phenotype). I use the previously computed perturbed phenotypical vectors, $P^{\text{pert}}$, to calculate the sensitivity coefficients of the phenotype with respect to the components of the biological process. The sensitivity coefficients allow us to assess the variation of the phenotype with respect to a variation or perturbation of each input, and they are defined as:

$$S_k = \frac{\|P^{\text{typ}} - P^{\text{pert}}_k\|_1}{p\% \cdot \alpha_k}, \quad \alpha_k = x_k(0), \quad \forall k \in [1, n] \tag{3.32}$$

where the difference between the reference phenotypical vector and the each perturbed vector is computed with the 1-norm, and $S_k$ measures the change in the phenotype relative to the change in the IC of component $k$.

I typically refer to the sensitivity coefficients with the term impact factors, for I do not calculate these coefficients in the traditional sense (a black box model relates the inputs and the output of the SA algorithm).

5. Find the components and functional units of the system with the largest impact factors

We finally identify the most influent components on the phenotype. The largest impact factors $S_k$ will correspond to the components that have the greatest impact on the phenotype. This finding will indicate which are the components that need to be preserved by a reduction algorithm. This can also be used to conduct a preliminary analysis of the components that need to be targeted by experimental studies to achieve some desired outcome of a phenotype. This, as previously stated, would reduce the amount of candidates that experiments should target.

The application to the model of the decomposition algorithm prior performing the SA, will also allow us to separate the components by modules, and thus, find out which are the functional units with largest impact factors.

3.3 Chapter Summary

In the previous chapter, I introduced the problems that motivated this research. These problems are concern about describing impacts of low level characteristics on upper level phenotypes, and connecting models at several levels of abstraction to achieve multihierarchial models. I also identified that the two main impediments to solve these problems were 1) the lack of mechanistic information about how interactions of low level components impact higher levels phenotypes, and 2) the complexity of the models that describe low level biological processes.
In this chapter, I presented a framework that embraces two algorithms to solve these problems. The first one was a decomposition method, which addresses the problem of the high model complexity by providing some intuitive information of the model. The second one was a method that relates a lower level model with some phenotype, addressing the problem of not having mechanistic information to relate lower level models and phenotypes. This research combines these two methods—a decomposition method and a computational structure—to identify combinatorial interactions between components that influence phenotypic traits of interest.

### 3.3.1 Decomposition method

Decomposition of dynamic systems is an important tool in the process of analyzing large complex systems. For the case of biological systems, mathematical models do not always provide the insight needed to understand control hierarchies or lower level functional relationships, and hence decomposition techniques are critical to provide more functional information. Decomposition of dynamic biological models such as biochemical systems, however, becomes challenging due to the large dimensionality of the systems of practical interest. Previous decomposition techniques for dynamic biological models focused on finding non overlapping functional groups. In biochemical systems, however, components can appear in multiple reactions, and thus, may be located in multiple functional modules. Relationships developed based on mutually exclusive decomposition methods, then, can lead to misleading results when the models that they are applied to contain a high percentage of repeated components in distinct reactions. I have presented in this chapter an algorithm that addresses this problem. My work is a novel extension of previous studies on biological pathways decomposition.

I presented a method that identifies primary and secondary relationships between components of biochemical pathways, and that can assess how relationships between components or groups of components change over time.

My hierarchical decomposition, along with the transient relationships can help us elucidate hierarchical control, giving strategic insight into components that are critical for controlling the pathway.

### 3.3.2 Model-to-phenotype method

The second technique that I presented is an identification of groups of components and parameters of a model that are relevant to a higher level of abstraction. The method finds the group of components that need to be targeted by experiments, in studies that aim some specific outcome on a phenotype. As far as I know, this is the first time that an algorithm—non specific to a particular biological process—can relate several levels of abstraction in a biological model to find critical components to control an abstract phenomena. The presented algorithm achieves
its goal by combining SA and black-box modeling. This results in a technique that can assess phenotype sensitivity across different levels of abstraction.
Chapter 4

Results

In this chapter, I apply the previous algorithms to the Epidermal Growth Factor (EGF) receptor-induced MAPK, presented in [74] by Schoeberl et al., and to the C3 photosynthesis model available in the BioModels database [50] and developed by Laisk et al. in [45] and [46]. The EGF induced MAPK and the C3 photosynthesis are two large systems from human and plant biology, respectively. I perform an exhaustive analysis of the application of the decomposition algorithm on the EGF induced MAPK application, which allows us to experimentally analyze the method in detail. Both of the algorithms are applied to the C3 photosynthesis model. The decomposition of the C3 photosynthesis application is shown more concisely than that of the EGF model, while the main focus of the photosynthesis model is on the SA/Black-Box Modeling method. My results and comparisons to characteristics previously identified in the literature demonstrate the utility of my approach.

4.1 EGF Induced MAPK

The EGF is a mammalian receptor that belongs to the tyrosine kinase family of receptors. EGF receptors are involved in embryonic and postnatal development and in the progression of tumors. Binding of EGF to the extracellular domain of the EGF receptor initiates two principal pathways, Shc-dependent and Shc-independent, leading to activation of Ras-GTP, and in turn, activation of the MAP kinase cascade.

The EGF induced MAPK model is available in the BioModels database [50], and has a total of 100 variables and 125 reaction rates. Mechanistically, this model describes the reactions from the biding of the EGF molecules to their receptors, to the phosphorylation of MEK, and the catalyzation of the dual phosphorylation of ERK. It includes the binding of the EGF, the formation of signaling complexes, the activation of Ras, and the activation of the MAP-kinase
cascade. This system also models the EGF receptor internalization.

The EGF receptor-induced MAP kinase cascade has been widely studied, which has lead prior system decomposition or reduction algorithm studies [72][27][3][18] to test their methods by means of this model. As stated in [3], decomposition algorithms can help us gaining understanding of the complete pathway in a systematic component-wise manner based on the system dynamics, for inferring the functionality of the complete pathway by directly examining the full system is very difficult due to the coupling present in the system equations. Particularly, and in spite of the fact that the EGF Induced MAPK is well understood, the decomposition of its components in primary and secondary relationships reveals information that is difficult to extract from the system. Our decomposition results in a clear map of connections among components and different functional units. This information can have an impact on, for example, studies about the relationship between mutations of the EGFR and cancer. For example, these maps of relationships could be used to help explaining how inhibition of EGFR signaling pathway can become a useful therapeutic strategy for tumor cells [76][59].

4.1.1 Analysis of Previous Decompositions of the EGF Induced MAPK

Previous computational studies have decomposed this model in several different ways [27][3][18], although they share a similar structure in their underlying classification. For example, Conzelmann et al. [18] present a decomposition that results in the following modules:

1. EGF binding
2. signaling complex formation without Shc
3. signaling complex formation with Shc
4. activation of Raf and Ras
5. the activation of MEK
6. the activation of ERK
where (5) and (6) form the MAPK cascade.

Ederer et al. [27] obtain the following modules:

1. EGF binding
2. Shc independent signaling complex formation
3. Shc dependent signaling complex formation and Ras activation
4. Shc independent Ras activation
5. MAPK cascade
6. negative feedback.
The main difference between both of them lies on the modules related to the Ras activation. The components associated to the \textit{Shc dependent Ras activation} combine with those from \textit{Shc dependent signaling complex formation} to form a module in \cite{27}. The \textit{Shc dependent Ras activation} however, pairs with the \textit{Shc independent Ras activation} to form a module in \cite{18}. These two decompositions seem contradictory. The results in \cite{18} are supported by the decomposition results in \cite{73}, where it is stated that the \textit{Shc dependent Ras activation} and the \textit{Shc indepen-
dent Ras activation are strongly coupled. The results in [27] are supported, as they indicate, by biologist. Analyzing the original reactions from [74] shows that none of these results are wrong. Both, however, convey an incomplete picture of the EGF induced MAPK pathway that can lead to misleading hypotheses. The Shc dependent Ras activation is strongly related to both, the Shc dependent signaling complex formation, and the Shc dependent Ras activation. The type of decomposition that they perform however, (i.e., a division of only primary relationships among components in nonoverlapping clusters), will be unable to show this. I list as follows some specific components that appear in two of these modules at the same time based on [74].

Component 28 (Ras-GTP), participates in the the reaction rates v19, v27, v28 (not including the internalized reaction rates). Reaction v19 belongs to the Shc independent Ras activation functional unit, whereas v27 and v28 belong to the Shc dependent Ras activation unit. Similarly,
component 43 (Ras-GTP*), participates in v20, v29 and v30. Reaction v20 belongs to the *Shc independent Ras activation* unit, whereas v29 and v30 belong to the *Shc dependent Ras activation* unit. A similar situation occurs with component 26 (Ras-GDP). We see that 3 of the 5 components that form the *Shc independent Ras activation* module, appear in the *Shc dependent Ras activation* unit, which has also 5 components. These two functional units thus, are very coupled. This is shown in Fig. 4.2, where the cited components are circled. We can now look at the second scenario: the 5 components that form the *Shc dependent Ras activation* unit, participate in reactions involved with the *Shc dependent signaling complex formation* (See highlighted reactions in Fig. 4.2): Component 36 (EGF-EGFR2-GAP-Shc-Grb2-Sos-Ras-GDP) participates in v26 and v27, component 37 (EGF-EGFR*2-GAP-Shc-Grb2-Sos-Ras-GTP) in v30 and v31, component 28 (Ras-GTP) in v27, component 26 (Ras-GDP) in v26 and v31, and component 43 (Ras-GTP) in v30. This could lead to think that they are indeed the same module, as Ederer proposed. However, if we keep in mind the strong coupling with the *Shc independent Ras activation* module, then the three of these modules could form one big module. We could then keep doing this until we combined all the system components in one unique module. Another solution would be to pick one path and miss some information, as Ederer et al. or Colzemann et al. did. The solution that I suggest instead, is the use of a hierarchical structure, which would find the primary relationships among components, and secondary relationships with components outside of its own module.

I just presented an example that shows a contradiction between the classifications of the EGF induced MAPK components in [18] and [27]. We could also look at other components that have been classified in the same way by both of the authors, and where both have failed to capture that they should be located in multiple modules—as indicated in the reactions described in [74]. We can look at component 22 (Grb2) (circled in Fig. 4.2) for an illustrative example. This component is directly reacting with components that are located in the module *Shc independent signaling complex formation* within the reactions v16 and v35. This component however, is also located (visible in the figure) in module *Shc dependent signaling complex formation*, within the reactions v24 and v38. This also occurs with others components located in *Shc independent signaling complex formation* (Refer to [74] for more information). All of them should be located in two different modules, but previous classifications need to pick one, as they generated nonoverlapping clusters.

We can also analyze modules *Shc independent Ras activation* and *Shc independent signaling complex formation*. Most of the components in *Shc independent Ras activation* participate in reactions in *Shc independent signaling complex formation*, but previous literature couldn’t show a relationship between these two groups of components. I show in my results how my algorithm successfully locates these and other components in several clusters, with primary and secondary
4.1.2 Our Decomposition of the EGF Induced MAPK

I first generate typical trajectories and compute the activities, which I discretize in $m = 3$ levels—two levels cause empty clusters, indicating that more detailed data is needed. I then find the most appropriate number of clusters needed to describe the dynamic data, according to the criteria specified in Section 3.1.5. I evaluate the Silhouette index for clusterings ranging from 1 to 9 for each time value $t_i$ and identify the value $\xi_{i}^{\text{opt}}$ that maximizes the Silhouette index. Attempts to decompose the model into greater than 9 clusters resulted in empty clusters and hence were not used in the evaluation of the Silhouette index. Figure 4.3 shows the evolution of the Silhouette index over time. The average of all values $\xi_{i}^{\text{opt}}$ over time results in the most appropriate number of clusters in the system. I found that this average, $\xi_{\text{avg}}$, matches the number of functional groups identified in the literature. I find that using alternative metrics, such as k-means or FCM objective functions, to identify the appropriate number of clusters for decomposition resulted in the selection of 9 functional groups as the most appropriate, which to my knowledge has no biological significance.

In the next step, the dissimilarity matrix $D$ is generated as explained in section 3.1.4. I apply the RFCM algorithm to $D$ to obtain the final hierarchical decomposition. I show the primary relationships and the strongest higher order relationships in my results, using $\rho$ to decrease the visualization of spurious weak relationships. The level of the threshold was set, as explained in Section 3.1.4, to $\rho = \mu(d_k)$, where $d_k$ refers to the set of memberships of component $k$, and $\mu(\cdot)$ stands for the mean.

These results are shown in Fig. 4.4. The primary and secondary relationships obtained are represented by a black and grey color, respectively. The vertical axes corresponds to the
Figure 4.4: Primary and secondary memberships of the components in the EGF model for $\rho = \mu(d_k)$. Columns represent components, and rows represent clusters.
Figure 4.5: Decomposition of the EGF induced MAPK [74] in primary and secondary relationships. Decomposition corresponding to the clustering in Fig. 4.4.

components of the system. The horizontal axes corresponds to the 6 different clusters.

Fig. 4.5 gives a pathway specific description of the decomposition results shown in Fig. 4.4, where each component has been shaded with the color of its primary and/or secondary functional group. When only the primary membership is shown for a component, a shaded rectangle shows the color of its primary cluster. If the primary and secondary clusters are shown, the largest portion of the rectangle shows the primary cluster, whereas the smaller portion of the rectangle indicates the secondary cluster.

An analysis of Fig. 4.4 and Fig. 4.5 explains the inconsistencies between previous results in the literature, and illustrates the agreement of my results with the equations from the original model [74]. First, components 22 (Grb2), 24 (Sos), and 30(Grb2-Sos) are related to both, the *Shc independent signaling complex formation* and the *Shc dependent signaling complex formation* modules. Previous methods however, had found relationships to only one of these modules. My results uncover the connections between those components and both of the functional units. In my results (see Fig. 4.5), the cited components are primarily clustered in the *Shc independent signaling complex formation* unit, and secondarily related to
Shc dependent signaling complex formation (since all the components in Shc dependent signaling complex formation are secondary related to components in Shc independent signaling complex formation).

The link between components from modules Shc independent Ras activation and Shc independent signaling complex formation is also revealed in my results. As I stated, previous literature had failed to find that most of the components in Shc independent Ras activation participate in reactions in Shc independent signaling complex formation. My results show secondary relationships among them. We can also look at Shc independent Ras activation and Shc dependent Ras activation. These two modules are clustered together in my results (in agreement with Conzelmann et al. [18]). This is produced because, as I explained, many of the components in these modules (36 (EGF-EGFR2-GAP-Shc-Grb2-Sos-Ras-GDP), 37 (EGF-EGFR*-2-GAP-Shc-Grb2-Sos-Ras-GTP), 28 (Ras-GTP), 26 (Ras-GDP), and 43 (Ras-GTP), are indeed located in both of them. Components in Shc dependent Ras activation however, are also largely involved in reactions from Shc dependent signaling complex formation, as uncovered by Ederer et al. [27]. The results in [27] and in [18], show only one of these connections of components in Shc dependent Ras activation (either the connection with module Shc independent Ras activation or Shc dependent signaling complex formation). I uncover both: the connections of components in Shc dependent Ras activation with those from Shc independent Ras activation with primary relationships (they are in the same module), and with those from Shc dependent signaling complex formation with secondary relationships.

Further I see modules for EGFR internalization and degradation and EGF binding are strongly correlated. This would also be in line with Ederer’s work, which placed module 4 and 5 in the same module. Finally, the MAPK cascade process is identified as a separate module, for its level of intramodular interactions is much lower than in the rest of modules.

As shown, the addition of the secondary relationships in this example leads to a more complete solution of the decomposition by uncovering connections that primary relationships only cannot find. This allows us to better identify key components to control the pathway, or better assess the impact of components in several functional units of the pathway.

I present three figures (Fig. 4.6, Fig. 4.7 and Fig. 4.8) to show the transient of the relationships between three different pairs of components that were previously analyzed. I define index $T_{i,j}$, that takes the value 1 if the components $i$ and $j$ are primarily related, 0.5 if they were secondarily related, and 0 otherwise. Each of the three figures shows index $T_{i,j}$ for the corresponding shown components and for each time point (in this example, there were 100 time points, and thus 100 clusterings). The figures illustrate how the relationships among components change due to the dynamics of the system.

Fig. 4.6 analyzes components 22 (Grb2) and 27 ((EGF-EGFR*)-2-GAP-Shc*-Grb2). As I explained, component 22 (Grb2), among others, should be related to both, components in
module 1 and components in module 4. My algorithm locates it primarily in module 4, in agreement with previous literature [27][18]. I can see in Fig. 4.6 that the algorithm found that component 33 ((EGF-EGFR*)2-GAP-Shc*-Grb2) (in cluster 1) was not related to component 22 (Grb2) at the beginning of the integration time, but they were secondarily related through the middle points, and they were primarily related at the end of the integration time, indicating that they were strongly interacting at this stage of the reaction. This is also the case for other components in cluster 1. These individual decompositions allowed my algorithm to find a
global secondary relationship among them. This led to a result that, unlike previous methods, found the connections of components 22 (Grb2) (and others) to both of the modules. A similar analysis can be extracted from figures 4.7 and 4.8. In Fig. 4.7, components 22 (Grb2) and 27 ((EGF-EGFR*)2-GAP-Grb2-Sos-Ras-GDP), located primarily in modules 4 and 5 respectively, are shown, whereas Fig. 4.8 deals with components 34 ((EGF-EGFR*)2-GAP-Shc*-Grb2) and 43 (Ras-GTP*), located primarily in modules 1 and 5 respectively.

These results emphasize the need for using the individual clusterings at each time point to uncover secondary relationships. Not taking into consideration the partial results at each time point would mask the secondary results and lead to primary relationships only that would not reveal all the information provided above.

To assess the robustness of the algorithm, I obtained results for 30 independent runs. My algorithm was able to consistently identify primary (100% of the runs), and secondary (80% of the runs) relationships that existed within this pathway. The 6 runs that showed slightly different patterns of secondary relationships however, did not lead to different conclusions. The variation of the clustering resulted in an alternate path, where components from the *Shc dependent signaling complex formation* and from the *Shc independent signaling complex formation* presented secondary relationships in the *Ras activation* module. In this alternative solution, both modules were related through secondary relationships by a different pattern. This is the same conclusion that was reached using the illustrated results, where these two modules were related via secondary relationships. These results emphasize the robustness of my algorithm, in spite of the randomness typically associated to clustering methods.

The complexity of the algorithm was also measured for this application in terms of execution
time. The `tic` and `toc` built-in functions in Matlab were used to measure the execution time, indicating that the algorithm needed 154.0237s to achieve the results.

### 4.2 C3 Photosynthesis

In this section, I apply both of the algorithms described in the previous chapter to the C3 photosynthesis model available in the BioModels database [50] and developed by Laisk *et al.* in [45] and [46]. It consists of 42 state variables, each representing the concentration of a component in the system. The model includes the Calvin Cycle, as well as the Starch synthesis, and Sucrose synthesis. These three modules, as identified in the literature, have been marked in Fig. 4.15 with three large rectangles that enclose the corresponding compounds. The trajectories of the 42 components of the model for typical ICs are shown in Fig. 4.9.

To the best of my knowledge, prior studies have not attempted to apply a decomposition algorithm to the C3 Photosynthesis model. This model however, has been extensively studied, becoming the most studied process in plant biology. The amount of information available about photosynthesis makes it an excellent choice to test the algorithm. Particularly, the distinct functional units have been clearly defined in multiple studies [45][46][67][89], and enzymes from the model that impact the photosynthetic rate have also been identified [89] [82].

#### Decomposition of the C3 Photosynthesis

I apply the algorithm as it has been explained in Section 4.1.2. The difference with respect to the previous example resides in the values of some of the parameters. The data is discretized in the default number of levels, $m = 2$. I evaluate the Silhouette index for clusters ranging from 1 to 5 for each time value $i_i$ and identify the value $\xi^{opt}_i$ that maximizes the Silhouette index. Attempts to decompose the model into greater than 5 clusters resulted in empty clusters and hence were not used in the evaluation of the Silhouette index. The evolution of the Silhouette index is shown in Fig. 4.10. I find that the average of all values $\xi^{opt}_i$ over time, $\xi_{avg} = 3$, matches the number of functional groups identified in the literature.

Module 1 encompassed mainly the components involved in the Calvin Cycle and Starch Synthesis, module 2 assembled the energy equivalent intermediates that are involved in several steps of the Calvin Cycle and in the Starch Synthesis, and module 3 contained the components from the Sucrose Synthesis. I show in Fig. 4.15 my decomposition in primary and secondary relationships. The parameter $\rho$ was set to $\rho = \frac{1}{2} \mu(d_k)$, where $d_k$ refers to the set of memberships of component $k$, and $\mu(\cdot)$ stands for the mean. The level of this threshold was heuristically modified, since the original value ($\rho = \mu(d_k)$) did not result in enough secondary
Figure 4.9: Simulation of the C3 Photosynthesis model by Laisk et al. in [45][46] run in XPP. ICs of the trajectories provided in BioModels [50].

relationships to convey any conclusion. The relationships embedded in this data thus, were less pronounced and required a different threshold. This results in all the primary relationships and the strongest secondary connections, which are shown in Fig. 4.11. The primary and secondary relationships obtained are represented by a black and grey color, respectively. The vertical axes corresponds to the components of the system. The horizontal axes corresponds to the 6 different clusters. Fig. 4.15 gives a pathway specific description of the decomposition results shown in Fig. 4.11, where, as before, each component has been shaded with the color of its
An analysis of the equations of the model by Laisk et al. [50],[45],[46] and the model by Zhu et al. [89] indicates that the identified secondary relationships coincide with components that are involved in several of the functional units that I identified. Components F6P, G6P, and G1P (all of them from the stroma) vary according to reactions 6, 7, and 23. The two first reactions are located in the Calvin Cycle module, whereas reaction 23 occurs in the Starch synthesis module. These three components, which directly interact with both modules, were located in both modules by my algorithm, with primary and secondary relationships. Components S7P and Pi (from the cytosol), although not directly involved with the Starch synthesis, react with Pi (from the stroma), which is located in the Starch synthesis module. This fact provides both of these components with a primary membership in the Calvin Cycle, and a secondary membership in the Starch synthesis module. This secondary membership represents a strong interaction of S7P and Pi with one of the Starch synthesis components. Components Xu5P and Ri5P (both from the Stroma) participate in reaction 7, which occurs in the Calvin Cycle, but follows the Starch Synthesis in the cycle—and thus involves components from the Starch synthesis. Both are successfully located in both modules with primary and secondary relationships. Finally, component Ru5P—located in the Calvin Cycle—reacts with ATP, ADP, and Pi—all of them contained in the Starch synthesis module—in reaction 13. My algorithm captures this strong interaction with the Starch synthesis by granting a secondary membership to Ru5P in the Starch synthesis module. The algorithm thus, achieves a decomposition that can capture interactions that primary relationships alone are not able to do.

To assess the robustness of the algorithm, I obtained results for 30 independent runs. I was able to identify consistently primary (100%) and secondary (100%) relationships that existed within this pathway.

The complexity of the algorithm was also measured for this application in terms of execution...
Figure 4.11: Primary and secondary memberships of the components in the C3 photosynthesis model for $\rho = \frac{1}{2} \mu(d_k)$. Columns represent components, and rows represent clusters.
Figure 4.12: Decomposition of the photosynthesis model. Picture taken from [89] (reduced). The numbers in the figure identify each reaction. The metabolites in different compartments, i.e. cytosol and stroma, are represented separately in the model. The shaded boxes indicate the my decomposition. Decomposition corresponding to the clustering in Fig. 4.11.
time. The *tic* and *toc* built in functions in Matlab were used to measure the execution time, indicating that the algorithm needed 28.1414s to obtain the results, which is a considerable decrease of the needed time with respect to the previous application.

### 4.2.1 SA/Black-Box Modeling Application

In this section, I apply the SA/Black-Box algorithm to the C3 photosynthesis model. In the previous section, I showed how the decomposition algorithm broke this model down into functional units. Here, the SA/Black-Box algorithm identifies which are the most important components in the model to manipulate or control the photosynthesis. This identification of components shows, in turn, that the decomposition in functional units can also help to elucidate which are the most important components—by association with the most impacting functional units.

Large amounts of information are available, not only about the model, but also about the phenotype that I picked. The photosynthetic rate has been deeply studied, and thus, mathematical models exist that relate it with some components such as CO$_2$—in spite of the difficulties associated to generating mathematical models of some phenotypes as a function of low level biological models. This situation contributes to accredit this phenotype as my testing application. I do not make use of these models, and instead, I use this information [89] as a proof of concept to assess the validity of my approach. In most cases that a phenotype has to be studied, a model like this one will not exist, and this is where my method will become advantageous. I show that, without the use of these equations, I reach similar conclusions to those who did use them [89], and to those who made their conclusions from experimental results.

**Previous Analysis of the C3 Photosynthesis Model**

I have used results from previous studies [89][82] to validate my own conclusions. Zhu et al. [89] studied which enzymes needed to be varied to increase photosynthesis. They found that relative to the initial concentration of enzymes, increases in Rubisco, FBPase, SBPase and ADP Glucose Pyrophosphorylase (ADPGPP) were required to increase the photosynthetic rate. The photosynthetic rate thus is sensitive to the variation of these enzymes and to the components that are catalyzed by these enzymes. I will show that my algorithm identifies these components as the most impacting to vary of the photosynthetic rate.

Tamoi et al. [82] generated transgenic tobacco plants expressing cyanobacterial FBPase-II or Chlamydomonas SBPase in chloroplasts to clarify the contribution of FBPase and/or SBPase to CO$_2$ fixation, RuBP regeneration and biosynthesis of photosynthates. They found that the photosynthetic activities were increased for sufficiently large increased levels of FBPase and SBPase in leaves, compared with that of the wild-type plants. The enhancement of either more
than a 1.7-fold increase of FBPase or a 1.3-fold increase of SBPase in the chloroplasts resulted in an increase of the photosynthetic rate in the chloroplasts of the transgenic plants.

Poolman et al. [67] use results from previous studies and MCA—as defined in 2.15—to conclude that Rubisco and SBPase are the only enzymes that have a significant influence over the flux of CO$_2$ assimilation. The MCA method, although not purely defined as a sensitivity analysis, performs an analysis of the control that each enzyme performs on a flux. As other techniques currently in the literature, this technique is typically used in applications where a mechanistic model exists.

**Our Results**

I pursued the identification of the impact factor of each of the components of the model on the photosynthetic rate. For this, I applied the algorithm described in Section 3.2.2:

1. **The data:** I leveraged the real measurements contained in the Boreas te-09 photosynthetic response data [22] to apply my method. This data was used in the next step to estimate the parameters of the HMM. The Boreas te-09 photosynthetic response data contains photosynthetic response of boreal tree species to varied parameters, such as light, temperature, and CO$_2$ concentration. Only one parameter was varied in each data point. I was interested in obtaining a set of data where measurements of the photosynthetic rate had been obtained as a response to a variation of the concentration of some component involved in the C3 photosynthesis pathway, while the remaining parameters were not varied—recall that we need a sequence of observations or component values, and a sequence of states or phenotypical values, to generate the HMM. I thus, picked a subset of this data consisting of the measurements where CO$_2$ is the parameter varied, for this is a component involved in the photosynthesis pathway. This lead to a total of 60 data points. Table 4.1 describes the 20 fields contained in each point in the data set. I make use, exclusively, of the fields “photosynthetic rate” and “intercellular CO$_2$ concentration”, which become the state and the observation measurements, respectively.

2. **HMM generation:** I generated an HMM based on the data previously described, i.e., I fit the parameters of the HMM to a set of experimental measurements of the photosynthetic rate as a response to a variation of CO$_2$. First, I needed to determine the number of discretization levels of the data. Recall that the appropriate number of the discretization levels depends on the number of data points that are available, and the level of detail of the prediction needed. A finer discretization of the states allows more level of detail in the future predicted hidden states. The number of parameters that we need to fit however, rises with the number of states and observation symbols, and so does the amount of data.
Table 4.1: Description of the data point in the Boreas te-09 photosynthetic response data [22].

<table>
<thead>
<tr>
<th>FIELD</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SITE_NAME</td>
<td>The identifier assigned to the site by BOREAS, in the format SSS-TTT-CCCCC, where SSS identifies the portion of the study area: NSA, SSA, REG, TRN, and TTT identifies the cover type for the site, 999 if unknown, and CCCCC is the identifier for site, exactly what it means will vary with site type.</td>
</tr>
<tr>
<td>SUB_SITE</td>
<td>The identifier assigned to the sub-site by BOREAS, in the format GGGGG-IIIII, where GGGGG is the group associated with the sub-site instrument, e.g. HYD06 or STAFF, and IIIII is the identifier for sub-site, often this will refer to an instrument.</td>
</tr>
<tr>
<td>START_DATE</td>
<td>Starting date of the referenced data collection.</td>
</tr>
<tr>
<td>END_DATE</td>
<td>Termination date of the referenced data collection.</td>
</tr>
<tr>
<td>SPECIES</td>
<td>Botanical name of the species (Genus species).</td>
</tr>
<tr>
<td>PARAM_VARIED</td>
<td>The parameter varied to study photosynthetic response to BOREAL tree species. i.e. CO2 CONCENTRATION=CO2 varied; LIGHT=light varied; TEMPERATURE=temperature varied; VAPOR PRESS DEFICIT = vapor pressure deficit varied; WATER POTENTIAL = Water potential varied; DARK RESPIRATION=temperature varied in dark; HUMIDITY=humidity varied; CONSTANT CONDITIONS=nothing varied; STOMATAL MODEL=many things varied for model calibration.</td>
</tr>
<tr>
<td>LEAF_TEMP</td>
<td>The measured leaf or shoot temperature</td>
</tr>
<tr>
<td>AIR_TEMP</td>
<td>The measured air temperature.</td>
</tr>
<tr>
<td>CO2_CONC</td>
<td>CO2 concentration.</td>
</tr>
<tr>
<td>TRANSPIRATION_RATE</td>
<td>Transpiration rate (E)</td>
</tr>
<tr>
<td>PHOTOSYNTHETIC_RATE</td>
<td>Measured Net Photosynthesis</td>
</tr>
<tr>
<td>DOWN_PPFD</td>
<td>The downward photosynthetic photon flux density.</td>
</tr>
<tr>
<td>INTERCELL_CO2_CONC</td>
<td>Intercellular CO2 concentration</td>
</tr>
<tr>
<td>STOMATAL_CONDUCT.CO2</td>
<td>Stomatal conductance to CO2 (gs)</td>
</tr>
<tr>
<td>WATER_USE_EFF</td>
<td>Water use efficiency</td>
</tr>
<tr>
<td>WATER_POTENTIAL</td>
<td>Water Potential</td>
</tr>
<tr>
<td>VAPOR_PRESS_DEFICIT</td>
<td>Vapor Pressure Deficit (VPD)</td>
</tr>
<tr>
<td>VAPOR_PRESS_DEFICIT.AIR_TEMP</td>
<td>Air temperature when vapor pressure deficit varied.</td>
</tr>
<tr>
<td>CRTFCN_CODE</td>
<td>The BOREAS certification level of the data. Examples are CPI (Checked by PI), CGR (Certified by Group), and PRE (Preliminary).</td>
</tr>
<tr>
<td>REVISION_DATE</td>
<td>The most recent date when the information in the referenced data base table record was revised.</td>
</tr>
</tbody>
</table>
Figure 4.13: Measured photosynthetic rate, predicted photosynthetic rate, and measured CO₂. Amplitudes presented to scale of the discretization level (not real values in \([\text{millimoles/}\text{meter}^{-2}\times\text{second}^{-1}]\) and \([\text{parts per million}]\)).

needed. In this case of study, I had 60 data points available. The number of discretization levels was determined experimentally, by building multiple HMM configurations, and by choosing the configuration that minimized the error of the prediction. I particularly built multiple HMM configurations by varying the number of discretization levels of the data and estimating the HMM to fit this data, which lead to HMMs with different number of states and observation symbols. The minimum prediction error found was 15%, and it was given by a configuration with 3 states, and 20 observation symbols. Fig. 4.13 shows, for this configuration of the HMM, the measured photosynthetic rate versus the predicted photosynthetic rate for the 60 points sequence of CO₂. As a result, I obtained a state transition probability matrix, \(A\), and a observation probability matrix, \(B\), of sizes \(3 \times 3\) and \(3 \times 20\), respectively. The parameters of the HMM, i.e., the entries of these matrices, were determined in Matlab, as explained in Section 3.2.2. The HMM generated was used in the later steps to predict the values of the state (photosynthesis) given new observation sequences (CO₂ concentration values, varying in time). As it was expressed in previous section, I refer as \(H\) to the generated HMM.
3. **Prediction of typical photosynthetic values for this model:** In this step, I predicted the photosynthesis under typical conditions of the model. I first ran my model under typical ICs of all the components, $x(0)^{typ}$, which lead to typical trajectories, $\xi^{typ}(t, x(0)^{typ})$. Component CO$_2$ is the observation component in this application. Its typical trajectory thus, $\xi_{CO_2}^{typ}(t, x(0)^{typ})$, corresponds to the typical observation trajectory, $\xi_{obs}^{typ}(t, x(0)^{typ})$. I discretized the typical observation trajectory, $\xi_{obs}^{typ}(t, x(0)^{typ})$, in 20 levels, and used it along with the trained HMM, $H$, to predict the typical photosynthesis rate vector, $P^{typ}$. As mentioned, I performed this step in Matlab, which uses the Viterbi algorithm to calculate the most likely sequence for $P^{typ}$. Fig. 4.14 shows this prediction along with $\xi_{obs}^{typ}(t, x(0)^{typ})$.

4. **Prediction of photosynthetic values for perturbed conditions of the model:** Here, I predicted the photosynthesis under perturbed conditions of the model. This step was performed as follows:

- The ICs of each of the components were varied one at a time, with a percentage
of variation of a 10% of the original value. This value was chosen heuristically, as explained in Section 3.2.2. The value was chosen because perturbations typical of classical SA methods (0.1% to 1%) did not result in meaningful changes of the phenotype. Two components of the system had zero ICs. These components needed to be perturbed with a value different than zero. The perturbation was picked to be $x_k(0)_{\text{perturbed}} = 10^{-7}$. This value was also chosen heuristically from within a range of values that agreed with the 10% of the smallest ICs in the system. The perturbed ICs thus were defined as:

$$
\begin{align*}
\text{If } x_k(0) = \alpha_k, \text{ for } \alpha_k \neq 0, & \text{ then } x_k(0)_{\text{perturbed}} = \alpha_k + 10\% \cdot \alpha_k & (4.1) \\
\text{If } x_k(0) = \alpha_k, \text{ for } \alpha_k = 0, & \text{ then } x_k(0)_{\text{perturbed}} = 10^{-7} & (4.2)
\end{align*}
$$

- I ran the model after each perturbation, and $\xi_{\text{obs}}^k(t, x_k(0)_{\text{perturbed}})$ (the CO$_2$ trajectories) were stored.
- $\xi_{\text{obs}}^k(t, x_k(0)_{\text{perturbed}})$ and the trained HMM, $H$, were used to predict the photosynthetic rate that each perturbation produced, $P^k_{\text{per}}$, where index $k$ denotes that the photosynthetic trajectory was produced when component $x_k$ was perturbed.

5. **Calculation of the impact factors**: The sensitivity coefficients, or impact factors, were finally computed as indicated in 3.32. Fig. 4.15 shows the normalized impact factors obtained for each component, where the normalization was performed with respect to the largest impact factor found in the model.

The results in Fig. 4.15 show how my algorithm identified components CO$_2$, Ru5P, DHAP, EPP, F6P, SBP, ER, EPG, TP, ADP, and EP, in the given order, as the most impacting components (results shown in table 4.2). CO$_2$ however, reached a normalized impact factor of three orders of magnitude larger than the rest of the components provided in the list—all of them of the same order of magnitude. This outcome is in accordance with the fact that CO$_2$ is the most important internal controller of photosynthesis—this is proven by the fact that photosynthesis models express a direct dependence on internal CO$_2$ [89][49]. Other components that were found to have some capacity to influence photosynthesis were PGA, cytosolic GAP, cytosolic G1P, cytosolic FBP, RUBP, S7P, and Pi. These components had normalized impact factors one order of magnitude smaller than the first set of components listed. I thus, focus my analysis on the set of the most impacting components (CO$_2$, Ru5P, DHAP, EPP, F6P, SBP, ER, EPG, TP, ADP, and EP). Recall that the enzymes Rubisco, FBPase, SBPase and ADPGPP had been identified in the literature as being required to increase the photosynthetic rate. The components that I identified in my method are involved in reactions catalyzed by these enzymes.
Figure 4.15: Photosynthesis model. Picture taken from [89] (reduced). The impact factors of each of the components on the photosynthetic rate are represented by numbers in brackets. The most impacting factors component have been circled. The metabolites in different compartments, i.e. cytosol and stroma, are represented separately in the model.
Table 4.2: Components with highest impact factors.

<table>
<thead>
<tr>
<th>COMPONENT</th>
<th>IMPACT FACTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO₂</td>
<td>1</td>
</tr>
<tr>
<td>Ru₅P</td>
<td>0.0087</td>
</tr>
<tr>
<td>DHAP</td>
<td>0.0029</td>
</tr>
<tr>
<td>EPP</td>
<td>0.0027</td>
</tr>
<tr>
<td>F₆P</td>
<td>0.0025</td>
</tr>
<tr>
<td>SBP</td>
<td>0.0024</td>
</tr>
<tr>
<td>ER</td>
<td>0.0024</td>
</tr>
<tr>
<td>EPG</td>
<td>0.0023</td>
</tr>
<tr>
<td>TP</td>
<td>0.0015</td>
</tr>
<tr>
<td>ADP</td>
<td>0.0013</td>
</tr>
<tr>
<td>EP</td>
<td>0.0010</td>
</tr>
</tbody>
</table>

Components CO₂, EPG, EPP, ER, EP are associated to Rubisco, which controls the reaction in which RuBP is combined with CO₂ to form PGA. Ru₅P, although not directly associated with Rubisco, plays an important role in this enzyme’s reaction, for it is converted to RuBP by the enzyme Rpi. DHAP is a primary component in the synthesis of both, FBP and SBP, which are the substrates of the reactions that FBPase and SBPase catalyze, respectively. Finally, F₆P is the product of the reaction that FBPase catalyzes. My results thus, are in strong agreement with Zhu et al. [89], Poolman et al. [67] and Tamoï et al. [82], since most of the components that I found as the most impacting are associated with the enzymes previously found in the literature. The exceptions to this agreement are ADP and TP, which are the only components identified as being highly impacting that are not strongly related to any of the enzymes previously identified in the literature. Both, ADP and TP, however, are components involved in a large number of reactions. ADP is involved in reactions catalyzed by the enzymes pgk, phosphoribulokinase, and starch synthase, and TP is involved in reactions catalyzed by the enzymes FBP Aldolase, PGA Kinase, F₆P Transketolase, SBP Aldolase, S₇P Transketolase, and TPT. The large number of reactions in which these components are involved may have caused the assignment of high impact factors. This identification can serve to indicate that these components could be further tested in future experiments. Also, all the highly impacting components identified belong to the Calvin Cycle, with the exception of ADP. This component is classified by the decomposition step in the Starch synthesis module. The Starch Synthesis module and components from the Calvin Cycle, however, are secondarily related, and ADP, indeed, participates in reactions from both modules. Also, ADP, although not directly involved with the enzyme ADPGPP, participates in multiple reactions with ATP, Pi, and ADPG, which are the components that participate in
the reaction catalyzed by ADPGPP—last enzyme identified by Zhu et al. This may have been another factor that lead to find a relatively high impact factor on ADP.

I thus, clearly identify three enzymes (Rubisco, FBPase, SBPase) out of the four that were found in [89]. These finings occur in the context of a model [45, 46] where there are a total of 27 reactions catalyzed by 22 different enzymes. This numbers highlight again the potential of my technique and the methodologies that can follow my work.

The results show as well the agreement of impact factors and modules. This is more clear if we look at the control factor of each module:

\[ \rho_1 = 1.0000, \quad \rho_2 = 0.0026, \quad \rho_3 = 0.0066. \]  

(4.3)

where \( \rho_i \), for \( i = 1, 2, 3 \) is defined as the average of the impact factors of all the components in module \( i \), normalized with respect to the largest \( \rho_i \). These indices show the fact that as a group, some modules have higher impact factors than others. The control factors also highlight the dispensability of each functional unit for constructing multihierarchical models that focus
on the study a phenotype.

The complexity of the algorithm was also measured for this application in terms of execution time. The `tic` and `toc` built in functions in Matlab were used to measure the execution time, indicating that the algorithm needed 11.3066s to obtain the results.

4.3 Chapter Summary

In this chapter, I have applied the algorithms presented in Chapter 3 to two different models. First, the decomposition algorithm was applied to the EGF induced MAPK of Schoeberl et al. I showed that I found 1) primary relationships among components that are in agreement with previous computational decomposition studies, and 2) secondary relationships that make the solution more complete, by providing information about relationships that are present in the original equations, but that do not appear as primary relationships.

I showed that some contradictory classifications were achieved in prior studies. I analyzed that the reason for this phenomena is that there are some ambiguities in the components classifications. I demonstrated that I can capture these situations by presenting my results as primary and strongest higher order relationships. The strongest higher order relationships can uncover situations where components interact with several modules, and they should indeed appear in all of them.

I finally presented three figures that captured the variation over time of the relationship between three pairs of components. This indicated that my algorithm can capture the transient of component relationships in the network, due to the dynamic nature of the reactions. This supports and validates my method—it shows that it is essential to take into consideration the clustering at each individual time point to reveal secondary relationships.

The final results show primary and strongest higher order relationships, which along with the transient relationships, can help us elucidate hierarchical control, giving strategic insight into components that are critical for controlling the pathway.

In second place, I applied both, the decomposition and the sensitivity analysis algorithms to the photosynthesis pathway developed by [45]. I showed that the decomposition algorithm successfully separated the model in three functional units. I showed that the method could find secondary relationships of components that were involved in two functional units, and that primary relationships only could not reveal.

The SA algorithm afterwards, found which were the components with highest impact coefficient with regards to controlling the photosynthetic rate. Previous studies had found important enzymes to increase photosynthesis either through experimental methods, or by modeling the relationship between the photosynthesis and the pathway. My results are in strong agreement with the literature. I identified components related to Rubisco, FBPase, and SBPase as the
most important to control the photosynthesis rate. My method however, reached these results without requiring a mathematical relationship between the model and the phenotype, for it is not available in most cases of study.

The application of both of my algorithms to the C3 photosynthesis model allowed us to show that the decomposition algorithm is an important tool towards achieving multihierarchical models. I showed that I could identify functional units that could be discarded by a reduction algorithm, while other units were necessary to control or manipulate the photosynthetic rate.
Chapter 5

Conclusions and Future Work

5.1 Conclusions

I have presented in the previous chapters a framework that encompasses two computational strategies that will lay the ground towards multiscale modeling. The first one is a model analysis and decomposition technique. I claim that my decomposition method, which performs an analysis of primary and secondary relationships among components in a model, helps to reveal the underlying functional structure of the models. The second technique that I present is an identification of groups of components and parameters of a model that are relevant to a higher level of abstraction. To my knowledge, this is the first attempt to solve this problem. I argue that the combination of these techniques can be an important tool to link several models from distinct levels of abstraction, saving the bridge between current models and multiscale models.

Merging the decomposition and the SA/black-box approaches allows us to perform a study of the sensitivity of a phenotype to the variation of component concentrations from a biological process. Sensitivity analysis methods require the existence of a description of the system under study, which is not typically available across levels of abstraction. In my method, a black-box models the relationship between the process and the phenotype, which provides the needed bridge to perform an analysis of the effects of component perturbations on the phenotype. Next, the decomposition algorithm identifies related components prior the application of the local SA method. This solution grants us the identification of, not only individual components, but also functional units that impact the phenotype.

I studied the application of my methods to two different biological processes. The decomposition method was applied to the EGF Induced MAPK model. Previous methods had decomposed this model in functional units, achieving contradictory results. I showed that my algorithm could clarify these contradictions by performing a decomposition, not only of pri-
mary relationships, but also of higher order relationships. The algorithm uncovered that some components interacted with more than one functional unit, which had resulted in different decompositions when studied by different methods. My algorithm lead to a more complete solution of the EGF decomposition than previous algorithms, for it captured the information of both of the previous decompositions. The ability of my algorithm to find secondary relationships can have an impact in identifying groups of components that can control a phenotype, and in the generation of multilevel models. When the decomposition algorithm precedes the SA algorithm in a model application, it adds to the approach the capacity of knowing which components are secondarily related to the most impacting functional units. These components, then, could be targeted by experimental results along with the primary members of these impacting units.

The application of both, the decomposition and the sensitivity analysis algorithms, was tested on the C3 photosynthesis model. The decomposition algorithm successfully separated the model in three functional units. I showed that the method could find secondary relationships of components that were involved in two functional units, and that primary relationships only could not reveal. The SA algorithm identified the components that impact in a greater factor the photosynthetic rate. I validated my results by comparison with previous studies, and I showed that my results are in strong agreement with them. My methods, however, unlike previous research, reached these results without requiring a mathematical relationship between the model and the phenotype—for it is not available in most cases of study—, and without the need of performing experiments to modify and measure each of the components of the model.

I stated at the beginning of this document the importance of moving towards multihierarchical models to explain, describe, and control complex biological phenomena. I argued as well about the difficulties associated with current modeling techniques to achieve multilevel models. One of the biggest adversities is the high complexity of biological models, which hinders the combination of multiple models at a multilevel scale. As stated by Dada & Mendes in [21], “there is a need for automated methods to help in analysing a complex model defined at a particular spatial and temporal scale in order to compute the parameters of a simpler model that captures the model behaviour relevant to the scales above”. It is necessary, thus, to develop reduction techniques capable of keeping the necessary information “relevant to the scales above”, or, in other words, reduction techniques should achieve “phenotype-dependent” reduced models.

I claim that the findings of my algorithms can provide the necessary preceding information prior application of reduction algorithms in the context that I just presented. I identified which components and functional units from a model were or were not relevant to a phenotype, thus this information can indicate which functional units and groups of components need to be discarded or highlighted by reduction techniques. Quoting, again, Dada & Mendes [21], “it is only by combining multi-scale modeling strategies with advances in computational technology that
the aims of systems biology can fully be achieved”, and my proposed computational framework is a big step in this direction.

5.2 Future Work

The work presented in this document can be expanded in several directions.

5.2.1 Small Scale

1. Reduction:
   
   As I stated, the approach above presented lays the ground for the development of “phenotype dependent” reduction techniques. Reduction techniques based on the results provided by my methods, thus, are the next natural step of my work. Reduction techniques for dynamic biological systems have been developed in the past. They, however, perform classical reductions that do not consider the influence of other levels of abstraction. Adding a “phenotype-dependent” reduction technique to my work, and applying it to a meaningful large complex model would better demonstrate the potential of these approaches.

5.2.2 Large Scale

1. Multihierarchical Modeling:
   
   Our work can have a more direct impact on multilevel modeling by expanding the type of applications that were shown in this document. My framework could be used to relate two models that reside in two different levels of abstraction, rather than a model and a phenotype. The methods could find which groups of components from one model impact another process that occurs in a higher level. In this scenario, my methods could serve as an initial tool to concatenate models across biological levels of operation.

2. Synthetic Biology:
   
   The generation of multihierarchical models has direct applications on synthetic biology. This area of research aims to design biological systems, and it is closely related to the area of systems biology. Techniques that facilitate the understanding of living organisms across levels of abstraction have an obvious repercussion on the design of new systems. The multihierarchical understanding of living organisms has been identified as a need to continue the progress in the synthetic biology area. Applications of these algorithms to meaningful problems that concern synthetic biology, and collaborations with synthetic
biology researchers that can lead to the development of new methods that satisfy their needs, is an exciting perspective for expanding the presented research.

5.3 Publications

The work presented in this thesis lead to two journal papers:


In addition, the following conference papers resulted from related research done during my Ph.D. program:


REFERENCES


[82] Masahiro Tamoi, Miki Nagaoka, Yoshiko Miyagawa, and Shigeru Shigeoka. Contribution of fructose-1, 6-bisphosphatase and sedoheptulose-1, 7-bisphosphatase to the photosynthetic


Appendix A

Robustness of the Algorithms

Robustness studies of the decomposition and the model to phenotype algorithms were performed under the variation of some of their critical parameters. The robustness studies were evaluated for both of the applications presented in this document.

A.1 Decomposition Algorithm

In Chapter 4, I presented the results of a robustness study of the decomposition algorithm for invariant parameters (i.e., I obtained results for independent runs of the algorithm without varying any of the parameters).

For the EGF Induced MAPK case of study, I obtained results for 30 independent runs of the algorithm, as explained in Chapter 4. My algorithm was able to consistently identify primary (100% of the runs), and secondary (80% of the runs) relationships that existed within this pathway. The 6 runs that showed slightly different patterns of secondary relationships however, did not lead to different conclusions. The variation of the clustering resulted in an alternate path, where components from the Shc dependent signaling complex formation and from the Shc independent signaling complex formation presented secondary relationships in the Ras activation module. In this alternative solution, both modules were related through secondary relationships by a different pattern. This is the same conclusion that was reached using the illustrated results, where these two modules were related via secondary relationships. These results emphasize the the robustness of my algorithm, in spite of the randomness typically associated to clustering methods.

For the C3 Photosynthesis case of study, I obtained results for 30 independent runs as well. I was able to identify consistently primary (100%) and secondary (100%) relationships that existed within this pathway.

In addition to these tests, I present in this section a robustness study of the decomposition
algorithm under the variation of 1) the number of the activity discretization levels, \( m \), and 2) the value of the activity discretization levels, \( a_k(t) \), for a fixed number of levels, \( m \), and for both cases of study.

A.1.1 Robustness Under Variations of the Activity Discretization Levels

EGF Induced MAPK

The number of discretization levels, \( m \), used in the results shown in Chapter 4, was \( m = 3 \). I present in this section the variability of the results when \( m \) takes values \( m = 5, 9 \). Recall that smaller values of \( m \) in this application (i.e., \( m = 2 \)), results in empty clusters, thus this is not a valid number of discretization levels for this application.

- \( m = 5 \): I collected results for 10 runs of the algorithm for \( m = 5 \). I obtained a consistency of 80\% of the primary and secondary relationships across the 10 runs, i.e., 8 runs achieved the same decomposition with variations of at most 2\% among them. This decomposition is shown in Fig. A.1, and differs from the one that was obtained for \( m = 3 \), indicating that some noise was introduced for this value of parameter \( m \). The remaining 20\% of the runs showed a different pattern, where 8\% of the primary relationships were different from that shown in Fig. A.1.

- \( m = 9 \): I collected results for 10 runs of the algorithm for \( m = 9 \). This resulted in decompositions that varied 5 - 10\% among runs, indicating a decrease in the consistency with respect to the previous cases of study. The decomposition of one of these runs is shown in Fig. A.2, where the results show a noisier pattern than the previous cases of study.

C3 Photosynthesis

The number of discretization levels, \( m \), used in the results shown in Chapter 4, was \( m = 2 \). I present in this section the variability of the results when \( m \) takes values \( m = 3, 9 \).

- \( m = 3 \): I collected results for 10 runs of the algorithm for \( m = 3 \). The results obtained in 5 of the runs are shown in Fig. A.3. I obtained two additional patterns of the decomposition, with differences of 7\% (in three of the runs) and 14\% (in two of the runs) in the primary relationships with respect to the decomposition shown in the figure. The results shown indicate an increase of noise with respect to the \( m = 2 \) case.

- \( m = 9 \): I collected results for 10 runs of the algorithm for \( m = 9 \). The results obtained in 7 of the runs are shown in Fig. A.4. I obtained one additional pattern of the decomposition,
Figure A.1: Primary and secondary memberships of the components in the EGF model for $m = 5$. Columns represent components, and rows represent clusters.
Figure A.2: Primary and secondary memberships of the components in the EGF model for $m = 9$. Columns represent components, and rows represent clusters.
Figure A.3: Primary and secondary memberships of the components in the C3 photosynthesis model for $m = 3$. Columns represent components, and rows represent clusters.
with differences of 21% in the primary relationships with respect to the decomposition shown in the figure. The results shown indicate an increase of noise with respect to the $m = 2$ and $m = 3$ cases.

As expected, we obtained increasing noise for increasing values of $m$. This fact does not imply a lack of robustness of the algorithm with respect to variations of $m$, however, for it was previously justified the need for selecting small values of this parameter to avoid computations with fast variations of the parameters.

A.1.2 Robustness Under Variations of the Activity Values for a Fixed Number of Levels

EGF Induced MAPK

I obtained the results of 10 runs of the algorithm, where an offset of 20% of the mean of the derivatives was added to the levels of discretization for generating the activity vectors. This resulted in the same decomposition that the one shown in Fig. 4.5 for 6 of the runs, while a slightly different pattern (10% of primary relationships were different), was obtained for the remaining 4 runs. These results indicate that the algorithm is robust against variations of the level of the activity values.

C3 Photosynthesis

Adding any significant offset (as small as 0.1% of the mean of the derivative values) to the levels of discretization for generating the activity vectors, resulted in empty clusters for this particular application. Although one may think that this may be a sign of lack of robustness, this result is in fact not an indication of the degree of robustness. This result instead, is caused by the proximity of the data used in this application, and conclusions about robustness can not be drawn in this particular case.

A.2 Model to Phenotype Algorithm

I obtained results for 30 independent runs of the algorithm. My algorithm was able to consistently identify the same impact factors in each run. This fact emphasizes the robustness of the algorithm.

The most critical parameter of the algorithm is the number of discretization levels of the phenotypical measurements and the component measurements. These parameters, however, were optimized before the execution of the algorithm, and changes in these values would lead
Figure A.4: Primary and secondary memberships of the components in the C3 photosynthesis model for $m = 9$. Columns represent components, and rows represent clusters.
to misleading results. Particularly, the number of discretization levels was determined exper-
imentally, by building multiple HMM configurations, and by choosing the configuration that
minimized the error of the prediction. Varying these parameters thus, increases the error of
the HMM prediction, which results in impact factors that badly calculated. Measuring the
robustness of the algorithm as a variation of these parameters is consequently inappropriate,
and in fact, their optimization must always be performed prior application of this algorithm,
as stated in Chapter 3.

I limit the robustness study of this algorithm thus, to the non-varying parameters case,
concluding that the model to phenotype method is robust in terms of consistency in its results.