Modeling and simulation of Hybrid Systems and Cell factory applications

Rodrigo Assar

N°d'ordre : 4335

THÈSE

PRÉSENTÉE À

L'UNIVERSITÉ BORDEAUX I

ÉCOLE DOCTORALE DE MATHÉMATIQUES ET D'INFORMATIQUE

Par **Rodrigo ASSAR CUEVAS**

POUR OBTENIR LE GRADE DE

DOCTEUR

SPÉCIALITÉ : **INFORMATIQUE**

Modeling and simulation of Hybrid Systems and Cell factory applications

Soutenue le : 21 Octobre 2011

Après avis des rapporteurs :

Jean-Michel Couvreur	Professeur
Servet Martínez	Director de Recherche

Devant la commission d'examen composée de :

Jean-Michel Couvreur	Professeur	Président de jury
Alain Griffault	Maître de Conférences	Rapporteur
Servet Martínez	Director de Recherche	Rapporteur
David J. Sherman ...	Professeur	Directeur de Thèse

– 2011 –

Acknowledgments

Firstly, I would like to thank my director of thesis, Professor David Sherman for their guidance, patience and outstanding support throughout my doctoral studies. I would also like to extend my gratitude to all the MAGNOME team of the LaBRI, who helped me to learn important lessons of life.

The INRIA is gratefully acknowledged for financially supporting my studies in France on a CORDI-S fellowship.

Thanks to all the jury: Servet Martínez and Jean-Michel Couvreur who reviewed my thesis, and Alain Griffault who gentily has taken part of the jury.

I want to finish by thanking to my family, my parents and Coté, who have supported and encouraged me throughout my PhD studies. I dedicate this thesis to them.

Modeling and simulation of Hybrid Systems and Cell factory applications

Abstract : The main aim of this thesis is to develop an approach that allows us to describe biological systems with theoretical sustenance and good results in practice. Biological functions are the result of the interaction of many processes, that connect different hierarchy levels going from macroscopic to microscopic level. Each process works in different way, with its own goal, complexity and hierarchy level. In addition, it is common to observe that changes in the conditions, such as nutrients or environment, modify the behavior of the systems. So, to describe the behavior of a biological system over time, it is convenient to combine different types of models: continuous models for gradual changes, discrete models for instantaneous changes, deterministic models for completely predictable behaviors, and stochastic or non-deterministic models to describe behaviors with imprecise or incomplete information. In this thesis we use the theory of Composition and Hybrid Systems as basis, and the BioRica framework as tool to model biological systems and analyze their emergent properties *in silico.*

With respect to Hybrid Systems, we considered continuous models given by sets of differential equations or more general dynamics. We used Stochastic Transition Systems to describe the dynamics of model changes, allowing *coefficient switches* that control the parameters of the continuous model, and *strong switches* that choose different models. Composition, reconciliation and reusing of models allow us to build complete and consistent descriptions of complex biological systems by combining them. Compositions of hybrid systems are hybrid systems, and the refinement of a model forming part of a composed system results in a refinement of the composed system. To implement our approach ideas we complemented the theory of our approach with the improving of the BioRica framework. We contributed to do that giving a BioRica specification of Hybrid Systems that assures integrity of models, allowing composition, reconciliation, and reuse of models with SBML specification.

We applied our approach to describe two systems: wine fermentation kinetics, and cell fate decisions leading to bone and fat formation. In the case of wine fermentation, we reused known models that describe the responses of yeasts cells to different temperatures, quantities of resources and toxins, and we reconciled these models choosing the model with best adjustment to experimental data depending on the initial conditions and fermentation variable. The resulting model can be applied to avoid process problems as stuck and sluggish fermentations. With respect to cell fate decisions the idea is very ambitious. By using accurate models to predict the bone and fat formation in response to activation of pathways such as the Wnt pathway, and changes of conditions affecting these functions such as increments in Homocysteine, one can analyze the responses to treatments for osteoporosis and other bone mass disorders. We think that here we are giving a first step to obtain *in silico* evaluations of medical treatments before testing them *in vitro* or *in vivo.*

Keywords : Biological systems, hybrid systems, composition, reusing and reconciling models, BioRica, wine fermentation kinetics, bone formation.

Discipline : Computer-Science

LaBRI
Université Bordeaux 1
351 cours de la Libération,
33405 Talence Cedex (FRANCE)

Résumé : Les Fonctions biologiques sont le résultat de l'interaction de beaucoup de processus, avec differents objectives, complexités, niveaux d'hiérarchie, et changements de conditions que modifient le comportement de systèmes. Nous utilisons des équations diferenciales ou dynamiques plus générales, et Stochastic Systèmes de Transition pour décrire la dynamique de changements des modèles. La composition, réconciliation et reutilisation des modèles nous permettent d'obtenir des descriptions de systèmes biologiques complètes et compatibles et leur combiner. Notre spécification de Systèmes Hybrides avec BioRica assures l'intégrité de modèles, et implement notre approche. Nous appliquons notre approche pour décrire in-silico deux systèmes: la dynamique de la fermentation du vin, et des décisions cellulaires associées à la formation de tissu d'os.

Mots clefs : Systèmes biologiques, Systèmes hybrides, composition, réconciliation et reutilisation des modèles, BioRica, dynamique de la fermentation du vin, formation de tissu d'os.

Discipline : Informatique

LaBRI
Université Bordeaux 1
351 cours de la Libération,
33405 Talence Cedex (FRANCE)

Contents

List of Figures

Chapter 1

Introduction

The aim of this chapter is to present the main elements for modeling biological systems. We summarize about the challenges to model this type of systems, the methodology and specifications. The main points that we discuss are the presence of modules and hierarchies, the associated numerical problems, the role of composition of models, and the consideration of randomness and hybrid systems to describe biological systems.

To describe the behavior of a biological system over time, it is convenient to combine different types of models: continuous models to describe gradual changes over time, discrete models for instantaneous changes, deterministic models to represent completely predictable behaviors and stochastic models to introduce randomness. Hybrid systems and Composition, as we define them here, allow us to combine all these elements.

1.1 Dynamical Models for Biological Systems

Since many years living organisms have been an object of study by natural science, which tries to understand the different mechanisms that are developed by organisms to live and to interact between themselves and with the environment. Living organisms are analyzed from many points of view to explain different biological processes that go from macroscopic to microscopic level. To live they must interact with other organisms, its organs, its cells, cell organelles, proteins, genes and transcription factors, etc.

The need to accurately explain biological mechanisms means that the use of Mathematics and Computer Science is essential. Mathematics allows us to represent and to analyze interactions between entities by studying the variables that characterize the entities, their interactions and the factors that affects these processes. The development of Computer Science has made it possible to obtain accurate results when one considers that the number of variables or factors and the mathematical formulas representing the interactions are complex.

The set of biological entities and processes that is studied is called *Biological System.*

Definition 1.1 Complex Biological Systems.

- *In general, a complex system is one composed of interconnected components that as a whole exhibit properties not obvious from the properties of the individual parts ([61]).*

- *Complex Biological Systems are complex systems that come from observation of living organisms. A system is analyzed in a hierarchical way, defining it as being composed of components, where behaviors emerge from the associations and its diversity ([64]).*

The modern approach to study complex biological systems is called *Systems Biology.*

Definition 1.2 Systems Biology. *it is a biology-based inter-disciplinary study field that focuses on complex interactions in biological systems. Its paradigm is opposite to reductionism and favors the integration of multiple components ([64]). In System Biology hierarchical systems are described defining their levels, components and the relations between them ([109]).*

The formal way to describe a system is called *model*, if it uses Mathematics is called *Mathematical model.* The behavior of a biological system is summarized in a set of mathematical relations that characterize it.

Definition 1.3 Mathematical models and modeling.

- *A mathematical model is a description of a system using mathematical language. A model characterize the system by the called system variables, attributes that describe it. Mathematical notions are used to define relations between these variables and the so called factors (or explanatory variables) of the system.*

- *The process of building a mathematical model is called mathematical modeling.*

Definition 1.4 Dynamical models. *If the laws of a mathematical model establish the temporal behavior of the system we call them Dynamic Models. In this case, the future values of the system variables are considered to be functions of the current values (current state) and factor values.*

As one can observe living organisms from different levels, the modeling process implies an abstraction level. One must decide what entities to represent, what factors and interactions to include and how to do that.

For this thesis we will focus on *Dynamical models.* There are many different types of these models, which it is used depends of the analyzed system, what information is available, types of variables, and what questions we want to answer with the model. One can divide the family of models in many subgroups.

The first classification of models is separating them between continuous and discrete models.

Definition 1.5 Continuous and Discrete Models.

- *In continuous models, system variables change continuously over time. The system is abstracted in a model where it is relevant to know the behavior at any time.*
- *Continuous models use functional relations between system variables and factors. A typical way to represent them are the differential equations.*
- *On the other hand, if variables are not modeled at any moment. It is to say, time it is measured at isolated instances. Then the model is called discrete. In this case, the laws use logical and recurrence relations of variables.*

Another classification of models is related with the level of certainty of the variables and the answers of the model, and with the completeness of the model. Models are divided into *deterministic*, *stochastic* ([112]) and *non-deterministic.*

Definition 1.6 Deterministic, Stochastic and non deterministic Models.

- *A model is called deterministic if the system variables are uniquely determined by the factor variables.*
- *In opposite case, the model can be stochastic or non-deterministic. A model is called stochastic when the possibility of obtaining different answers to the same factor values is part of the model. It is to say the system variables values are described by probability distributions.*
- *One says that a model is non-deterministic (ND) if the fact of obtaining different answers is not modeled.*

An special class of models are the *Hybrid models*, that combine different types. The so called *Hybrid systems* are dynamic systems that respond to both continuous and discrete factors. An example is given by differential equations with discontinuous right hand, with applications in biological networks ([39]; [42]; [87]; [83]).

The key challenge is to have a modeling framework capable of integrating in a non-ambiguous way different types of models such as continuous and discrete; deterministic, stochastic and non deterministic happening within different timescales and with hierarchical levels.

In hierarchical approach of *Systems Biology*, a system is modeled defining their components and the relations between them ([78]). A model is defined as being composed of sub-models. *Base formalisms* capable of including the semantics of a wide variety of languages are used to define hybrid models ([13], [47], [86]) of the hierarchical system. In the last years research studies about reusability of models have been vigorous ([109]), searching how to define and simulate composed models in an unambiguous way ([33]).

1.1.1 Fitting biological models

Models try to accurately represent biological reality, by using empirical observations and knowledge. Consequently the notion of model is strongly linked to the observations of the reality that one wants to represent, but the observations are limited.

One wants to build models that are valid to explain the system in general conditions without testing it on all the conditions. The observations that are used to build the model are called *training data*. The larger the training data, the more accurate the model, but too much training data can leads to overfitting. To avoid this problem a statistical technique is to validate the model on other data set called *validation data*, one imposes that the model fits the validation data.

Techniques to correct the model with the validation data exists too. *Machine learning* is the scientific discipline about the design and development of learning systems that allows to learn from the data ([23]).

The process of construction of models can be divided in five steps:

1 Collection of experimental data.

2 Exploratory data analysis.

3 Selection of model type.

4 Building the model: tuning parameters according to training data.

5 Generalization of the model.

One starts by obtaining the experimental observations (step 1). They correspond to tuples of system variables and factors values that will be used to characterize the system. The system variables are the measures that the model explain and the factor variables are the measures used by the model to explain them. The field called *Experimental design* studies how one plans experiments to consider the combinations of factors with economy ([81]).

The experimental data is a set with the form:

$$\{x_{ik}: \quad i = 1, \dots, N, \text{ and } k = 1, \dots, l,$$

where we consider N individuals (called observations too) and l variables (called features too).

A important part of the process is *Exploratory data analysis* (step 2, [105]). Here, one pre-processes the data and selects the final system variables (*feature selection*). Between the *preprocessing* techniques, one uses *outlier removal* to eliminate observations with anomalous or spurious behavior (called *outliers*).

In many practical situations the data must be fixed to avoid that features with superior values scales have a over-estimated influence on the model. The typical way to correct this is by *normalizing* the features according to their means and variance. One can use a linear method of normalization, where the data is supposed to be distributed symmetrically around the mean, or it is possible to transform the data by applying logarithmic, exponential, trigonometric or more complicated functions before normalization (such as softmax scaling, [105]).

In *feature selection or reduction* is reduced the number of variables, by elimination or generation of combined variables. A large number of variables produces redundant information, that may contribute to make the model more complex. To more considered features more model parameters, and more difficult to generalize.

One way to reduce variables is based on statistical hypothesis testing. One eliminates the factors whose effect on system variables is not statistically significant, and the system variables with behavior too similar. In this exploratory step, hypothetical relations between system variables and factors are also analyzed. If it is beneficial for the future model, one detects simple relations between factors to reduce them, such as by replacing them by linear combinations of original factors (*Principal Components Analysis: PCA*, [105]).

The chosen model, result of step 3, is a consequence of the previous exploratory step. It can be automated to use known function such as polynomials, exponential and trigonometrical functions or others as needed by the researcher.

After the type of model has been decided, to obtain the model it is necessary to adjust the parameters or coefficients of the model (step 4). The parameters are obtained by applying the model on training data, the parameters are tuned to accurately fit the training data.

The last phase (step 5) is the *generalization of the model*. Since the process of construction uses a limited set of training data, to obtain general models it is necessary to use post-processing techniques. The model is evaluated and, if it is possible, corrected too. In the evaluation process of the model, one studies its performance on general data. The typical form to proceed is to test the model on not-used observations (validation data). The representativity of training and validation data are essential to obtain more general models.

One can use techniques to repair models with the inclusion of new experimental observations (*Machine learning*, [23]). With these techniques the process of modeling becomes iterative.

1.1.2 Model specifications

With respect to the form in which modular biological models are represented there are different alternatives. For biochemical reactions models, governed by temporal differential equations, the most popular abstraction is *SBML* that defines a machine-readable format ([56]) for mechanistic kinetic models. Last version of *SBML* (Level 3) also allows stochastic and discontinuous transitions by including the notion of *event* ([56]).

Here we used BioRica: a high-level modeling framework that integrates discrete, continuous, stochastic, non-deterministic and timed behaviors in a non-ambiguous way allowing multi-scale dynamics, composition of models, inclusion of SBML models, and hierarchical relations. The work here presented contributed to the development of BioRica, that currently allows continuous dynamics, such as differential equations systems, and hybrid systems with interactions between continuous and discrete dynamics. More details in section 4.4.

1.2 Composition and Hierarchy challenges

1.2.1 Modularity and Hierarchy of biological systems

The human understanding of Biology is modular. Most biological functions are obtained as the result of many interacting processes.

Definition 1.7 Module. *We consider a module to be any interacting entity forming part of a system ([22], case of cell biology: [49]). One talks of functional module when the module has a function associated.*

To explain its behavior, one associates a model to describe each module. Examples of functional modules are those for protein synthesis, DNA replication, glycolysis and other metabolic pathways ([89, 77]).Two questions arise:

1. Is the biology modular?

2. How build/detect modules?

At a cellular level modularity is intrinsically related with *evolution*, one talks about *evolutionary modules* ([22]).

Modules seem to exist naturally in biological systems. Some modules have been reconstituted *in vitro* and other ones theoretically modeled with success. *In silico* reconstruction of the yeast protein-protein interaction network, confirmed experimentally, has demonstrated the existence of particular proteins that interact with many partners to connect different biological processes ([48]). Another fact is that concepts needed for understanding biological systems, such as translation, inhibition or adaptation, arise from interactions among components ([22]). Another one is the robust functioning of modular models of biochemical networks ([65],[66]), that seems to be related to structural stability. In a higher biological level such as physiology, the modular approach has been useful to simulate human answers to changes of pressure or other stimuli, and to develop medicines to control these responses ([45], [46], [106], Figure 1.1).

The detection of modules many times is non-trivial. To represent a system with interacting modules we must decide what elements act in coordinate way to group them, that is to say we must define modules ([49, 88]). Essentially a module is a entity whose function is separable from those of other modules. To build modules we need to identify the functions, the elements that participate in them and how the interactions with other functions work. There are many factors to consider, such as the existence of hierarchy levels, specific and general functions, and the capacity of decomposing higher-level properties of the system into properties of modules.

It is in the construction of modular models of biological systems that synthetic sciences, Mathematics, Computer Science and Engineering, play a fundamental role together with strong interactions with experimental and theoretical biology. There exists a relative consensus about considering hierarchical definitions of modules. In a hierarchical model there are defined several levels of complexity, the modules are defined by groups of variables and parameters in one level that are lumped together to form the elements of the high level of complexity. Consequently, a hierarchical

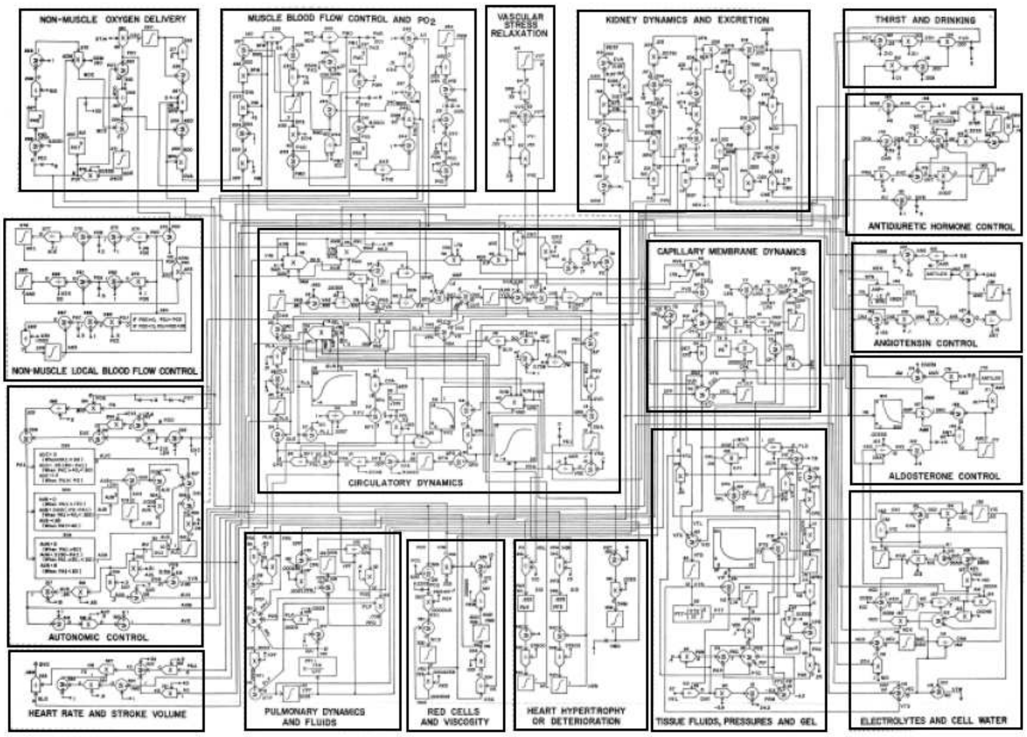

NON-MUSCLE OXYGEN DELIVERY
MUSCLE BLOOD FLOW CONTROL AND PO2
VASCULAR STRESS RELAXATION
KIDNEY DYNAMICS AND EXCRETION
THIRST AND DRINKING
ANTIDIURETIC HORMONE CONTROL
NON-MUSCLE LOCAL BLOOD FLOW CONTROL
CIRCULATORY DYNAMICS
CAPILLARY MEMBRANE DYNAMICS
ANGIOTENSIN CONTROL
ALDOSTERONE CONTROL
AUTONOMIC CONTROL
HEART RATE AND STROKE VOLUME
PULMONARY DYNAMICS AND FLUIDS
RED CELLS AND VISCOSITY
HEART HYPERTROPHY OR DETERIORATION
TISSUE FLUIDS, PRESSURES AND GEL
ELECTROLYTES AND CELL WATER

Figure 1.1: Guyton model [46]. The system is composed by 18 interconnected modules, each one modeling a different function associated with the human circulatory physiology. Systems of differential equations, DAEs and other models are used in each module.

models could include the levels: molecule-cell-tissue-organ-organism-population of a biological system.

There exists a set of tools have been used in different cases to implement the notion of *module*. They can built, among many methodologies, by analyzing the input-output responses of the interaction graph of the system, or grouping together nodes whose variations have comparable timescales ([49, 88]). Clustering tools have been used in some studies. In protein-protein interaction networks modules are defined by statistically analyzing topological properties like connectivity and the effect of remove proteins of the network. It is studied *in silico* in the modeled network, and *in vivo* with knockouts of genes ([89, 48]). With the mathematical notion of modularity, one quantifies the quality of a division of a network by assigning high values to those in which there are dense connections between the nodes within modules but only sparse connections between different modules ([52, 114]).

To study the quality of the modular separation one can analyze if global model properties are inherited from the modules. One would want to ensure properties such as *stability* and *robustness* of the model by these properties in its modules ([88]). *Stiffness* and timescales must be considered too.

1.2.2 Stiffness and timescales

Stiffness is a property of mathematical equations. It is associated to the capacity of numerically solving a problem. It is mainly studied in differential equations, where one uses numerical integrators with discrete time steps.

Definition 1.8 Stiffness. *A equation is stiff if numerical methods for solving it must use small time intervals to be accurate.*

Given a system, it is logical to have equations with different stiffness levels.

Moreover, the interaction between processes developed in different timescales produces a problem to solve complex systems. Sometimes, to see the changes in the behavior it is necessary to compare nearby times, but other times the changes happen in distant times. Often it is necessary to connect biological functions that pass in short time intervals with more slow functions.

Both characteristics, different *stiffness* levels and *timescales*, generate a complication to solve systems. Computation times and accuracy are affected. The use of modules solves in part these problems. If the equations solved by a module are stiff one uses smaller time intervals only at that module. If each module is capable to have its own timescale, one improves the representation of the processes of a system.

In fact, stiffness and timescales can be used as criteria to build or to qualify modules. Ideally functions at the same module must have similar stiffness and timescales.

1.2.3 Composition: its challenges

The act of building a model that is valid for two or more modules is called *Composition.*

Definition 1.9 Composition. *Let us consider the modules M_1 and M_2 (with two associated models). The composition of M_1 and M_2 is the model that explain the behavior of both interacting modules.*

An important need to develop science is to be able of reuse it. The advance of science is based on the reuse, the application and the improvements of the scientific discoveries. As said Isaac Newton, "... If I have seen a little further it is by standing on the shoulders of Giants." In the case of building models, we need reuse the existing models to look beyond.

A good implementation of this concept is essential to take advantage of the modularity of biological systems to build accurate and complete models. The definition of composition must be sufficiently flexible to be capable of joining modules defined with different types of models, and to reuse modules that have been *a-priori* defined. This allows us to learn and to integrate knowledge of diverse type. The model can be extended and improved by introducing new modules that relates different functions, or that model behaviors of factors.

Another characteristic that one wishes is the reduction of composed system properties in properties of modules. That is to say, questions about the global system must can be answered by analyzing the modules and the interaction between them. Ideally by looking the modules it must be easier to answer to functional questions.

According to the observed relation between modularity and Hierarchy, the composition must be capable of including hierarchical relations to recover the *be part of* relation. It must be possible to go up and down the levels by using reduction operations that lump together variables and parameters and specification operations respectively.

The composition must answer coherently to refinements of its modules. If a module is refined by adding information of its behavior, the composition of this module with another module must result refined. In the opposite direction one want to check refinements of composed models by proving refinements between modules, this is called *Assume-guarantee* rules of composition under refinements.

It is necessary to impose technical requirements to avoid implementation problems. As explained in section 1.2.2, composition has to be capable of integrating modules with different stiffness and timescales.

1.3 Presence of randomness and use of Hybrid Systems in Biology

Randomness has two reasons to be present in biological models: sampling and incomplete information ([112]). Models are built by using data samples that allows us to build and validate them on a fraction of the total population. As we explained in section 1.1.1 one tries to use representative experimental data to obtain a good representation of the reality, but this statistical sampling always provokes the so called *sampling error* of the model.

The other important error source is the fact that the available information often is incomplete. Due to obtain biological data often is expensive, one needs to work with incomplete information. Moreover, the intrinsic unknown nature of some biological process makes difficult to predict accurately the behavior. Non-deterministic and stochastic models allow us to include random decisions.

These facts give space to randomness in Biology. One uses non-deterministic and stochastic models to include estimations of error and random (or non-deterministic) decisions.

In addition, one considers Hybrid systems to combine different types of factor affecting a biological system. One uses continuous models to describe gradual changes over time, discrete models for instantaneous changes, deterministic models to represent completely predictable behaviors and stochastic models to introduce randomness. Changes in the conditions of the system, such as levels of nutrients or environmental conditions can change the behavior laws (equations) of the system, or biological facts such as a gene is activated, to be expressed as protein, by a transcription factor or regulator give power to this approach.

Hybrid systems as we define them here (section 2.1), are one way to perform this combination. Some applications in biology are shown in [1]. In a recent article by Alfieri *et al* ([3]) is used the theory of Hybrid systems to simulate the $R-$point transitions that occur during the cell cycle. Applications of Biotechnology to industrial processes, such as wine fermentation, are capable of approaching by Hybrid systems too (section 5).

An important application area is *Gene Regulatory Networks* (section 6.2). Some biological facts such as a gene is activated, to be expressed as protein, by a transcription factor or regulator give power to the idea of using *hybrid models*. In this thesis (section 6), they are analyzed some hybrid models of Gene Regulatory Networks with applications in Cell differentiation ([36, 96]) and cell division cycle ([83, 40]).Another application area here considered was the modeling of wine fermentation kinetics by reconciling different models ([10], section 5). According to initial condition levels and temporal phase the system switches between different modes, to obtain good predictions of its dynamics.

1.4 Outline of this thesis

In Chapter 2 we begin by describing the known elements about hybrid systems, and the particular case of switched systems in which the continuous dynamics is described by differential equations. We continue by summarizing the theory of stochastic transition systems, in which the discrete dynamics depend on stochastic decisions and schedulers that solve non-determinism. Finally we explain the main ideas about composition of models: synchronization of events according to the theory of transition systems and input-output connections.

In Chapter 3 we present our modeling approach of complex biological systems. We begin by describing our approach to reuse and reconcile models. We recall the main arguments to use modular descriptions of biological systems (explained in this chapter), and present ideas about how to modeling biological systems by hybrid systems and composition. After that, we formalize the notion of hybrid system: the discrete dynamics of mode variables is described by stochastic transition systems, while the effect of the mode transitions on the continuous dynamics is described by coefficient switches that modify coefficients of the continuous model, or by strong switches that change the model itself. We finish by presenting the guarantees of our approach, and summarizing the main contributions of this work with respect to modeling.

In Chapter 4 we present the implementation of our modeling ideas and theories. In the first section we describe the requirements of such a implementation. After that, we describe the computation steps of the solving process, as a general schema independent of the implementation used. The next sections are related with the BioRica framework, which we use for implementing our approach. We explicit the contributions done in this thesis to the development of BioRica to satisfy the implementation requirements, and describe the implementation of hybrid systems with BioRica, the syntax, the semantics and the process of simulation. We finish by summarizing the conclusions of this chapter.

In Chapter 5 we apply the approach presented here to modeling, implementing and simulating the wine fermentation kinetics. We analyze this application as a particular case of reconciliation of competing models. This hybrid model results from the reconciliation of three wine fermentation models ([10]): Coleman ([28]), Scaglia ([94]) and Pizarro ([85], [92]). For each factor configuration, one chooses the model that best fits the experimental data of three papers: [85], [75] and [79] used as training. In function of the initial conditions one chooses the model and arriving at the stable fermentation phase, the model is updated to obtain the best predictions. The effect that produces the level of nutrients on the behavior of the system is successfully described by switching the model to include competence coefficients when the resources are scarce.

In Chapter 6 we apply our approach to cell fate decisions associated with the formation of specific cells. We describe these types of processes by using the theory of switched systems, reusing and composition of models. The differentiation process is described by a system of differential equations, wich is affected by regulatory processes (implemented by reusing SBML models) that switch coefficients to favor one or other lineage. We focus in the case of bone and fat formation, in which the dynamics of the differentiation of progenitor cells into osteoblasts and adipocytes is controlled by the interactions between different processes. With this model, we want to predict the changes in bone or fat formation by stimulating (or inhibiting) the Wnt signaling pathway, the $PPAR\gamma$ pathway, the division of progenitor cells, and the apoptosis of progenitor or osteoblast cells. Based on this, one can analyze *in silico* the physiological responses to treatments of bone mass disorders based on the Wnt signaling pathway, and to explore the efficiency of new medical strategies before testing them in animal models.

Finally in Chapter 7 we concludes and discusses the basis, scopes and future improvements of our work.

Chapter 2

Preliminaries

Hybrid systems allow us to describe biological systems by including deterministic, non-deterministic, stochastic, continuous and discrete elements. With this approach one captures behavior changes by using the theory of *Stochastic transition systems*.

There are many applications of *Hybrid models* in other science areas. In modeling of physical phenomena it is usual to have partial or ordinary differential equations with right hands piecewise defined. Some classic examples are *heat and wave* equations, where changes in the medium produce hybrid systems ([95, 11]). Hybrid models are very useful to analyze *electronic circuits* and *electrical networks*. Variables like the current are continues, while switches are real electronic dispositives ([58, 67]).

In Biology, many dynamical systems are represented by ordinary differential equations. Changes in environmental conditions or controlled factors modify the development of diverse processes, which interact to realize the complex system behavior. In biological modeling randomness and non-determinism are common (section 1.3).

The non-ambiguous combination of different models by composition allows us to build complete models. One can reuse and reconcile existing models to obtain more general models.

2.1 Hybrid Systems

According to our definition of *Complex Biological Systems* (definition 1.1), the complexity of biological processes arises not only from the association of many components, but from the components too ([64]). One associates components with different characteristics, properties and laws.

This association of diverse behaviors carries to the use of *Hybrid models* to represent biological processes. With the aim of obtaining good modeling, wide range of models are allowed. The existence of different types of models to explain connected processes makes to be necessary to define tools to integrate them.

Definition 2.1 Hybrid systems and models. *One talks of Hybrid System if it responses to both continuous and discrete factors.*
Models that include both types of variables are called to be Hybrid models.

In our case, we are interested in *Hybrid dynamic systems*. They are described using a mixture of continuous dynamics, discrete dynamics, and logical relations. The dependent variables of the system, $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$, are called *state variables* in analogy with Transition Systems. One considers as factors the continuous variables $u = (u_1, \ldots, u_k) \in \mathbb{R}^k$, and the mode variables $mode \in M = \{1, \ldots, M\}$. The values over time of these variables are called respectively the continuous and discrete dynamics.

Hybrid Systems can be seen from two points of view: function ([18, 100, 104]) and implementation ([50, 60]). The first one, called *switched systems*, focuses in human comprehension, the second type is more general and focuses in automatic interpretation. This second approach uses tools of Automata theory and is easy to understand in terms of Transition Systems theory ([7, 29]). Continuous variables evolve according to continuous models, but at any time mode changes can change the definition of the continuous model. These changes are called *mode transitions* and are considered to be transitions in the sense of Transition Systems theory.

Given an action producing a mode transition, the next mode is chosen according to transition probabilities and system schedule laws. The conditions that allow mode transitions are called *guards*. For each mode, the system evolves in function of its continuous dynamics. In literature, the discrete dynamics is modeled by deterministic models, and allowing non-determinism with ambiguous guards. Here we extend the formal notion of hybrid system to include stochastic behaviors by considering stochastic transition systems ([29], section 2.2).

Hybrid systems are described using a mixture of continuous, discrete dynamics and logical relations. The continuous and discrete dynamics interact, so that the changes in discrete variables provoke changes in continuous models and vice versa. To define the dynamics of Hybrid Systems it is necessary to also describe interaction between the continuous and discrete dynamics.

2.1.1 Continuous dynamics

The best known form of dynamic system is a set of ordinary differential equations. In such models, the dynamics of one or more variables (x) is described in a equation with the form shown in equation 2.1, where the temporal rates of x at time t are denoted $\dot{x}(t)$ and called *temporal derivatives*. So, at any time t the temporal derivative of each variable x_i depends on a function (F_i) of all the values of state variables $x(t)$ and the values of continuous control variables $u(t)$. One finds notable examples in classic mechanics such as the equation of motion, and in biological systems such as models of two or more populations competing by resources (Lotka-Volterra equations, [98]).

$$\dot{x}(t) = F(x(t), u(t)) \tag{2.1}$$

The solution of equation 2.1 depends on the initial values of x, called *initial conditions*. Often it is not possible to obtain solutions with explicit formulas, and one has to use numerical methods to approximate the solutions at a given time using iterative schemes.

In addition, when physical or mechanical systems are determined by conservation laws, such as the conservation of energy or momentum, these systems are modeled by equation 2.2. The function G is given by algebraic combinations of the state variables x, derivatives $\dot{x}$ and continuous control u, which is constrained to be equal to zero. The model is called differential algebraic equations (DAE) ([38]).

$$G(\dot{x}(t), x(t), u(t)) = 0 \tag{2.2}$$

In general DAEs, we can use the form of equation 2.2 to separate the model into a system of differential equations and a set of implicit algebraic constraints (equations 2.3, 2.4). This is done by decomposing the variable x into two variables x and z ([68]). The vector of variables x represents the dependent variables whose derivatives are as in the equation 2.1, while z corresponds to algebraic variables whose derivatives are not considered.

$$\dot{x}(t) = H(x(t), u(t), z(t)) \tag{2.3}$$
$$\tilde{G}(x(t), u(t), z(t)) = 0 \tag{2.4}$$

As example, let us consider the Lotka-Volterra equations ([98]) to model two populations: prey x, and predator y. If we assume that the prey have an unlimited food supply (unlimited proliferation in absence of predator), the dynamics is described by the set of differential equations 2.5-2.6 below:

$$\dot{x}(t) = x \cdot (\alpha - \beta \cdot y) \tag{2.5}$$
$$\dot{y}(t) = -y \cdot (\gamma - \delta \cdot x),\ where \tag{2.6}$$

α is the growth rate coefficient of the prey, β the rate of depredation coefficient, γ the predator mortality or emigration coefficient, and δ is associated to the growth of the predator population. All of these coefficients are considered to be positive constant over time.

One can describes the same system but with limited supply by the DAE system with differential equations 2.7, 2.8 and algebraic constraints (with form 2.4) 2.9, 2.10. The coefficients β, γ, δ, a, b and F are positive constant; while α and z are algebraic variables.

$$\dot{x}(t) = x \cdot (\alpha - \beta \cdot y) \tag{2.7}$$
$$\dot{y}(t) = -y \cdot (\gamma - \delta \cdot x) \tag{2.8}$$
$$\alpha = k \cdot z \tag{2.9}$$
$$a \cdot x + b \cdot y + z = F \tag{2.10}$$

The equations 2.9 and 2.10 describe how the growth rate coefficient decreases if the rate of food consumption $a \cdot x + b \cdot y$ is near the allowed rate F.

Partial differential equations are also common in many areas of science and technology. Such models use the form of equation 2.2, but G is allowed to depend on $k-$order derivatives of each state variable x_i with respect to any of the other ones x_j $(i, j \in \{1, \ldots, n\})$.

2.1.2 Discrete dynamics

The discrete dynamics is given by the changes over time of the mode variables *mode*. Discrete dynamics can be deterministic, stochastic or non-deterministic. Approaches from the implementation point of view favor the use of all these possible behaviors.

2.1.3 Interactions between continuous and discrete dynamics

The way in which continuous and discrete models interact is by switches, which control changes in the continuous model by modifying the values of coefficients or changing the definition of the continuous model (F in case of ordinary differential equations as 2.1, and G for DAE or partial differential equations with form shown in equation 2.2). The discrete dynamics of the system, modeling the temporal evolution of the discrete variables, is determined by such switches. In the same way that discrete dynamics can be stochastic or non-deterministic, the values of the discrete factors can provoke switches in the system both stochastically and non-deterministically. That is to say, given a discrete factor value it switches the continuous model according to deterministic, stochastic or non-deterministic rules.

At the same time, the values over time of the discrete factors depend on time, the previous values, and the values of the state variables. Specific values of time or state variables provoke the change in the value of the discrete factors.

2.1.4 Switched Systems

An special type of *Hybrid system* are the so called *Switched Systems* ([18, 100]).

Let us consider a *Hybrid System* whose model has continuous variables x and discrete variables m. Independent of x, the value of the right hand function F changes in function of the value of discrete variable m. If for each value element of S the model change its form, laws, its equations: its *configuration*. We will say in these cases, that changes in discrete variables values carry *switches* between model configurations, the *Hybrid system* is called *Switched System*.

Each model configuration is called a *mode*, the set M corresponds to the set of modes. To be capable of simulating it, generally it is assumed that there are only finite switches in finite time.

Switched Systems, associated to differential equations 2.1, have the form of the equation 2.11.

$$\dot{x}(t) = F_{m(t)}(x(t), u(t)), \tag{2.11}$$

where $F_{m(t)}$ depends on the mode $m(t)$. It is to say, m defines the mode dynamics of the system (Figure 2.1).

The dynamics of m can be represented by the equation 2.12

$$m(t) = G(x(t), u(t), m(t^{-})),\text{where } G : \mathbb{R}^n \times \mathbb{R}^k \times M \longrightarrow M \tag{2.12}$$

Starting at $(x(0) = x_0, u(0) = u_0, m(0) = i \in M)$ the continuous trajectory evolves over time according to $\dot{x}(t) = F_i(x(t), u(t))$, at the first time $t_j > 0$ where

$\exists j \in M : (x(\cdot), u(\cdot)) \in G(\cdot, \cdot, i)^{-1}(j)$ then the variables become $(x(t_j), u(t_j), j)$ and the process continues.

Other approach to represent *switched systems* is by a set of Differential Algebraic Equations ($DAEs$). This implicit representation takes the form of Equation 2.13.

$$0 = \tilde{G}(x_r(t), u_r(t), m(t)) \tag{2.13}$$

The variables x_r and u_r are reduced versions of x and u. For each possible value of $m(t)$ there is different DAE representing the system dynamics. The implicit equation 2.13 defines the discrete dynamics. It has a solution for each $m(t)$. The equation 2.1 gives the continuous dynamics.

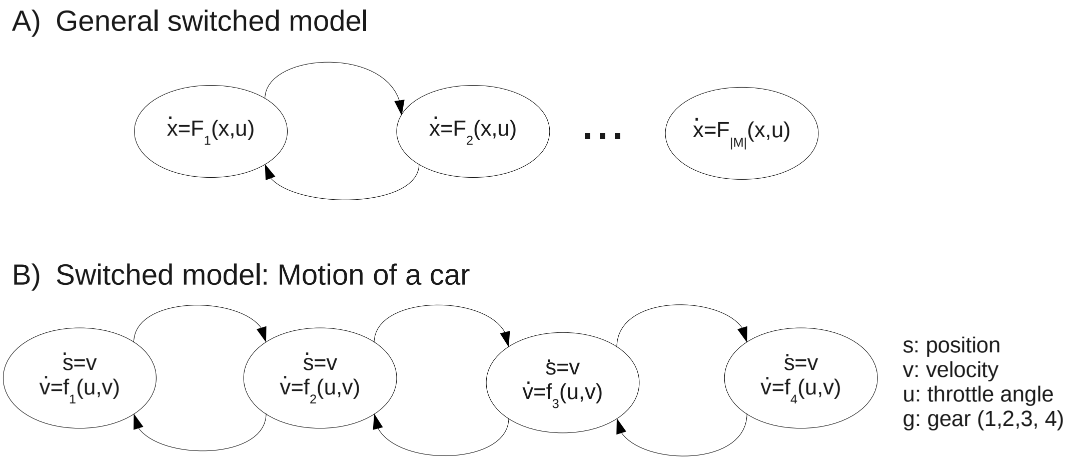


Figure 2.1: *A)*: The dynamics of a *Switched System*. The model moves between different modes, nodes in the picture, defining its discrete dynamics. At each mode, the system changes its continuous dynamics. *B)*: The *Switched System* of the motion of a car. The continuous dynamics, represented by the evolution of the variables position s and velocity v, changes its laws according to the engaged gear.

A very intuitive example of switched system is the *Motion of an automobile with a manual gearbox* ([100], Figure 2.1). This system is characterized by two continuous variables: velocity and position. The manner in which these variables responds to the throttle angle depends on the engaged gear. In each *mode* of the engaged gear the dynamics evolve in some continuous specific way. The dynamics of this system is hybrid in its nature.

2.2 Stochastic Transition Systems

We use the theory of Stochastic transition systems (STS) to describe transitions in Hybrid systems (section 2.1). Introduced by De Alfaro ([29]), STS cover the need of modeling dynamics of biological systems with inclusion of uncertainty and the presence of uncontrolled external factors. The implemented notions of Composition and Synchronization allow us to define systems by composing interacting modules and synchronizing actions.

The state of the system in a given time is represented by the value of its observable (*interface* and *external*) state variables. The system evolves over time by changing its state. The transitions between states are caused by actions and spent time. One studies the stochastic behaviors of the system over time, depending on the actions at each state and the transition chosen for each action.

STS correspond to Continuous Time Markov Decision Process (*CTMDP*) ([15], section 2.2.3) in that one uses *randomized schedulers* to solve non-determinism in simulations. The concepts were extended to continuous domains by using probability measures on sub set of states and actions ([25]). Let us begin with the theory.

Definition 2.2 Transition System. *A Transition System S ([7]) is a tuple* $(Q, Q_0, A, \longrightarrow)$ *verifying:*

1. *Q is a set (commonly finite) of states.*

2. $Q_0 \subset Q$ *is the set of possible initial states.*

3. *A is a set of actions.*

4. $\longrightarrow \quad \subset \quad (Q \times A \times Q)$ *is the set of transitions.*

One denotes $q \xrightarrow{a} q'$ *if* $(q, a, q') \in\longrightarrow$. *One defines the set of actions available at the state q as* $A(q) = \{a \in A : \exists q' with q \xrightarrow{a} q'\}$.

Definition 2.3 Executions and observable executions. *An execution is a sequence* $\sigma = q_0 a_0 q_1 \ldots q_n a_n q_{n+1}, \ldots$. *If* $\forall i \geqslant 0 \quad (q_i, a_i, q_{i+1}) \in\longrightarrow$ *one says that* $\sigma = q_0 \xrightarrow{a_0} q_1, \ldots, q_n \xrightarrow{a_n} q_{n+1}, \ldots$ *is observable in S.*

Definition 2.4 Sets of executions. *Exec, Exec*, Exec*^{w} *denote the set of executions, finite executions ending with an state, and infinite executions respectively.*

Definition 2.5 Traces. *Sequences of states are called Q−traces, of actions A−traces.*

The first state of a execution σ is denoted $first(\sigma)$. The last one, if the execution is finite, is $last(\sigma)$.

In the example (Figure 2.2), the set of states is $Q = \{q_0, q_1, q_2\}$, q_0 is the (unique) initial state, the set of actions is $A = \{s, a, e\}$, and the transitions of the system are (q_0, s, q_0), (q_0, a, q_1), (q_1, e, q_0) and (q_1, e, q_2). An observable finite execution is $q_0 s q_0 a q_1$ with finite trace of actions sa.

If ones assigns probabilities to the transitions, it is called a *Stochastic Transition System* ([29]).

Definition 2.6 Stochastic Transition System. *One says that S is an Stochastic Transition System if it is a transition system (definition 2.2) that fulfills* 5 *and* 6.

5. *An execution of S is chosen to start at* $q \in Q_0$ *with probability* $P_0(q)$.

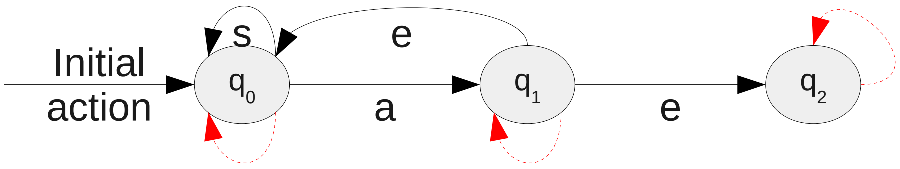


Figure 2.2: Example of Transition system for cell division. Circles represent states, black arrows transitions. Transitions are labeled by actions, but different transitions can have the same label to represent that an action may produce many responses. The initial action selects the initial state of the system, always begin at q_0. States: q_0, q_1, q_2, transitions: (q_0, s, q_0), (q_0, a, q_1), (q_1, e, q_0) and (q_1, e, q_2). Red arrows are inherent idle transitions that are added to the system to assure infinite executions, and to define composition.

6. *Given $q \in Q$, $a \in A$, the probability to choose to continue the state $r \in Q$, verifying $(q, a, r) \in \longrightarrow$, is $P_a(q, r)$.*

In the example 2.2, for the state q_1 we see that given the action e, two successors q_0 and q_2 are possible. If we add probabilities to each option we have an Stochastic Transition System. It is not necessary to define probabilities to start the execution because only q_0 was considered as initial state.

2.2.1 Randomness and non-determinism

We saw that randomness is present in the definition of Stochastic Transition Systems (definition 2.6): given an state and an action there are two or more possible transitions to continue the execution (Figure 2.2, state q_1 and action e).

The non-determinism arises from the fact that given an state q of an finite observable execution, many actions could be available. By definition, this fact is not probabilistically modeled *a priori* and hence, to simulate executions, one must solve it. In the example of Figure 2.2, given the state q_0 it could select the action s or the action a with different successors.

A way of solving the non-determinism is to introduce the concept of *Scheduler*. The Scheduler, [113], specifies the criteria to choose actions. They can be deterministic or randomized, with memory or not. With these elements, a first definition of randomized scheduler is:

Definition 2.7 Randomized Scheduler. *A (randomized) scheduler is a function $D : Exec^* \longrightarrow Distr(A)$ verifying that $\forall \sigma \in Exec^*$ $supp(D(\sigma)) = A(last(\sigma))$.*

With the aim of having infinite executions one add an implicit *idle* transition (that stay in the state) to each state of the Stochastic Transition System. Consequently, an *STS* produces a *Markov Decision Process* (definition 2.8) for each non-determinism solution.

Definition 2.8 Markov Decision Processes generated by Stochastic Transition Systems. *If for each $q \in Q$ we impose that $A(q)$, the set of actions available at q, is non-empty for all $q \in Q$, the triple $\Pi = (Q, A, P)$ (with P given by $P_a(q, r)$ the probability of a transition from q to r when action a is selected) is called a Markov Decision Process (MDP).*

Let us consider our example (Figure 2.2) and the scheduler that selects after the state q_0 the action s with probability 0.5 and the action a with probability 0.5. This scheduler allows us to obtain executions of the system and it does not have memory: it is not important how one arrives to q_0 to decide the next action. We say that a scheduler D is memoryless if it can be written as function from Q to $Distr(Q)$, as in this case. A scheduler is deterministic if it returns a Dirac distribution for any execution, that is to say after each execution it is assigned an unique action to continue (for example if in q_0 the scheduler always selects the actions a).

With these elements one can simulates the Transition System, whose results depends of the scheduler that is chosen. But, what does the dynamics of the system represent? what do the states represent? We have not established the relation between a *Transition System* and *Dynamic models*.

2.2.2 Dynamics of system variables: Probabilistic modules

In the original approach of *Transition Systems* ([7]), one associates states with values of variables. Given a Transition System $S = (Q, Q_0, A, \longrightarrow)$, each state in Q represents a valuation of the system variables. With this interpretation, state transitions are changes in system variables values.

Definition 2.9 Typed state variables. *It is considered the set x* of typed state variables to be a set of names of variables, each one of them with finite domain. Q is the finite set of interpretations of X, at each state $q \in Q$ the value of each variable $x \in X$ can be obtained. The set Q_0 is written as $\{q \in Q : q \quad \text{satisfies} \quad \Theta\}$, where Θ is an assertion over X. Consequently, actions correspond to changes in the system variables.

Let us consider the example of Figure 2.2, and the following interpretation of states:
Variables: N : number of cells, A : activity level, G : growth.
States (valuations of variables):

- $q_0 = \{A = 0, N = 1, G \in [0, 1]\}$,
- $q_1 = \{A = 1, N = 1, G \in [0, 1]\}$,
- $q_2 = \{A = 1, N = 2, G \in [0, 1]\}$.

Given a system, some variables are modeled and others, representing factors with unknown behavior, are not controlled. Continuing with our example (Figure 2.2), let us consider that the growth G is not controlled while A and N are part of the model, but only N can be accessed by other Transition System (A is private or internal). These considerations are denoted $ext = \{G\}$, $intf = \{N\}$

and $priv = \{A\}$. The interface $intf$ and external ext variables form the set of observable variables, $obs = \{N, G\}$ in our example, while private and interface variables form the set of controlled variables, $cont = \{A, N\}$.

So, we can consider the following **Interpretation of the actions:**

- *initial action*: $\varnothing \longrightarrow \{N := 1, A := 0\}$,
- $s : \{G, N, A\} \longrightarrow if(G \cdot N < 0.5) \qquad \{N := N, A := A\}$,
- $a : \{G, N, A\} \longrightarrow if(A = 0) \qquad \{N := N, A := 1\}$,
- $e : \{G, N, A\} \longrightarrow if(A = 1, N = 1) \qquad \{N := N + 1, A := A\} \quad \vee \quad \{N := N, A := 0\}$ each option with probability 0.5.

In our example, only the action e is strictly probabilistic. The action s includes a dependence on the external variable G, not controlled by the model, and at q_0 both actions s and a could be possible. The action *initial action* selects the first state, the first valuation of variables that in this case is not probabilistic. As the external variables are not controlled by the system, the scheduler that selects the next action cannot depend on private variables of the module. In order to correctly define schedulers on composition of modules, the controlled variables are partitioned into *atoms* (definition 2.10) with associated read variables. An Stochastic Transition System, with these interpretations of states and actions is called a *Probabilistic module* (definition 2.11).

Definition 2.10 Probabilistic X-atom. *A probabilistic atom N consists of a set $read(N) \subseteq X$ of read variables, $ctr(N) \subseteq X$ of controlled variables, $init(N)$ an initial action from $\varnothing$ to $ctr(N)$, and an update action $upd(N)$ from $read(N)$ to $ctr(N)$.*

Definition 2.11 Probabilistic module. *A probabilistic module P consists of two parts:*

- ***Declaration:*** *variables $X(P)$ partitioned into: $ext(P)$ the external variables, $intf(P)$ the interface variables, and $priv(P)$ the private variables.*
- ***Body:*** *finite set $atoms(P)$ of probabilistic $X(P)$-atoms such that:*
  1. $intf(P) \cup priv(P) = \bigcup_{a \in atoms(P)} ctr(a)$,
  2. $\forall a_1, a_2 \in atoms(P) : ctr(a_1) \cap ctr(a_2) = \varnothing$.

In summary, for each module we need a scheduler to simulate the environment (the initial and updated values for the external variables), and a scheduler for each atom to chose the initial values and the dynamics of the variables controlled by that atom (more details in subsection 2.3.1).

2.2.3 Inclusion of Time: Continuous-Time Markov Decision Process

Although with these elements one can study the dynamics of the system, an important element is not considered: the advance of time. To each transition one can associates a unity of time, but it is logical to think that not all the transitions use the same quantity of time.

The usual way to include time is by considering *transition times* and to link the probabilities of successors with the time ([29], [113]).

Definition 2.12 Stochastic Transition System as Timed Probabilistic System.
If we add to the Markov Decision Process $\Pi = (Q, A, P)$*, the set of possible initial states* Q_0 *and a labeling time function time, where* $time(q, a)$ *is a non-negative value representing the expected amount of time spent at* q *when action* $a \in A(q)$ *is chosen,* $\Pi = (Q, A, P, Q_0, time)$ *is a Timed Probabilistic System (TPS).*

According to [29], some transitions can be *immediate* and others *delayed*. The first ones have priority and do not spend time, the last ones have associated strictly positive times of transition.

To relate spent times with the probability of transitions P, one considers that for each action the spent time of transitions has exponential distribution. That is to say, one includes as part of the model a three-dimensional rate matrix R.

Definition 2.13 Rate matrix. *Is a three-dimensional matrix that satisfies:*

1 $R : Q \times A \times Q \longrightarrow \mathbb{R}^+$.

2 Given an state q *and an selected action* a*, the probability that any of the successors is reached within time* t *is*

$$1 - e^{-R(q,a,Q)\cdot t}, \textit{ where } R(q, a, Q) = \sum_{r \in Q} R(q, a, r)$$

3 Given an state q *and a chosen action* a*, the transition is selected according to competition between possibilities. That is to say, the probability to chose the successor* r *is* $P_a(q, r) = \frac{R(q,a,r)}{R(q,a,Q)}$.

In this case $\Pi = (Q, A, R, P_0)$ is called a *Continuous-Time Markov Decision Process* (*CTMDP*), object that has been strongly studied in *Probabilities* ([15]).

Exponential distributions describe the time between events, in a process in which the events occur continuously and independently at a constant average rate (*Poisson process*, [112]). Given an state and an action, the properties 2 and 3 are direct consequences of supposing that the transition options are time independent and exponentially distributed.

In our example, Figure 2.2, by adding the rate matrix R to the model we have that

$$P_e(q_1, q_0) = \frac{R(q_1, e, q_0)}{R(q_1, e, q_0) + R(q_1, e, q_2)}$$
$$P_e(q_1, q_2) = \frac{R(q_1, e, q_2)}{R(q_1, e, q_0) + R(q_1, e, q_2)}$$

According to property 2 in definition 2.13, the *soujourn time* in $q \in Q$ under an action a is determined by a exponential distribution that includes all the probable transitions to leave q by the action a. In our example, for the state q_1 and the action e the *soujourn time* has exponential distribution with rate $R(q_1, e, q_0) + R(q_1, e, q_2)$. The probability to reach an successor within time t is $1 - e^{-((R(q_1,e,q_0)+R(q_1,e,q_2)))\cdot t}$.

2.2.4 Schedulers and Timed executions

As we explained in section 2.2.1, the existing non-determinism in Transition Systems is solved by *randomized schedulers*. A randomized scheduler assigns probabilities to the possible actions given a finite execution. When these probabilities are chosen, one can completely simulate the dynamics of the system.

The dynamic of the model includes the time, consequently one can add the transition times to the notion of execution of a *Stochastic Transition System*.

Definition 2.14 Timed executions. *A timed execution is a possibly infinite sequence of states, actions and time values* $\sigma = q_0 a_0 t_0 q_1, \ldots, q_n a_n t_n q_{n+1}, \ldots$. *The set of timed executions, finite timed executions ending with an state, and infinite timed executions are denoted* Σ, Σ^*, Σ^w *respectively. Given a stochastic transition system one says that a timed execution is observable if it can be obtained with probability not* 0 *by simulating the system.*

In our example, let us consider exponential distributions for spent times as is shown in the Figure 2.3.

In addition, let us consider a randomized scheduler D that solves the non-determinism in q_0 by selecting both the actions a and e with the same probability (0.5). On other states, the scheduler answers with the unique possible action (with probability 1). This scheduler is memoryless, it depends only on the current state. Given $T > 0$, we will compute the probability to obtain the finite execution $q_0 a q_1 e q_2$ with a total spent time lower than T. It corresponds to obtaining $P(\{q_0 a t_1 q_1 e t_2 q_2 : t_1 + t_2 \leqslant T\})$, assuming the scheduler to be D. At computing:

$$\begin{aligned} P(\{q_0 a t_1 q_1 e t_2 q_2 : t_1 + t_2 \leqslant T\}) &= D(q_0)(a) P_a(q_0, q_1) D(q_0 a q_1)(e) P_e(q_1, q_2) P(t_1 + t_2 \leqslant T) \\ &= 0.5 \cdot 1 \cdot 1 \cdot \frac{1}{1+1} \cdot \int_{t_1=0}^{T} \int_{t_2=0}^{T-t_1} 2e^{-2t_1} \cdot 2e^{-2t_2} dt_2 dt_1 \\ &= 0.25 \cdot \big(1 - (1+2T)e^{-2T}\big) \end{aligned}$$

For example, if $T = 1$ the probability is $0.25 \cdot (1 - 3e^{-2}) \approx 0.1485$.

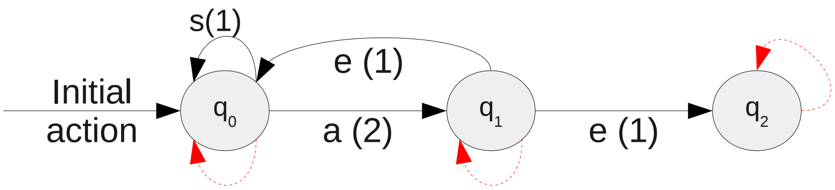


Figure 2.3: Example of Stochastic transition system. The presence of randomness (two possible successors of q_1 under the action e: q_0 and q_2) and the time advance are modeled by considering that spent time of transitions has exponential distribution. We show the rate values R at each arrow between parenthesis. The non-determinism, associated to have two or more actions possible at the same state (case of q_0), can be solved by a *randomized scheduler* that assigns them probabilities.

2.2.5 Extensions of Stochastic Transition Systems

To continue we explore some of the main existing extensions of *Stochastic Transition Systems*, that generalize the notion of scheduler and the states and actions domains.

It is possible to extend the notion of scheduler to consider that the selection of actions depends not only on previous states and actions, but also on the transition times that have been used ([25, 113]). It corresponds to functions from finite timed executions (Σ^*) to probability measures on actions ($Distr(A)$)

Definition 2.15 Randomized timed history-dependent scheduler. *A randomized timed history-dependent scheduler is a function* $D : \Sigma^* \longrightarrow Distr(A)$ *verifying that* $\forall \sigma \in \Sigma^*\ supp(D(\sigma)) = A(last(\sigma))$.

In the Figure 2.4, we show part of a transition system. To go out from q_1, there are two possible actions: a and b. Let us consider the scheduler D_2 such that:

$$D_2(q_0ctq_1) = \begin{cases} \delta_a \text{ if } t \in V \\ \delta_b \text{ otherwise,} \end{cases}$$

where δ_a is the distribution that selects the action a with probability 1, analogously δ_b for the action b. This scheduler satisfies the last definition, depending explicitly on time.

Thus, given this scheduler the probability to reach q_3 in two steps is

$$P(t \in V) \cdot P_a(q_1, q_3) = \int_{t \in V} 2e^{-2t}dt \quad \cdot \frac{1}{2},$$

that is defined if the set V is measurable in $\mathbb{R}^+$. Consequently, to include time dependencies in the scheduler definition is necessary to have considerations on measurability.

Another extension of scheduler arises from observing that an important element of non-determinism is not being considered: given an action, maybe one does not

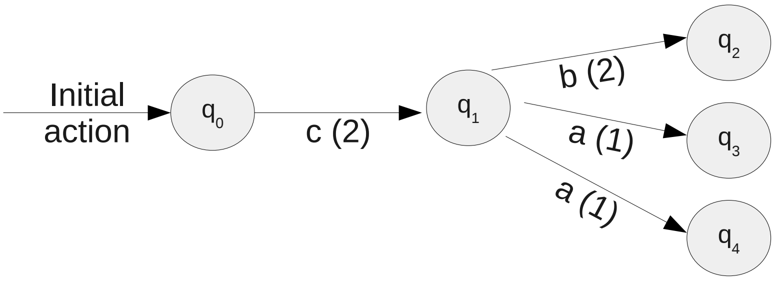


Figure 2.4: Stochastic Transition system with inclusion of rate values for spent times (between parenthesis). At state q_1 two actions, a or b.

know the probability of each transition. In our example (Figure 2.2), given the action e after q_1 the probability to obtain each successor (q_0 or q_2 in this case) could be unknown ($P_e(q_1, q_0)$ and $P_e(q_1, q_2)$ with Stochastic Transition Systems notation, or R and if we work with exponential distributions of transition times). We need a definition of scheduler that considers this case too.

In [25] is introduced a wider definition of *Stochastic Transition System* that corrects this last problem in the scheduler definition together with extending stochastic transition systems to continuous spaces of states and actions. Here, we modified the definitions to allow a measurable set of possible initial states.

Definition 2.16 Stochastic Transition Systems with continuous spaces. *A Stochastic Transition System is a tuple* $\big((Q, F_Q), Q_0, (A, F_A), \longrightarrow\big)$ *where:*

- (Q, F_Q) *is the analytic space of states,*
- $Q_0 \in F_Q$ *is the set of possible initial states,*
- (A, F_A) *is the the analytic space of actions,*
- $\longrightarrow \subset T = Q \times A \times Dist(Q, F_Q)$ *is the set of probabilistic transitions.*

The fact that in a general model variables can take a continuity of values gives sense to allowing continuous states domains. The possibility of having continuous action spaces links the time to the associated action. It allows us to have models with general temporal distributions of transition times. But, as a consequence of this extension, the computations become more difficult.

The main analysis difference is the way to measure probability of executions, which appears the notion of *measure space of transitions* and the scheduler definition is adapted to the continuous case.

Definition 2.17 Measure space of transitions. *Given the set* $T = Q \times A \times Dist(Q, F_Q)$, *it is defined the* $\sigma-$*algebra* $F_T = F_Q \otimes F_A \otimes F_{Dist(Q,F_Q)}$. *Thus,* (T, F_T) *is a measure space.*

F_T is the *set of measurable transitions*. It is formed by the product of measurable sets of states, actions and probability measures that characterize the transitions. The set of *transitions enabled from a state* q is $T(q) = \{(r, a, \mu) \in \longrightarrow \text{with} r = q\}$.

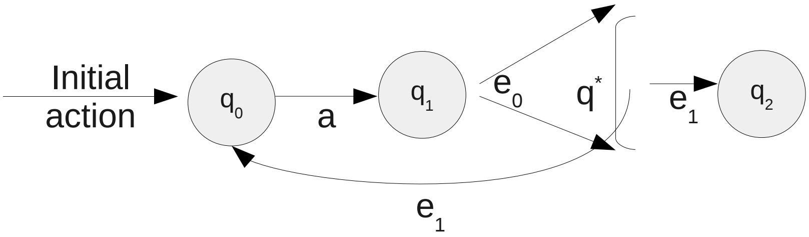


Figure 2.5: Stochastic Transition system with continuous space of states. The action e of our old example (Figure 2.2) was decomposed into two actions e_0 and e_1. The action e_0 has an infinity of possible transitions: any $q^* \in \{\{A = 1, N = 1, G \in [0, 1]\}$ with G given by$\mu\}$.

In Figure 2.5 we show a new stochastic transition system, in this example the action e of our old example (Figure 2.2) is decomposed into two actions e_0 and e_1. Let us consider that the states have the following interpretations:

- $q_0 = \big\{A = 0, N = 1, G \in [0, 1]\big\}$
- $q_1 = \big\{A = 1, N = 1, G \in [0, 1]\big\}$
- $q^* \in \big\{\{A = 1, N = 1, G \in [0, 1]\}$, with G given by the measure $\mu\big\}$
- $q_2 = \big\{A = 1, N = 2, G \in [0, 1]\big\}$

So, the values of G are chosen by the probability measure μ. We have that

$$P(q_2|q_1) = \int_{[0,1]} P(q_2|G)\mu(dG)$$

Let suppose that the action e_1 has the following behavior (with V a known subset of $[0, 1]$):

$$P\big(q_2 \mid q^* = \{A = 1, N = 1, G \in [0, 1]\}\big) = \begin{cases} 1 \text{ if } G \in V \\ 0 \text{ otherwise,} \end{cases}$$

With this, $P(q_2|q_1) = \int_V \mu(dG) = \mu(V)$, and we need V measurable in $[0, 1]$.

As in discrete case, the way to solve non-determinism is by using the scheduler notion (definition 2.18). Given a finite execution (where the time can be included in selected actions), it returns a probability measure over the set of probabilistic transitions T enabled from the last visited state.

Definition 2.18 Scheduler (for STSs with continuous spaces). *$D : Exec^* \longrightarrow Dist(T, F_T)$ such that $\forall \sigma \in Exec^*, supp(D(\sigma)) = T(last(\sigma))$*

2.2.6 Measurability of schedulers and executions

In the previous section (2.2.5) we showed evidence for the need of including notions of measurability to analyze schedulers and executions. It is not possible to compute probabilities to any scheduler and execution. It is necessary to define *measurable schedulers*, *measurable probabilistic executions* and the relations between these elements. Although the discrete case of [113] can be seen as a sub case of the general stochastic transition systems of [25], for simplicity we will explain in detail the discrete approach.

When one considers that schedulers can depend on previous transition times, it is necessary to impose measurability conditions to schedulers to be capable of computing probabilities of executions or timed executions (see section 2.2.5). With definition 2.19, the scheduler D_2 in previous sub-section is measurable if and only if V is measurable in $\mathbb{R}^+$ (*i.e.* iff $V \in$ Borel).

Definition 2.19 Measurability of timed history-dependent schedulers. *One will say that a timed history-dependent scheduler is measurable if* $\forall L \in F_A, D(\cdot, L) : Exec^* \longrightarrow [0,1]$ *(where* $D(\cdot, L)$ *is defined by* $D(\sigma, L) = D(\sigma)(L)$*) is a measurable function.*

To define measurability of timed executions, one starts by defining the measurable sets of a given finite length, the measurable sets of any length and finally one joins the sets (definition 2.20).

Definition 2.20 Measurability on sets of timed executions.

- *As* Q *and* A *are considered to be finite, for them one uses the powerset* $\sigma-$*algebras. To measure the time it is considered the Borel* $\sigma-$*algebra.*

- *Thus, for the timed executions with length* n*, one generates* $F_{\Sigma^n} = F_{Q\times(A\times\mathbb{R}^+\times Q)^n}$ *with the rectangles* $Q_0 \times M_1 \times \ldots \times M_n$*, where* $Q_0 \in F_Q, M_i \in F_{A\times\mathbb{R}^+\times Q}$*.*

- *The generators of* F_{Σ^w} *are the cylinders with base a measurable rectangle in* Σ^k*, with* $k \in \mathbb{N}$*. It is to say, it is generated by the sets* $R(\Lambda) = \{\sigma \in \Sigma^w : \sigma^k \in R\}$ *where* R *is a rectangle in* F_Σ^k *and* σ^k *is the prefix of length* k *of* σ*.*

- *Finally, for the arbitrary timed executions it is used the standard* $\sigma-$*algebra (*F_Σ*) formed by the disjoint union of the families* $\{F_{\Sigma^n}\}_{n\in\mathbb{N}}$ *and* F_{Σ^w}*.*

In the case of *Stochastic Transition Systems with continuous spaces* ([25]), the notion of measurability of executions is similar, but measurability of schedulers is a little more difficult to understand (definition 2.21).

Definition 2.21 Measurability on set of executions with continuous spaces of states and actions.

- *As* Q *and* A *are considered to be continuous, one considers* $\Lambda = Q_0 \times A_1 \times Q_1 \times \ldots \times A_n \times Q_n$ *the sequences of measurable sets (*$Q_i \in F_Q$*,* $A_j \in F_A$ *with* $i \in \{0, \ldots, n\}, j \in \{1, \ldots, n\}$*).*

- *The set of finite executions F_{Exec^*} is the $\sigma-$algebra generated by all the the sets Λ, with $n \in \mathbb{N}$.*
- *Thus, the cylinders with base Λ are defined as*

  $C_\Lambda = \{q_0a_1q_1 \dots a_nq_n\alpha\} : \forall i \in \{0, \dots, n\}q_i \in Q_i, \forall i \in \{1, \dots, n\}a_i \in A_i, \alpha \in Exec$

- *The set of measurable set of executions F_{Exec} is the $\sigma-$algebra generated by C_Λ.*

To clarify concepts, let us consider the example of Figure 2.5 (section 2.2.5) with the successor of q^* chosen by a scheduler D_3 defined such that:

$$D_3(\dots q^*) = \begin{cases} q_2 \text{ if } q^* \in V \\ q_0 \text{ otherwise,} \end{cases}$$

where V is a subset of Q.

For this scheduler, the probability $P(q_2|q_1) = \int_V \mu(dG) = \mu(V)$, and we need V measurable in $[0, 1]$ to be capable to compute it.

One defines measurability of *Scheduler for Stochastic Transition Systems with continuous spaces* to relate measurable executions with measurable schedulers, generalizing the example.

Definition 2.22 Measurability of schedulers, flattening of a scheduler.

- *Given a scheduler (for Stochastic Transition Systems with continuous spaces) D, and $\sigma \in Exec^*$ a finite execution. The probability measure $\mu_{D(\sigma)}$ over $((A \times Q), F_A \otimes F_Q)$ defined $\forall A \in F_A, X \in F_Q$ by:*

  $$\mu_{D(\sigma)}(A \times X) = \int_{(q,a,\mu)\in T} 1_A(a)\mu(X)dD(\sigma),$$

  is called the flattening of D in σ.

- *A scheduler is measurable if its flattening is a measurable as function from $(Exec^*, F_{Exec^*})$ to $(Dist(A \times Q), F_{Dist(A\times Q)})$*

The flattening corresponds to the combined transition probability of actions and states to continue the executions. That is to say, it measures the probability of continuing with each set in $(F_A \otimes F_Q)$ (actions and probabilities) after the execution σ and with the scheduler D.

In our example, $\mu_{D_3(\cdot)}$ is a measurable function if and only if

$$\mu^{-1}_{D_3(\cdot)}(1_V(\{q_2\})) = V \text{ is measurable.}$$

Thus, D_3 is a measurable scheduler if and only if V is measurable.

To link measurable schedulers with measurable executions, one introduces the notion of *probabilistic execution* and its measurability.

Definition 2.23 *Probabilistic execution and measurability.*

- *Given an stochastic transition system (with continuous spaces) S and D a scheduler. The probabilistic execution $P_{S,D}$ for S and D is the tuple $(Exec^*, F_{A\times Q}, \mu)$, where $\mu : Exec^* \times F_{A\times Q} \longrightarrow [0,1]$ such that for each $\sigma \in Exec^*$ $\mu(\sigma, \cdot)$ defined by $\mu_{D(\sigma)}$ is a probability measure over $A \times Q$.*

- *A probabilistic execution $P_{S,D} = (Exec^*, F_{A\times Q}, \mu)$ is measurable if $\mu(\cdot, X)$ is a measurable function for each $X \in F_{A\times Q}$.*

Consequently, a probabilistic execution defines the transitions probabilities induced by a scheduler and a finite execution.

In this case, from the measurability definitions one obtains the following measurability relation:

Theorem 2.1 *Given an stochastic transition system (with continuous spaces) S, and D a scheduler. D is measurable if an only if $P_{S,D}$ is measurable.*

2.3 Composition of models

The composition of modules is based on two elements: synchronization of events and input-output relations. The first element is associated with the stochastic transition system (section 2.2) that describes the transitions of the hybrid system. By synchronizing two events we associate probabilistic modules by imposing that the occurrence of an specific event in one module coincides with the occurrence of another event in the other one. Given two modules, we use input-output connections to impose that the output of one module is input of other one. So, updates of the output imply updates of the input.

2.3.1 Composition of probabilistic modules

Let's return to the previous section 2.2. The presence of sources of non-determinism allows that Stochastic Transition Systems can grow by joining modules to obtain more complete interacting models. As we shown in our example (Figure 2.2 and 2.2.2), actions can depend on external variables not controlled by the module (action s, variable G). The dynamics of external variables can be considered to be modeled by another module.

As example, we consider that the dynamics of the variable G is given by the module Q (Figure 2.6), whose actions are:

- *initial action*: $\varnothing \longrightarrow g_0 = \{G := 0\} \vee g_1 = \{G := 0.5\} \vee g_2 = \{G := 1\}$ with probabilities $\frac{1}{3}$,

- $g : \{G\} \longrightarrow \{G := G + 0.5\}$,

The module Q controls the behavior of G, the growth of the cell.

We connect the modules P and Q, modeling division, by using the notion of *composition* (definition 2.24, [30]).

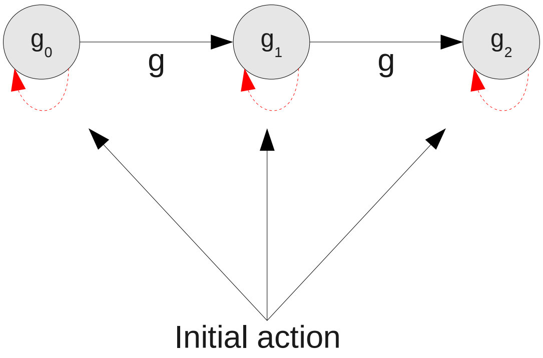


Figure 2.6: *Stochastic transition system* that represents the behavior of G (growth of cell). It is formed by the states: g_0, g_1 and g_2. The initial action selects with the same probability any state, the action g relates the states.

Definition 2.24 Composition. *The composition of two stochastic transition systems P and Q, denoted $P||Q$, is the stochastic transition system that models the probabilistic behavior of P and Q that interact synchronously.*

For *probabilistic modules* (where states are interpreted as valuations of variables and actions as probabilistic changes of its values, see section 2.2.2), the composition has to be consistent with respect to the mutually visible variables. It is to say, in a execution of $P||Q$, the interface variables of P that are external variables of Q (and vice verse) must agree to have unique values per each observable variable.

The modules *can be composed* if they are consistent with respect to the types of variables. These facts are summarized in the following affirmations:

- Two modules, P and Q, can be composed if:
  1. $ext(P) \cup intf(P) = ext(Q) \cup int(Q)$,
  2. $intf(P) \cap intf(Q) = \varnothing$,
  3. $priv(P) \cap V(Q) = \varnothing$,
  4. $priv(Q) \cap V(P) = \varnothing$.
- If they can be composed, then $P||Q$ is a module such that:
  1. $ext(P||Q) = (ext(P) \cup ext(Q)) \setminus intf(P||Q)$,
  2. $intf(P||Q) = intf(P) \cup intf(Q)$,
  3. $priv(P||Q) = priv(P) \cup priv(Q)$,
  4. $atoms(P||Q) = atoms(P) \cup atoms(Q)$.

The way to represent the behavior of a system is by obtaining the values of its answers, but it is not possible to use only the notion of execution. As stochastic transition systems include randomness and non-determinism, the number of different executions can be infinite. To add the stochastic element to the behavior characterization, one defines the notion of *bundle* ([30]).

Definition 2.25 Bundles and semantics of probabilistic modules.

- *A bundle of length n is a probability distribution over $Q-$traces of length n.*
- *It is defined the semantics of a module P, denoted $[[P]]$, as the set of bundles associated to the module.*

Each bundle records the outcome of a particular sequence of randomized choices (answers of randomized schedulers) made by the system. The use of randomized schedulers, that transform non determinism in randomness, makes possible this definition.

One can add the interpretation of the states in the semantics definition (definition 2.26) by replacing $Q-$traces by $V-$traces (with V the set of typed variables), thus bundles are probability over $V-$traces and they are called $V-$bundles.

To define correctly the composition of modules, one considers only the observable variables, then $[[P]]$ is a probabilistic $(extl(P) \cup intf(P)$-language ([30]).

Definition 2.26 Semantics of composition of probabilistic modules. *If two modules P and Q can be composed, the semantics of the composition is the intersection of the both semantics, it is to say*

$$[[P||Q]] = [[P]] \cap [[Q]]$$

.

With definition 2.27, the equality $[[P||Q]] = [[P]] \cap [[Q]] = [[P]] \times [[Q]]$ defines the composition as $(V_P \cup V_Q)-$language.

Definition 2.27 Product of bundles. *Given two typed set of variables, V_P and V_Q for the modules P and Q respectively. Two bundles b_P and b_Q, for P and Q respectively, can be multiplied if $b_P[V_P \cap V_Q] = b_Q[V_P \cap V_Q]$ (i.e. the probabilities coincide for $V_P \cap V_Q-$traces).*

The product of b_P and b_Q bundles of length n, $b_P \times b_Q$, is a $(V_P \cup V_Q)-$bundle of length n such that $\forall (V_P \cup V_Q)-$trace t of length n:

$$(b_P \times b_Q)(t) = \frac{b_P(t[V_P]) \cdot b_Q(t[V_Q])}{b_P(t[V_P \cap V_Q])}$$

An extension of our model of cell division is obtained by considering the modules P and Q (Figure 2.7), to allow K divisions of the cell.

We defined the concept of composition of modules that allows us to join different models. However, in that notion do not exist restrictions about what actions are compatible. If a transitions happens in a module any possible transition is allowed in the other one, but often actions of different systems are associated. In our example, a cell must grow if it is not capable of dividing, consequently the action g in Q happens if the action s occurs in P. This fact is included by defining the *Synchronized product of Transition Systems* ([6, 24], definition 2.28).

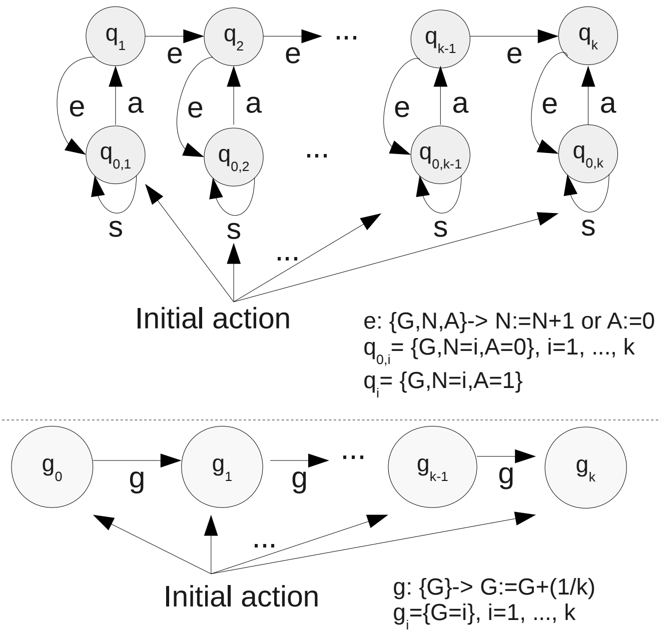


Figure 2.7: Two *Transition systems* that at composing them represents the cell division process as depending on a growth index. Extension of models of Figure 2.2 and 2.6. We do not show the idle transitions at each state.

Definition 2.28 Synchronized product of Transition Systems. *Given n sets of actions $A_1, \dots, A_n$, and a synchronization constraint $I \subset A_1 \times \dots \times A_n$. Let us $TS_1, TS_2, \dots TS_n$ transition systems with states sets Q_i, actions sets A_i, initial states Q_0^i and transitions $\longrightarrow_i$ for each $i \in \{1, \dots, n\}$. The synchronized product of $TS_1, \dots, TS_n$ is the transition system $\big((Q_1 \times \dots Q_n), (Q_0^1 \times Q_0^n), (A_1, \dots, A_n), \longrightarrow\big)$, defined by the following equivalence:*

$$(q_1, \dots, q_n) \xrightarrow{(a_1,\dots,a_n)} (r_1, \dots, r_n) \Longleftrightarrow (a_1, \dots, a_n) \in I \wedge (\forall 1 \leqslant i \leqslant n,\ q_i \xrightarrow{a_i} r_i)$$

With these composition elements one can build models formed by many interacting components. These ideas give the primary support to develop hierarchical models with different complexity levels. The inclusion of randomness and non-determinism grants a major level of reality and flexibility to the models, to consider different probable answers of the system and to improve models by composing them and synchronizing actions. The technical way to describe the concept of improvement of models is called *refinement*. One wants that the refinement relation behaves well on compositions. That is to say, be capable of proving refinement of composed models by checking refinement properties on the components. This is achieved by having *assume-guarantee* rules for composition under refinements. The definition of composition of probabilistic modules satisfies a very strong rule.

The refinement of a probabilistic module intuitively corresponds to the specification of the model. For example non-determinism reductions by associating probabilities to actions refine the module. A refinement reduces the set of possible behaviors of the system.

Definition 2.29 Refinement of modules. *One says that a probabilistic module P refines another module P^* if the following properties are satisfied:*

1. $intf(P^*) \subseteq intf(P)$, $extl(P^*) \subseteq extl(P) \cup intf(P)$,
2. *Given typed variables* $v_1, v_2 \in intf(P^*)$, *if an atom* $A \in Atoms(P)$ *such that* $v_1, v_2 \in ctr(A)$ *then* $\exists A^* \in Atoms(P^*)$ *such that* $v_1, v_2 \in ctr(A^*)$,
3. $\forall u \in intf(P^*), v \in intf(P^*) \cup extl(P^*)$, *if* $\exists A \in Atoms(P) : u \in ctr(A), v \in read(A)$ *then* $\exists A^* \in Atoms(P^*) : u \in ctr(A^*), v \in read(A^*)$,
4. *Bundle containment:* $[[P]]$ *is contained* $(\subseteq)$ *in* $[[P^*]]$.

The three first properties are structural, the last one relates the behaviors.

It is clear that an stochastic transition system is a refinement of a transition system. Solving non-determinism of a module by adding randomness schedulers to the module is a way to refine, as is adding composition or synchronization of two modules. One can prove ([30]) the validity of an circular *assume-guarantee* rule for probabilistic modules given by the following assertion:

$$(P||Q^* \preceq Q, \quad Q||P^* \preceq Q^*) \Longrightarrow P||P^* \preceq Q||Q^*$$

The utility of this rule is to prove that a composed module refines another when the module is composed by many components. Consequently, to check that the composition of two modules $Q||Q^*$ is refined by $P||P^*$, it is sufficient to check refinements of the components of the two original modules Q and Q^*. In particular, this says that to obtain a refinement of $Q||Q^*$ one must find a component P that, composed with Q^*, refines Q and other component P^* that composed with Q refines Q^*.

2.3.2 Input-output connections

A natural way to integrate two models is by using the outputs of one as inputs of another one. It is useful to decompose extensive systems, or to connect processes *a priori* separated. In application sections (5 and 6), we will see how to use these relations to build reconciled models (case of wine fermentation kinetics) and to connect regulatory systems to control cell fate.

As example, if we restrict the physiological circulatory model by Guyton ([45, 46, 106], Figure 1.1) to those modules of the renin-angiotensin-aldosterone system (RAAS) associated to renal control of the blood pressure (crucial for blood circulation [91]), we have the input-output connections in Figure 2.8.

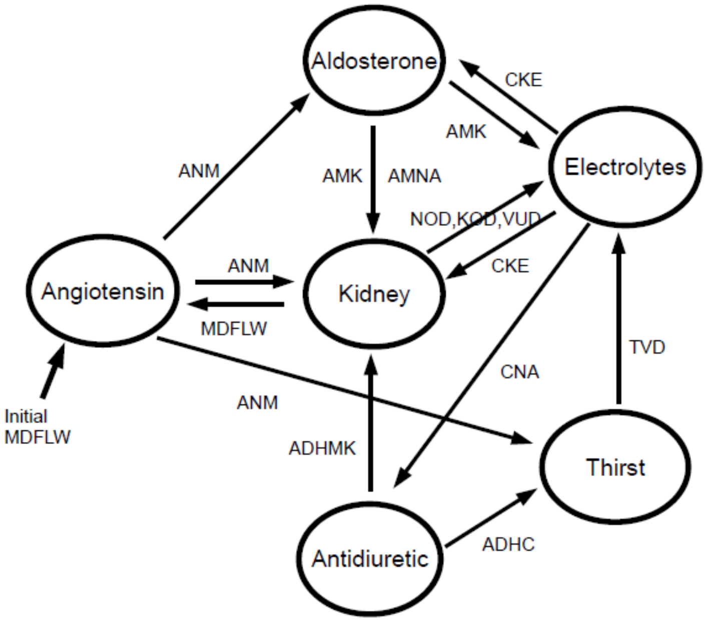


Figure 2.8: Guyton model [46] restricted to the renin-angiotensin-aldosterone system. Systems of differential equations, DAEs and other models are used in each module, and they are connected by input-output relations. Rates variables: $MDFLW$ (fluid in the renal tubules at macula densa), TVD (fluid intake), NOD, KOD, VUD (urine output), ADH (rate of hormone entry into the fluids). Concentrations: CKE, CNA, $ADHC$ (antidiuretic hormone in the blood). Multiplier effects: ANM (angiotensin), AMK, $AMNA$, $ADHMK$.

Chapter 3

Our approach: Modeling biological systems

Here we present our approach to describe complex biological systems by hybrid systems and composition of models, with the goal of modeling by reuse, reconciliation and composition of models.

We formalize the theory of hybrid systems to combine continuous and discrete models, with the inclusion of stochastic and non-deterministic behaviors, and to allow two types of descriptions: with coefficient switches and with strong switches (section 3.3). The implementations of these descriptions with the new improved version of the BioRica framework are presented in the next chapter (4). We finish by summarizing the main contributions of this work.

3.1 Reusing and reconciling models by composition

An important need to develop science is to be able to reuse it. The advance of science is based on the reuse, the application and the improvements of the scientific discoveries. In the last years research on reusability of models is vigorous ([109]), searching how to define and simulate composed models in an unambiguous way is the goal we are approaching now.

By accepting that biological systems are modular (section 1.2), the composition of different modules allows us to build complete descriptions. Modularity allows us, between many things, to connect regulation and controlled models.

Here, we compose models by connecting them by events synchronizations or input-output relations. Event synchronizations connect the discrete dynamics of the models, that is to say the stochastic transition systems. The composition of such systems, explained in section 2.3.1, allow us to combine different discrete dynamics and to synchronize transitions. The input-output connections allow us to separate extensive continuous models into simpler models to solve them easier, or to combine the models of different processes to describe more general behaviors.

When a scientific area is attractive it is common that different models are developed, each one constructed to meet particular needs and to fit specific data. We call reconciliation of models the case in which different models independently developed to describe a system, are analyzed to merge the best between them according to observable factors.

The reconciliation process has three steps. In the first step we have the original models, which one can execute separately. In the second step, we can execute them in parallel using a single platform, and in the third we can control which model we use according to the factor conditions, for obtaining the best agreements to reality. For each system, we implement the last step of the reconciliation process in different ways in function of the strength of the transformations induced by the switches. If it is possible to model those transformations as changes in the values of coefficients, we rewrite all the possible models in a same base and identify common elements. In particular for ODE models, we rewrite the models into polynomial form. When the changes are stronger than changes in coefficients, we use our approach with strong switches in the continuous dynamics.

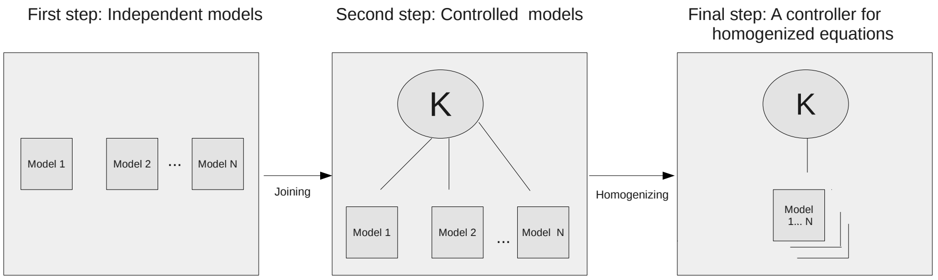

First step: Independent models
Model 1
Model 2
...
Model N
Joining
Second step: Controlled models
K
Model 1
Model 2
...
Model N
Homogenizing
Final step: A controller for homogenized equations
K
Model 1... N

Figure 3.1: Reconciliation process. First we have the original models, then we control them to run them in parallel, finally we control the selection of models to chose the best one in function of factor conditions.

The diversity of models to explain connected processes makes it necessary to define theory and tools to integrate them in a composed model. To compose models we use the non-ambiguous semantics of BioRica, which is based on the AltaRica framework ([7, 8]) and the use of Stochastic Transition Systems ([29]).

3.2 Modeling biological systems as Hybrid systems

Due to the special conditions that one observes in biological systems: complexity of behaviors, dependence of environmental conditions, and modularity (section 1.3) we approach biological systems from the perspective of hybrid systems. We decompose the system into different modules to represent individual processes. Each module is separately modeled, and we model the interactions between them by composition.

In practice, the modeling process follows both directions. One starts by modeling individual biological processes, which can be modeled by decomposing into simpler modules and can be composed with other modules to describe more general processes. Each module is described by a model. Discrete controllers and controlled models define Hybrid systems, and the composition of all these models is itself a hybrid system (section 3.4).

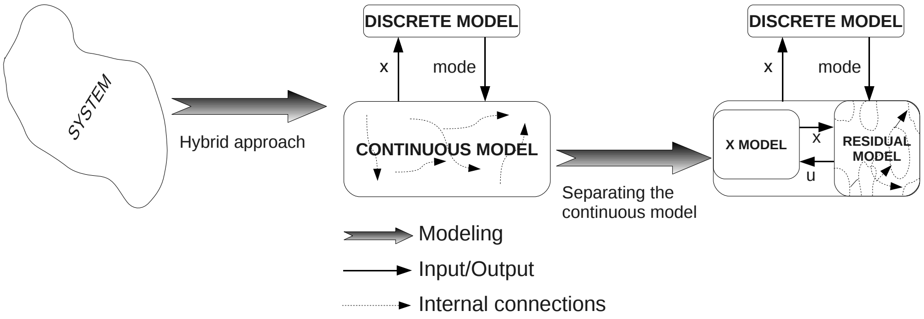


Figure 3.2: Modeling schema of Complex Biological Systems using Hybrid models. First one identifies the discrete and continuous interacting dynamics, then one separates the continuous dynamics into two interacting models: the *X MODEL* that describes the dynamics of X, and the residual model.

The possibility of composing models gives a way to improve and extend models, to obtain more and more complete descriptions of biological processes depending on the level of details that is wished and the quantity of available information. In absence of information, one can build models with open specifications by defining stochastic or non-determinist modules.

This approach allows us to combine different types of models to describe complex processes and to represent behavior changes provoked by environmental conditions. In particular, given a set of available models we can choose the 'best' model according to the environmental conditions (section 5), and we can combine signaling pathways with Gene Regulatory Networks as in case of our model of cell fate decisions to form bone or not (section 6).

3.3 Describing hybrid systems

As we explained before (section 2.1), Hybrid Systems can be seen from two points of view: function ([18, 100], [104]) and implementation ([50, 60]). The first one, called Switched Systems, focuses in human comprehension, the second type is more general and focuses in automatic interpretation. This second approach uses tools of Automata theory and is easy to understand in terms of Transition systems theory ([7, 29]).

Our approach uses elements of both approaches. Most of continuous models are given by sets of ordinary differential equations, but we allow other type of continuous models, which can be specified externally by using the SBML specification ([56]).

Continuous variables evolve according to continuous models, but at any time mode changes can change the definition of the continuous model. These changes are called *mode transitions* and are considered to be transitions in the sense of Transition Systems theory. Given an action producing a mode transition, the next mode is chosen according to transition probabilities and system schedule laws. The conditions that provoke mode transitions are called *guards*. For each mode, the system evolves in function of its continuous dynamics.

Our use of Stochastic transition systems ([29]) to describe mode transitions, allows us to extend the notion of Hybrid system (section 2.1) to include stochastic and non-deterministic behaviors. To model hybrid systems, we consider that the continuous dynamics is given by ordinary differential equations or other models, while the discrete dynamics is described by Stochastic transition systems as defined in section 2.2 with general temporal distributions for mode transitions (not only exponential ones). Using modular representation is possible to include logic based functions to define hybrid systems. Input/output connections, switches, control and hierarchical relations relate modules.

3.3.1 Formalization

According to the theory of *stochastic transition systems* (section 2.2), the transitions may have stochastic and non-deterministic behaviors. Given an action producing a transition, in this case a mode change, the following mode is chosen randomly according to transition probabilities.

Formally, the model is represented as systems whose continuous model has transitions given by the following equations 3.1,3.2, and 3.3 above:

At any time t, we denote $P(ev|(x(t), u(t), mode(t))$ the probability of choosing the event ev when the values of the state variables x are $x(t)$, the values of the continuous control variables u are $u(t)$ and the value of the mode variable $mode$ is $mode(t)$; $time((x(t), u(t), mode(t)), ev)$ denotes the delay time of the event ev that is modeled to have distribution $Dist_{ev}\{p_{event,1}, \ldots, p_{event,m}\}$.

$$P(ev|(x(t), u(t), mode(t)) = \frac{w_{ev}}{\sum\limits_{e \in A(x(t),u(t),mode(t))} w_e}, \tag{3.1}$$

$$time((x(t), u(t), mode(t)), ev) \sim Dist_{ev}\{p_{event,1}, \ldots, p_{event,m}\}, \tag{3.2}$$

where w_e is the probability weight assigned to the event e, and $A(x(t), u(t), mode(t))$ is the set of available actions when x takes the value $x(t)$, u the values $u(t)$ and the mode variable $mode$ the value $mode(t)$ (equation 3.3).

$$A(x(t), u(t), mode(t)) = \{ev \in EVENTS : G_{ev}(x(t), u(t), mode(t)) = TRUE\} \tag{3.3}$$

and $EVENTS$ is the set of events considered.

3.3.2 Two types of hybrid systems: with coefficient switches and with strong switches

Systems are described here as Hybrid systems, whose modules are described by models. We consider two classes of switches: coefficient switches and strong switches.

We call Hybrid system with coefficient switches those hybrid systems in which the structure of the model is conserved and only specific coefficient values are controlled by the mode transitions. The continuous model is composed of an unique controlled model $X\ MODEL$ that interacts with the controller $CONTROL$ (Figure 3.3*(A)*). We define Hybrid systems with strong switches as hybrid systems that allow different types of models. In such a case, each type of continuous model is represented by two interacting modules: $MODEL\ i$ and $CONTROL_i$ (Figure 3.3*(B)*).

In the case of Switched Systems, where one assumes continuous dynamics based on differential equations, the global dynamics comes from the interaction of the continuous and the discrete dynamics, affecting the differential equations. Mode transitions provoke changes in the continuous dynamics without changing the structure of the module, but only values of coefficients (Figure 3.3*(A)*).

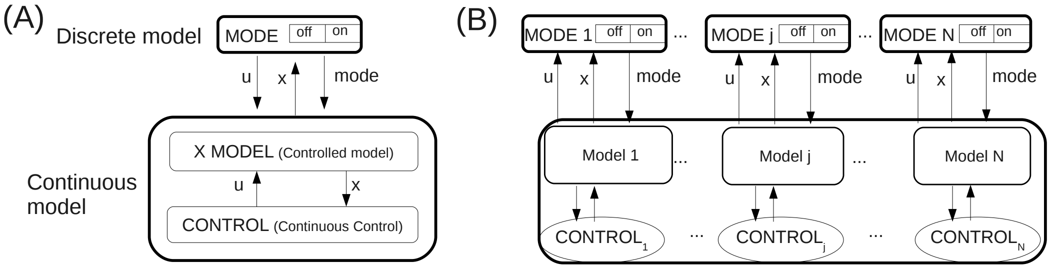


Figure 3.3: Abstraction of Hybrid systems. (A) The module $X\ MODEL$ describes the continuous model of the state variables x. It interacts with two other modules: $CONTROL$ and $MODE$. $CONTROL$ computes the values of the continuous control variables u, $MODE$ decides the values of the mode variables. Arrows denote input-output relations. (B) Strong switches between N models. The $j-$th model is represented by its continuous dynamics $Model\ j$, that interacts with two modules: the controller $CONTROL_j$, and $MODE\ j$ that decides if the model is active or not.

For strong switches, the system takes the form of Figure 3.3*(B)*. By means of the modules $MODE\ i$, the system switches between different modules that represent different types of continuous models. At the same time, each module $Model\ j$ can interact with its continuous controller and evolves over time.

When each continuous dynamics has a different type of formalism, software or type of mathematical description, the possible continuous models can not be uniquely represented considering switches as changes in its coefficients.

When one has access to all the models, considering the models as structurally different or not is a modeling decision. Let's suppose for example that we know that given an specific condition a variable x_i responses linearly with respect to other variable x_j, but in other condition this dependence is quadratic. One must decide to join the models, by linear combination of both dependences and choosing the coefficients in function of the conditions, or to consider them separately.

If one decides to join different models, to use Hybrid systems with coefficient changes, one must rewrite the models to represent them within an unique structure by identifying types of dependences and by grouping them. However, in general cases rewriting is not the best option. The rewriting of models can help to interpret the results, but it consumes time, increases the possibility of mistakes, and it is not practical to have to rewrite models whenever we want to reuse them. Moreover, if one does not have access to one model but to the state variables values computed by it (e.g: executing a software), this option is not possible. Such a model is seen as a black box with a different structure, and the system is modeled with strong switches.

If we consider the approach strong switches (Figure 3.3*(B)*) and group all the mode modules in an unique module $MODE$ and consider the control modules included in each model, this corresponds to the case when many models are used to describe the same system and we reconcile the models into an unique one (Figure 3.4): a reconciled model where module $MODE$ considers the associated mode variable active if the factor values belong to the optimal conditions, chosen according to the calibration of the model or validating on new data. From the definition of Hybrid systems (section 2.1), one observes that the system resulting from reconciling models is a Hybrid system.

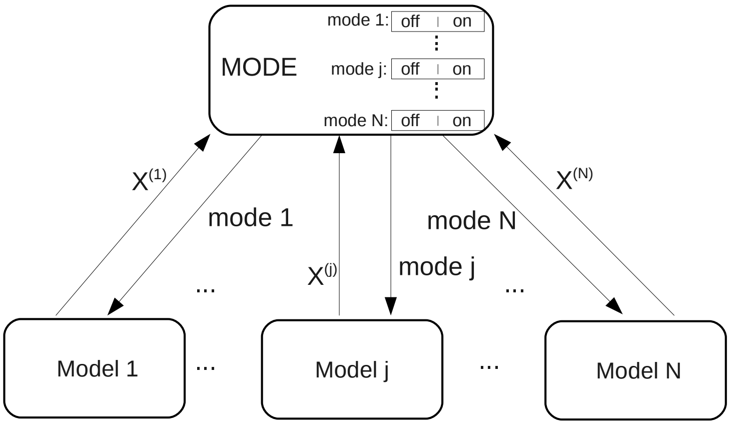


Figure 3.4: Abstraction of the reconciliation of models as hybrid system with strong switches. The $j-$th model is represented by its continuous dynamics $Model\ j$, that interacts with two modules: the controller $CONTROL_j$, and $MODES$ that decides if the model is active or not by means of the mode variable $mode\ j$.

With this approach it is possible to describe Hybrid systems with coefficient switches and with strong switches within a single formalism. This approach is independent of the type of continuous dynamics. The modeler can chose which models to join in an common module and which ones to consider as different modules according to common characteristics and accessibility of the models.

3.4 Guarantees

We obtain some assume-guarantee rules with respect to properties of Hybrid systems and compositions of them.

The stability (in the sense of Lyapunov) of Switched systems is not direct from the stability of each model configuration, one has to study the interactions between the continuous and discrete dynamics. As a result, stability is assured given the conditions of Theorem 6 in [18], which establishes the required properties of the so called switching sequences (initial states values and mode-time pairs of switches):

'existence of Lyapunov-like functions that satisfy the non increasing condition, and mode variables forming a compact set'. Other hybrid systems properties are presented in [50]. It defines decidability properties of verification tasks such as reachability, emptiness and trace inclusion problems on hybrid systems or subsets of them.

The theory presented here allows us to obtain two properties that are closed with respect to the composition of hybrid systems: the definition itself, and the refinement of hybrid systems.

From section 2.3 is clear that the composition of hybrid systems, by event synchronizations or input-output connections, is a hybrid system. Compositions of the associated stochastic transition systems are stochastic transition systems, and the result of joining continuous models by inputs/outputs connections is a continuous model.

As we explained in section 2.3.1, the refinement of a module forming part of a composed system results in a refinement of the composed system. That is to say, if one specifies the behavior of one module by restricting its behavior (its model) as result one specifies the composed model. So, the process of improving models can be carried out by improving the individual models of the composed system.

Our approach assures the consistency of composed models with respect to the state variables. That is to say, given a composed model and a variable that is used by many sub-models, one can control how the changes on this variable are perceived by the sub-models. It is obtained by associating a flow variable to each shared variable, and imposing assertions of equality between the versions of that flow variable on different modules.

3.5 General contributions

In this chapter we have presented the development of an approach with theoretical substance and good results in practice. It allows modeling, implementation and simulation of real biological applications. Concrete examples will be seen in later chapters.

This approach takes its place in themes of current research: composition and reusing of models, and coupling of metabolic pathways and Gene Regulatory Networks. Composition of validated models is considered a way to build complete descriptions of complex biological systems ([109, 33]). Our approach and the BioRica framework allow us to combine different types of models, by implementing the notion of composition and importing SBML models. Metabolic coupling in Gene Regulatory Networks ([12]) has been considered as a way to obtain indirect interaction between genes, to obtain better descriptions of underlying behaviors.

We further considered formalization of the Hybrid Systems theory by using Stochastic Transition Systems ([29]) to describe stochastic and non-deterministic changes of the continuous dynamics. This allows us to include general temporal distributions of mode transitions, which are responsible of model changes according to two possible types of transformations: coefficient switches and strong switches. For coefficient switches the structure of the model is conserved, and mode transitions provoke changes in their coefficients. For strong switches, the system switches between different modules that represent different types of continuous models. With this theory one can consider different levels of model changes according to the modeling decisions, coefficient switches are done by modifying the models, while with strong switches the models are directly reused.

We obtain some assume-guarantee rules with respect to properties of Hybrid systems and compositions of them. The stability (in the sense of Lyapunov) of Switched systems is assured given the conditions of Theorem 6 in [18]. Compositions of Hybrid systems are Hybrid systems, and refining of a module forming part of a composed system results in a refinement of the composed system. The consistency of composed models, with respect to shared variables, is assured too. More details in section 3.4.

We have identified implementation challenges: stiffness, different timescales, and transitions happening in not considered times; and we have proposed solutions: the current use of modules with independent solvers, and the future possibility of including solvers based on the QSS (Quantized States Systems, [68]) approach.

The development of the theoretical basis of our approach was complemented with the improving of the BioRica framework. We identified problems of the previous BioRica version, and proposed solutions that contributed to the development of important improvements included in the current version with the aim of modeling real biological systems, composing models and allowing hybrid systems. More details can be seen in section 4.3.2.

Chapter 4

Our implementation: Simulating biological systems

In the previous section (3), we explained how to describe complex biological systems by using hybrid systems. This theory allows us to describe interaction of different processes, which are separately modeled by using different types of dynamics. The inclusion of hierarchical relations, combination, reusing, and reconciliation of models gives the possibility of building complete descriptions of biological systems.

Here we focus on the implementation of these ideas and theories, we go from the theory to the practice. To take advantage of all the qualities of the theory, the implementation must allows general definitions of hybrid systems, and reusing existing models. We need a framework able to solve systems with continuos, differential equations systems, and discrete interacting dynamics.

We start by discussing the requirements of the implementation, and describing the different steps to solve hybrid systems to obtain a general schema of the solution process. After that, we argue our decision of implementing these steps with the BioRica framework, which we contributed to improve and extend to work with hybrid systems.

4.1 Requirements of the implementation

We describe the main requirements are necessary to implement our approach. Some needs are related with the way of solving and coding hybrid systems, others with the possibility of composing and reusing models.

1. Able to solve continuous models. In particular, we need to solve systems of differential equations with different complexity degrees (stiffness, section 1.2.2) and work with different timescales.

2. Able to solve discrete models given by stochastic transition systems (2.2), which are used by our definition of hybrid systems. We require define mode transitions with inclusion of stochastic and non-deterministic decisions, which are associated with the chosen transition, action and the spent time.

3. Previous points are related with the ability to solve a wide range of models. In addition, we must be able to use them in the same framework. The framework must be capable of integrating different types of solvers, which are adapted to solve specific models, without being a compendium of different programming languages.

4. Able to implement composition. The combination of different types of models by means of input-output connections (assertions), event synchronizations is fundamental in our modeling approach.

5. Able to incorporate hierarchical relations between models. It allows us to define control relations between processes, and to go from macroscopic to microscopic descriptions levels.

6. Able to implement reusing of models. The framework must allow the reutilization of validated models without rewriting them.

4.2 Solving hybrid systems

Here we describe the solving process of hybrid system. The process is independent of which framework is used to implement our approach.

Figure 4.1 shows the computation steps to solve a hybrid system with state variables x and mode variables $mode$. The process begins reading inputs: initial conditions IC and parameters values P. The time is initialized to zero, and the initial values of the mode variables $mode(0)$ are computed in function of the input values. Then the initial definition of the model $MODEL(mode(0))$ is obtained. The next steps of the diagram are the main part of the process: solving and updating. At any given time the model is solved, and the model is updated when detecting one of the the M guard conditions associated to each possible new value of $mode$ (denoted by $GUARD(x, mode, u, time)$ in Figure 4.1). The process finishes when $END(x, mode, time)$ is verified and the results are printed.

This general schema (Figure 4.1) allows both types of hybrid systems we defined in section 3. According to the effect of the assignation $MODEL(mode)$ after updating a mode, we can be solving hybrid systems with coefficient switches which just change coefficient values, or with strong switches by changing of model in function of the value of $mode$ (Figure 3.3). As we anticipated, to implement this solution schema we used BioRica.

As we explained in section 3, the type of hybrid system chosen is a modeling decision. Given a system, one uses coefficient switches to conserve a basic structure of the model and using mode transitions to do changes in their coefficients. On the other hand, for strong switches, one considers that the system switches between different types of continuous models to allow the reusing of models. The combination of many models favor the use of the strong switches approach.

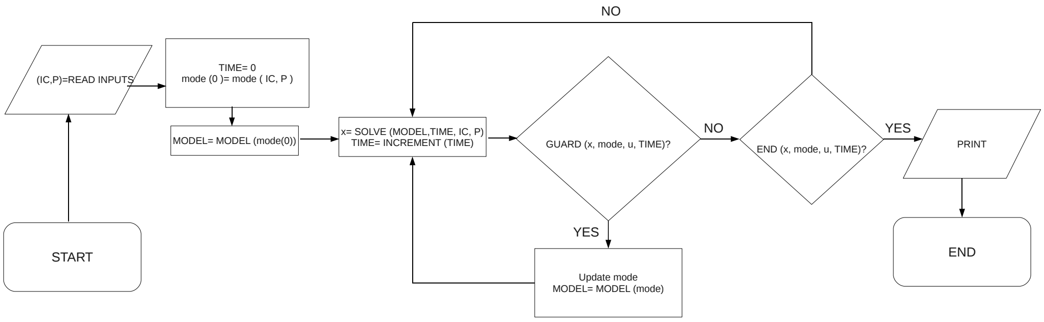


Figure 4.1: Computation steps to solve an hybrid system, options used for wine fermentation shown in red. The process begins reading inputs: initial conditions IC and parameters values P. It is initialized the time in 0, and computed the initial values of the mode variables. Then, one obtains the initial definition of the model $MODEL(mode(0))$. At any time the model is solved, it is updated when detecting the guard condition $GUARD(x, mode, u, time)$, and the process finishes when it is verified $END(x, mode, u, time)$.

4.3 Improving the BioRica framework for simulating biological systems

The implementation of our ideas is based on the BioRica framework. The improved version of BioRica (section 4.4) satisfies the requirements of section 4.1. Modules are described by BioRica nodes, which interact to model the emergent properties of the system. The current version of BioRica was developed by Alice Garcia in the MAGNOME team and is distributed under an open source license. It allows one to describe hybrid systems, solving differential equations and simulating stochastic and non-deterministic behaviors, together with reusing and combining existing models.

4.3.1 The BioRica prototype

The creation of BioRica was motivated by the idea of applying the ideas of AltaRica Dataflow language ([7, 8]) to biological systems. This first extension looks for describing stochastic transition systems (section 2.2, [29]) by incorporating stochastic and timed behaviors, and dataflow links to include composition. A prototype, used internally by the MAGNOME team was developed ([101]).

Each BioRica node is composed by the specification of its fields: state, flow, sub, event, trans, assert, init and extern. In the field *state*, one declares the state variables and their domains. In *flow*, one includes inputs (or outputs) from (to) other BioRica nodes and the field *sub* defines other node objects the node use (hierarchy notion). In *event* one declares the names of the possible actions, and the transitions provoked by these events (when they happen and the effects) are described within the key *trans*. In *assert*, one imposes relations between flows and/or states, and in *init* one defines the initial values of the state variables. Finally, the field *extern* allows the inclusion of external directives about distributions of event delays and priority between events.

BioRica defines a global semantics to compose interacting modules. It is based on the automata semantics of AltaRica ([7, 8]), and the theory of Stochastic Transition Systems ([29]) that allow the inclusion of randomness and non-determinism. Given a BioRica node, one computes the probability of the state dynamics and considers non-deterministic decisions solved by random schedulers. The simulation generates a trace giving the values of the system variables at each times. The resulting semantics it is preserved with respect to flow relations and event synchronizations.

Although they constituted the first theoretical ideas about BioRica as compositional tool for stochastic transition system (requirements 2, 4 and 5 of section 4.1), the usability of this framework would require many improvements and extensions.

4.3.2 Improving BioRica

During this work we have contributed to the development of the new implementation of BioRica by suggesting extensions, discussing implementations, and proveding applications. This version includes important improvements to be capable of modeling real biological systems. We detected the problems of the previous version, proposed and checked possible solutions. As result, with the new version we are capable of describing biological systems as hybrid systems that integrates continuous, discrete, deterministic, stochastic and non-deterministic behaviors. Here we present our main contributions.

- Identification of the need to recode and redesign the BioRica prototype, to build an unified, structured and usable tool of mathematical modeling of biological systems. The implementation of the old version used many different programming languages, and its applicability was limited to discrete models with integer variables.

- Identification and modeling of biological systems not implementable with the BioRica prototype. For discrete systems, we established the need to define transition computations using not only the basic algebraic operations and integer variables. To describe more general systems it was necessary to include continuous elements, which carried to the use of hybrid systems.

- A decision to abandon Matlab as external numerical tool to compute not basic algebraic operations and to solve differential equations. The bridge between the BioRica prototype and Matlab, by means of the engine library of $C++$, worked too slowly and complicated significantly the implementations.

- A decision to adopt the Python libraries to compute general algebraic operations and to solve systems of differential equations, and the possibility to include new libraries to solve other types of equations. The new implementation of BioRica by using Python makes it easy to use all the Python libraries which are available free and have permanent development.

- Modeling and BioRica specification of Hybrid Systems by composing their continuous parts and discrete controllers, assuring integrity of individual models and including hierarchical relations.

- Development of an approach to reconciliate competing models based on Hybrid Systems theory. Modeling and BioRica specification of that approach.
- Development of a general modeling and BioRica implementation of Switched Systems, particular type of Hybrid System in which continuous models are systems of differential equations, by using two types of model changes: coefficient switches and strong switches.
- Proposition of a new BioRica syntax to allow descriptions of general hybrid systems with other continuous models or external implementations. By using other Python libraries one would solve hybrid systems in which the continuous model is described by models such as differential algebraic equations or partial differential equations.

4.4 Implementing and simulating hybrid systems with BioRica

We code and simulate the computations steps needed to solve hybrid systems (Figure 4.1) with BioRica. So, modules are implemented by BioRica nodes. The BioRica syntax is sufficiently general to describe these types of systems with many interacting components, the implementation here explained satisfies all the requirements of section 4.1.

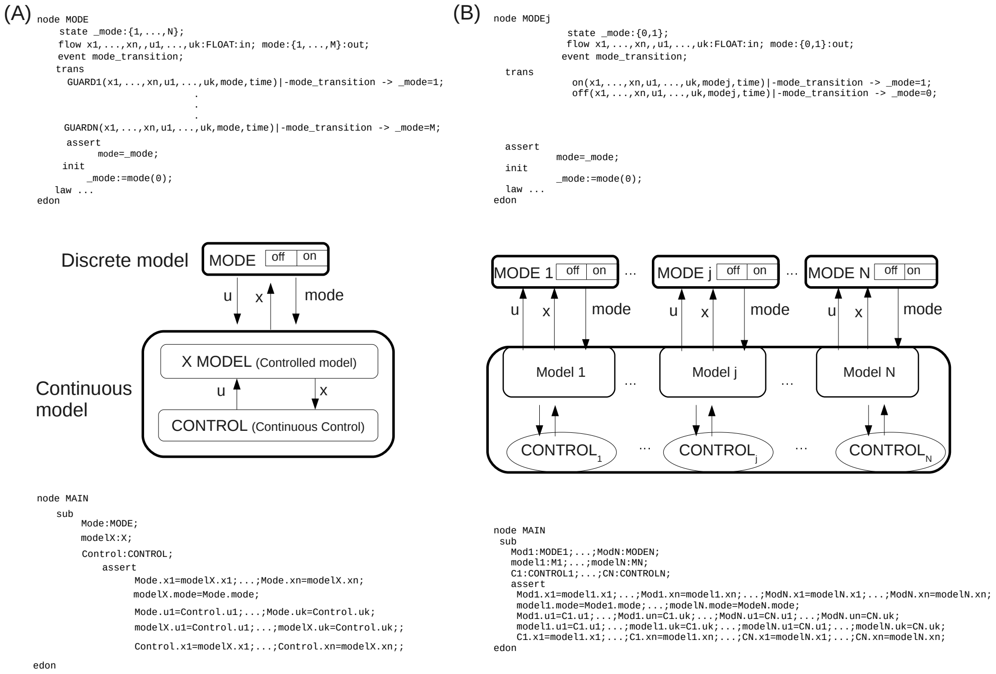
To define BioRica nodes we use the structure of Figure 4.2 that implements the abstractions of Figure 3.3. When mode transitions only change the values of coefficients of the continuous model, we use the approach of Figure 4.2(A). If the changes are stronger than simple changes in coefficients values, we consider the approach of Figure 4.2(B) for strong switches. Each node *Model j* represents a continuous model, while the nodes *MODE i* control if each model is active by deciding the value of the mode variables associated. In case of reconciliation of models, the mode variables are grouped and each continuous model is represented by only one module.

In Figure 4.2 we do not include all the possible fields, such as temporal laws or synchronizations between events. Event synchronizations can be used to reduce the number of assignments in input-output connections, improving the computing time. So, for example for coefficient switches, the node X updates the value of *mode* just if mode transitions happen in the node *MODE* (Figure 4.2). In simple cases, the discrete and continuous dynamics can be included in the same node (Figure 4.3). For see examples of extensive models go to application chapters (sections 5 and 6).

The process can continue. This implementation approach allows the reuse of the combined models. So, the notion of compositions is implemented allowing combination, reusing and reconciling of models.

4.4.1 Syntax and semantics of BioRica for hybrid systems

BioRica allows the interaction between continuous and discrete dynamics. The current version of BioRica considers continuous dynamics given by sets of ordinary



```
node MODE
    state _mode:{1,...,N};
    flow x1,...,xn,,u1,...,uk:FLOAT:in; mode:{1,...,M}:out;
    event mode_transition;
   trans
     GUARD1(x1,...,xn,u1,...,uk,mode,time)|-mode_transition -> _mode=1;
                            .
                            .
                            .
    GUARDN(x1,...,xn,u1,...,uk,mode,time)|-mode_transition -> _mode=M;
     assert
          mode=_mode;
    init
        _mode:=mode(0);
   law ...
edon
```

```
node MODEj
             state _mode:{0,1};
             flow x1,...,xn,,u1,...,uk:FLOAT:in; mode:{0,1}:out;
            event mode_transition;
  trans
              on(x1,...,xn,u1,...,uk,modej,time)|-mode_transition -> _mode=1;
              off(x1,...,xn,u1,...,uk,modej,time)|-mode_transition -> _mode=0;


  assert
           mode=_mode;
  init
           _mode:=mode(0);
  law ...
edon
```

```
node MAIN
   sub
      Mode:MODE;
      modelX:X;
      Control:CONTROL;
          assert
                Mode.x1=modelX.x1;...;Mode.xn=modelX.xn;
                modelX.mode=Mode.mode;
                Mode.u1=Control.u1;...;Mode.uk=Control.uk;
                modelX.u1=Control.u1;...;modelX.uk=Control.uk;;
                Control.x1=modelX.x1;...;Control.xn=modelX.xn;;
edon
```

```
node MAIN
 sub
   Mod1:MODE1;...;ModN:MODEN;
   model1:M1;...;modelN:MN;
   C1:CONTROL1;...;CN:CONTROLN;
   assert
    Mod1.x1=model1.x1;...;Mod1.xn=model1.xn;...;ModN.x1=modelN.x1;...;ModN.xn=modelN.xn;
    model1.mode=Mode1.mode;...;modelN.mode=ModeN.mode;
    Mod1.u1=C1.u1;...;Mod1.un=C1.uk;...;ModN.u1=CN.u1;...;ModN.un=CN.uk;
    model1.u1=C1.u1;...;model1.uk=C1.uk;...;modelN.u1=CN.u1;...;modelN.uk=CN.uk;
    C1.x1=model1.x1;...;C1.xn=model1.xn;...;CN.x1=modelN.x1;...;CN.xn=modelN.xn;
edon
```

Figure 4.2: Implementation of hybrid systems with BioRica. (A) The node X describes the continuous model of the state variables x. It interacts with two other modules: $CONTROL$ and $MODE$. $CONTROL$ computes the values of the continuous control variables u, $MODE$ decides the values of the mode variables. The node $MAIN$ defines their interactions. (B) Strong switches between N continuous models. The $j-$th model is described by the node Mj, that interacts with two modules: the controller $CONTROLj$, and $MODEj$ that decides if the model is active or not. The node $MAIN$ defines their interactions.

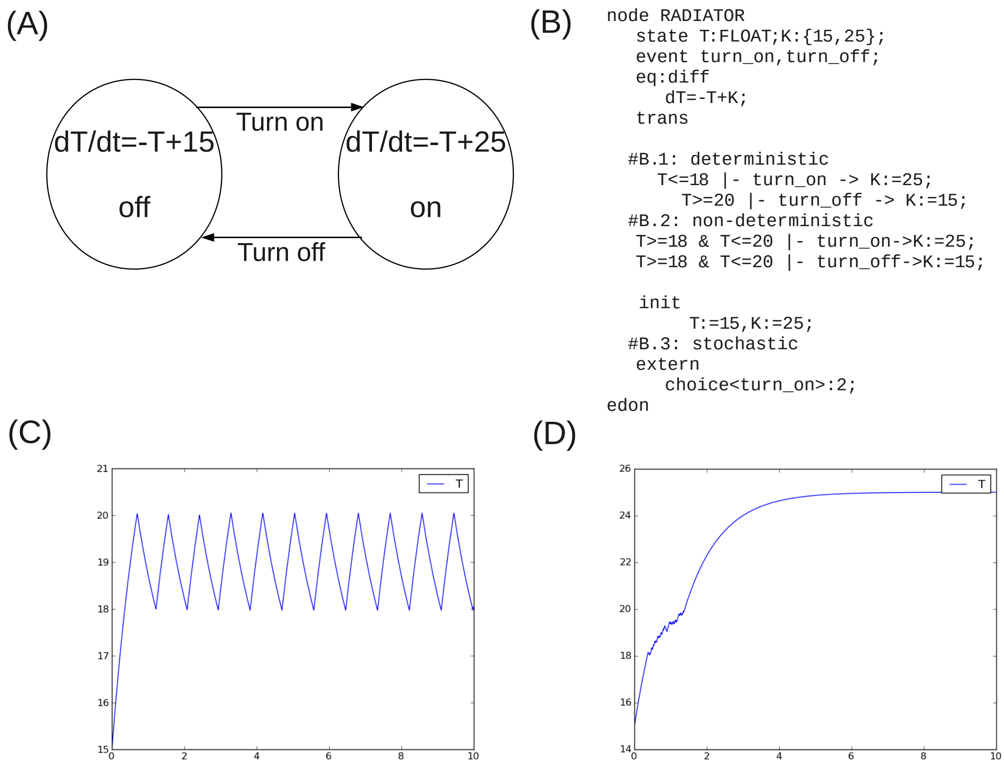
differential equations (switched systems), which are defined in the field *eqdiff*.

So, given a BioRica node (module), BioRica numerically solves the continuous model between transitions, computes the probability of the discrete dynamics and considers non-deterministic decisions solved by random schedulers.

A common specification of biological models is SBML ([56]), maybe the most popular abstraction for biochemical reactions models governed by temporal differential equations. The development by Alice Garcia includes a SBML parser that translates SBML models into BioRica models. So, it is possible to reuse and compose models previously specified in SBML to obtain more general models.

Models are composed by using assertions between flow variables of different nodes (input-output connections), and event synchronizations. The BioRica framework allows us to control the consistency of composed models, and local clocks and solvers allow us to manipulate diverse types of dynamics. Given a composed node and a variable that is used by many sub-nodes, one controls how the changes on this variable are perceived by the sub-nodes by associating to that variable a flow variable, and imposing assertions of equality between the versions of that flow variable on different nodes.

The BioRica compiler reads a specification of the hierarchical model and compiles it into an executable simulator. The compiled code uses the Python runtime environment and can be run stand-alone on most systems. The co-existence of multiple dynamics is assured by a pre-computation of each specified sub-model. Once computed, each part acts as a component that can be queried, but also modified by trajectory modification induced by discrete parts of the model.

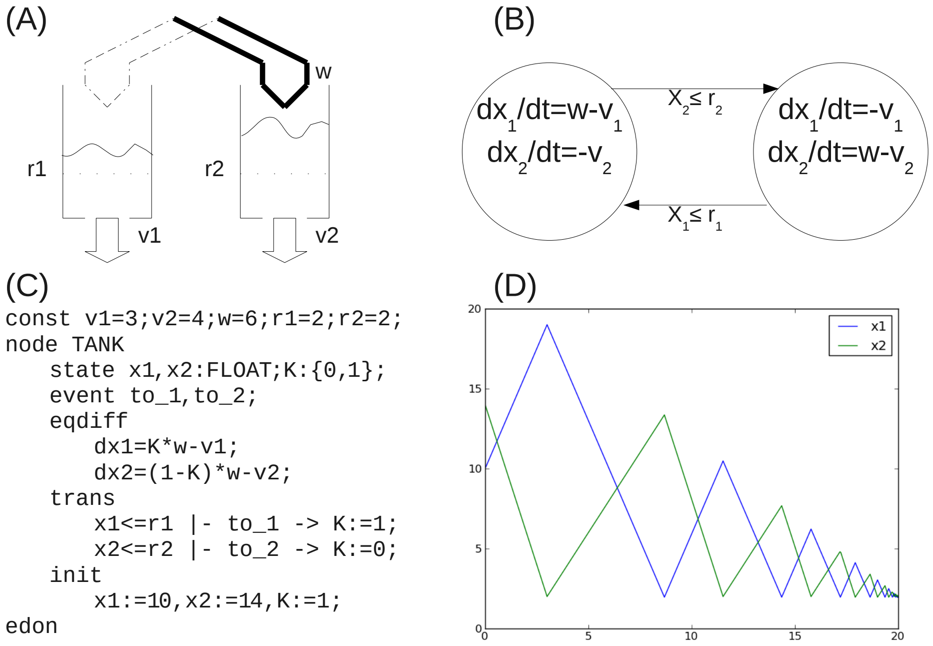


```
node RADIATOR
   state T:FLOAT;K:{15,25};
   event turn_on,turn_off;
   eq:diff
      dT=-T+K;
   trans

  #B.1: deterministic
     T<=18 |- turn_on -> K:=25;
       T>=20 |- turn_off -> K:=15;
  #B.2: non-deterministic
   T>=18 & T<=20 |- turn_on->K:=25;
   T>=18 & T<=20 |- turn_off->K:=15;

   init
        T:=15,K:=25;
  #B.3: stochastic
   extern
      choice<turn_on>:2;
edon
```

Figure 4.3: The radiator. (A)Schema of the Hybrid system. (B)Three models. (B.1) deterministic: it is turned on if the temperature T is lower than $18°C$ and turned off if it is bigger than $20°C$, (B.2) non-deterministic: both events can happen if $18 \leqslant T \leqslant 20$, and (B.3) stochastic giving probability $\frac{2}{3}$ to turn on. (C)Evolution of the temperature for the model (B.1), and (D) for (B.3).

With this approach we can compose processes that have different timescales and complexity levels (stiffness), see section 1.2.2. The use of modules solves in part this problem, each module has an specific timescale and solver, and improve the precision and the computation time. The processes with small timescales are observed at small time steps, while in case of long timescales we use longer time steps to reduce the number of simulations. Stiff equations are strongly manipulated just at the pertinent module.

As example, we consider the behavior of a radiator that controls the temperature of a room. A thermostat is activated when the temperature is detected to be low and it is regulated, if the temperature is high the system is turned off (Figure 4.3). This behavior can be modeled in different ways: deterministic, non-deterministic



```
const v1=3;v2=4;w=6;r1=2;r2=2;
node TANK
    state x1,x2:FLOAT;K:{0,1};
    event to_1,to_2;
    eqdiff
        dx1=K*w-v1;
        dx2=(1-K)*w-v2;
    trans
        x1<=r1 |- to_1 -> K:=1;
        x2<=r2 |- to_2 -> K:=0;
    init
        x1:=10,x2:=14,K:=1;
edon
```

Figure 4.4: (A) Diagram of a Zeno's water tank. (B) Switched model of the behavior. (C) BioRica code and (D) Simulation. The rule to decide into which tank water must be sent considers that the water level of tank 1 must be higher than r_1 and that the tank 2 higher than r_2. At $x1$ and $x2$, arriving to $r1 = r2 = 2$, the simulations run an increasing number of mode transitions to try to verify $x1 \geqslant r1$ and $x2 \geqslant r2$.

or stochastic. In Figure 4.3, the state T represents the temperature and the mode variable is K.

4.4.2 Simulations

BioRica works with switched systems that considers two types of models to define the system dynamics: differential equations and event transitions. To do that, it divides the time into intervals $(t_{i-1}, t_i]$ with i between 1 and the number of iterations, and estimates the values of the states variables at each time t_i by using numerical solvers of differential equations and checking the possibility to have mode transitions.

With respect to differential equations, the current version of BioRica uses Runge-Kutta to numerically solve differential equations systems, and include techniques of memory-less to reuse *a priori* computed expressions.

Given a BioRica node, the continuous model is solved and, at each time, the simulation checks the guard conditions. If the guard is verified, the transition happens and the resolutions in all the nodes are restarted from this time. The variables values over time are stored and displayed as trace and pictures are automatically done. The simulation is stopped when the maximal time (or iteration number 1000) is reached.

A necessary condition of simulations is that given a finite time interval, the number of mode transitions that are chosen is finite. When models present the so called *Zeno's paradox*, with infinite possible mode transitions in a finite time interval, they can be simulated but the number of mode transitions depends on the temporal accuracy that is allowed. That is the case of the water tanks system in Figure 4.4. This system decides between two tanks 1 and 2 that are continuously filled, with the constraint that the levels must not be lower than r_1 and r_2 respectively. The continuous dynamics of x_1 and x_2, representing the water level of each tank, is modeled by a system of differential equations. The mode variable is K: if its value is 1 then the system sends water to the first tank, and if it is 0 then to the second one. The simulation of the model produces many mode transitions in a finite interval of time.

4.5 Conclusion

In this chapter we established the requeriments to go from the theoretical ideas to the practice, defined a general schema for solving hybrid systems which is independent of the implementation, and showed how to implement hybrid systems with the BioRica framework.

The notions of hybrid systems, composition, hierarchy and reusing of models must be implemented by a framework that allows the integration of different types of models: continuous dynamics, discrete dynamics with stochastic and non-deterministic decisions and timed behaviors. In particular we need to solve systems of differential equations and, according to our modeling approach (section 3), we need to implement hybrid systems with coefficient switches, and strong switches that allows the reusing of models.

We built a general schema for solving hybrid systems, which is adapted for both types of hybrid systems. It shows the computation steps based on two actions: solving and updating. At any given time the continuous model is solved, and the model is updated when detecting a guard condition that changes the value of the mode variable, which updates the model (Figure 4.1). We chose the BioRica framework to implement our approach. As explained here, it allows us to implement hybrid systems including composition and reusing of models, satisfying all our requirements specified in section 4.1. During this work we contributed to improving BioRica.

Chapter 5

Application: Reconciling competing models, particular case of wine fermentation kinetics

When a given systems is investigated by many teams, it is common that competing models are developed, each one constructed to meet particular needs and to fit specific data. Reconciling models into one combined model allows us to respond to particular needs and obtain more general models. We use the theory of hybrid systems and the BioRica framework to investigate how one might be able to integrate and reuse groups of models and generalize them.

In this chapter we use the reconciliation between models of wine fermentation kinetics to study examples of hybrid systems defined by coefficient switches or by strong switches as explained in section 3.2. Some model compositions can be done by changing the coefficients of the differential equations, while the reconciliation of more divergent models needs to use strong switches between alternate continuous models. This hybrid model results from the reconciliation of three wine fermentation models ([10]): Coleman ([28]), Scaglia ([94]) and Pizarro ([85], [92]). For each factor configuration, one chooses the model that best fits the experimental data of three papers: [85], [75] and [79] used as training.

The reconciliation of wine fermentation kinetic models gives us a better prediction of the dynamics than any of the original models. In function of the initial conditions one chooses the model and arriving at the stable fermentation phase, the model is updated to obtain the best predictions. The effect that produces the level of nutrients on the behavior of the system is successfully described by switching the model to include competence coefficients when the resources are scarce. The resulting model can be used to predict problems such as stuck (with residual sugar) and sluggish (very slow) fermentations ([17]).

5.1 Modeling wine fermentation kinetics

The application of biotechnology to the wine fermentation process is fundamental because the production of ethanol is the result of the action of the yeasts to metabolize the sugar in an anaerobic way. The wine market is enormous and improvements of the process can reduce losses; stuck or sluggish fermentations ([17]), for example, cause worldwide losses estimated at 7 billion € annually. Modeling the kinetics of the process may give us the means to predict the best conditions to ferment, and to detect stuck and sluggish fermentation in time to save them.

The fermentation process transits by different steps, among other factors they depend on the quantity of organisms and of the nutrients they consume. As time passes these conditions change and consequently the laws of the dynamics change too. When few yeasts have high concentrations of sugar available the behavior is different than when many yeasts compete to consume low concentrations of nutrients. Consequently, it is understable that a change of follows change of conditions of factors such as concentration of yeasts and nutrients.

We consider three models to reconcile: Coleman ([28]), Scaglia ([94]) and Pizarro models ([85], [92]). Here we give a brief description of the models, see more details in [10]. Table 5.1 recalls the meaning of the variables and our notation as compared with the nomenclature of the original papers.

The Coleman model ([28]) consists of a 5 coupled ODEs (equations 5.1- 5.5), combined with 4 one-dimensional regression models to estimate parameters. The variables that are represented in differential equations are concentrations of: biomass (X), active biomass (X_A), nitrogen (N), ethanol ($EtOH$) and sugar (S). It considers biomass concentration controlled by the growth rate (μ) with nitrogen nutrition and without competition (Monod's equation, [80], see Table 5.1), the lower is the remaining nitrogen the lower the growth rate, and the death rate (τ) is proportional to the ethanol concentration. The other fermentation variables are obtained by estimating production rates (for $EtOH$) or consumption rates (for N and S) per biomass unit. The Scaglia model uses only 4 fermentation variables: X, S, CO_2 (carbonic dioxide gas concentration) and $EtOH$, equations 5.6- 5.9. In the cell growth expression (equation 5.6), the quadratic coefficient of population, β, models the competition for available resources. They consider that the lower is the remaining sugar the lower the growth rate, and the faster the decrease of substrate concentration the larger the increase in the cellular death rate. The carbon dioxide concentration (CO_2, equation 5.9) is estimated with a emission rate coefficient per biomass unity ν_{CO_2}, and the rate of an additional coefficient that we called $CO2Form$. The ethanol production rate is obtained by dividing ethanol produced by carbon dioxide emitted (equation 5.7, see the coefficient $Y_{CO_2/EtOH}$ in Table 5.1). The sugar consumption rate (equation 5.8) is composed by a term acting on biomass and a quadratic term associated to competition between organisms. The main difference between Coleman and Scaglia models is that the latter includes the competition for available resources. Coleman model has the advantage of including the temperature as a variable.

The Pizarro model ([85], [92], equations 5.10- 5.14) uses essentially the same differential equations as Coleman model, but it adds the variable glycerol Gly and does not consider active biomass concentration. Although the three models use differential equations, the Pizarro process combines them with an iterative optimization approach using FBA. It is built through an iterative process where intracellular network fluxes are bounded according to extracellular conditions, and for each iteration a maximization (of growth or glucose consumption rate) is performed to obtain uptakes and consumption rates that are used to predict extracellular concentrations of metabolites.

$$\frac{dX}{dt} = \mu \cdot X_A \qquad (5.1)$$

$$\frac{dX_A}{dt} = (\mu - \tau) \cdot X_A \qquad (5.2)$$

$$\frac{dN}{dt} = -\nu_N \cdot X_A \qquad (5.3)$$

$$\frac{d[EtOH]}{dt} = \nu_{EtOH} \cdot X_A \qquad (5.4)$$

$$\frac{dS}{dt} = -\nu_S \cdot X_A \qquad (5.5)$$

$$\frac{dX}{dt} = (F_\mu \cdot \mu - F_\tau \cdot \tau) \cdot X - F_\mu \cdot \beta \cdot X^2 \qquad (5.6)$$

$$\frac{d[EtOH]}{dt} = \tfrac{1}{Y_{CO_2/EtOH}} \cdot \tfrac{dCO_2}{dt} \qquad (5.7)$$

$$\frac{dS}{dt} = -\big((\nu_S + \nu_{S_0}) \cdot X - \tfrac{0.00002}{Y_{X/S}} \cdot X^2\big) \qquad (5.8)$$

$$\frac{dCO_2}{dt} = \nu_{CO_2} \cdot X + \tfrac{d(C02Form)}{dt} \qquad (5.9)$$

$$\frac{dX}{dt} = \mu \cdot X \qquad (5.10)$$

$$\frac{dN}{dt} = -\nu_N \cdot X \qquad (5.11)$$

$$\frac{d[EtOH]}{dt} = \nu_{EtOH} \cdot X \qquad (5.12)$$

$$\frac{dS}{dt} = -\nu_S \cdot X \qquad (5.13)$$

$$\frac{d[Gly]}{dt} = \nu_{Gly} \cdot X \qquad (5.14)$$

5.2 Experimental data and exploratory analysis

In our study we considered experimental data of three papers: [85], [75] and [79]. The experimental measures of the Pizarro team correspond to a wide range of data. We used laboratory results that were obtained with the strain *Prise de Mousse EC1118*, and industrial results for Industrial Cabernet Sauvignon, wine fermentations that were monitored during the 2003 vintage at a commercial winery in Chile. Sugar profiles for six batch fermentations at 28 °C with high/low nitrogen and other two at 12 °C and 17 °C with high conditions of nitrogen were used to

		Original notation and computation		
Notation	Meaning	Coleman ([28])	Pizarro ([85])	Scaglia ([94])
X	viable biomass concentration, $[g \cdot l^{-1}]]$	X total biomass concentration	X_V	X
X_A	active biomass concentration, $[g \cdot l^{-1}]]$	X_A		
N	nitrogen concentration, $[g \cdot l^{-1}]$	N	$NH4$	N
$EtOH$	ethanol concentration, $[g \cdot l^{-1}]$	E	$EtOH$	P
S	sugar concentration, $[g \cdot l^{-1}]$	S	glu	S
Gly	glycerol concentration, $[g \cdot l^{-1}]$	Gly		
CO_2	carbon dioxide concentration, $[g \cdot l^{-1}]$			CO_2
μ	growth rate of yeast cells, $[h^{-1}]$	Monod's model nitrogen nutrition: $\frac{max(\mu) \cdot N}{K_N + N}$	μ, Flux balance analysis	$12.072 \cdot \frac{max(\mu) \cdot S}{S + K_S \cdot 93.023^{1.508}}$
τ	death rate of yeast cells, $[h^{-1}]$	k_d, temperature and ethanol dependent		$(0.0001 - 0.047 \cdot \frac{dS}{dt})$
ν_N	nitrogen consumption rate per yeast mass, $[g \cdot g^{-1} \cdot h^{-1}]$	$\frac{\mu}{Y_{X/N}}$	ν_{NH4}, Flux balance analysis	
ν_{EtOH}	ethanol production rate per yeast mass, $[g \cdot g^{-1} \cdot h^{-1}]$	$\beta = \frac{max(\nu_{EtOH}) \cdot S}{K_S + S}$	ν_{EtOH}, Flux balance analysis	
ν_S	sugar consumption rate per yeast mass, $[g \cdot g^{-1} \cdot h^{-1}]$	$\frac{\nu_{EtOH}}{Y_{EtOH/S}}$	ν_{glu}, Flux balance analysis	$\frac{1}{Y_{X/S}} \cdot \frac{max(\mu) \cdot S}{S + K_S \cdot 93.023^{1.508}}$
ν_{S_0}	initial sugar consumption rate per yeast mass, $[g \cdot g^{-1} \cdot h^{-1}]$			$F = 0.008$, initial sugar dependent
ν_{gly}	glycerol production rate per yeast mass, $[g \cdot g^{-1} \cdot h^{-1}]$		ν_{gly}, Flux balance analysis	
ν_{CO2}	carbon dioxide production rate per yeast mass, $[g \cdot g^{-1} \cdot h^{-1}]$			$G \cdot \frac{max(\mu) \cdot S}{S + K_S \cdot 93.023^{1.445}}$
K_N	constant for nitrogen-limited growth, $[g \cdot l^{-1}]$	0.009 Monod constant		
K_S	constant for sugar utilization in growth, $[g \cdot l^{-1}]$	10.278 Michaelis-Menten-type constant		2.15 Monod constant for sugar-limited growth
$Y_{X/N}$	yield coefficient for cell mass grown per mass of nitrogen used, $[g \cdot g^{-1}]$	$Y_{X/N}$, temperature dependent		
$Y_{EtOH/S}$	yield coefficient for ethanol produced per sugar consumed, $[g \cdot g^{-1}]$	$Y_{E/S} = 0.55$		
$Y_{X/S}$	yield coefficient for cells formed per sugar consumed, $[g \cdot g^{-1}]$			0.029
$Y_{CO2/EtOH}$	yield coefficient for CO_2 formed per ethanol produced, $[g \cdot g^{-1}]$			$Y_{CO2/P}$, initial conditions dependent
$max(\mu)$	maximum growth rate of yeast cells, $[h^{-1}]$	μ_{max}, temperature dependent		μ_m, initial conditions dependent
$max(\nu_{EtOH})$	maximum ethanol production rate, $[g \cdot g^{-1} \cdot h^{-1}]$	β_{max}, temperature dependent		
$max(CO_2)$	maximum CO_2 for normal fermentation progress, $[g \cdot l^{-1}]$			$CO_{2,95} = 78$, corresponds to 95% of maximum emissions for initial conditions.
F_μ	correction factor of growth rate			$\frac{exp(-(CO_2 - max(CO_2)))}{exp(CO_2 - max(CO_2)) + exp(-(CO_2 - max(CO_2)))}$
F_τ	correction factor of death rate			$1 - F_\mu$
β	β competition coefficient in logistic equation $\frac{dX}{dt} = \mu \cdot X \cdot (1 - \frac{\beta}{\mu} \cdot X)$ [82], $l \cdot g^{-1} \cdot h^{-1}$			β, initial conditions dependent
$CO2Form$	additional Scaglia concentration coefficient for carbon dioxide formation, $[g \cdot l^{-1}]$			$241.44 \cdot max(\mu) \frac{S^2}{(S + K_S \cdot 93.023^{1.0})(S + K_S \cdot 93.023^{1.445})} \cdot X + 0.01 \cdot X$

Table 5.1: Notation, comparison with original papers nomenclature.

calibrate the model, and consequently we expect a better adjustment of this model for sugar in this conditions. By direct communication with [75], we obtained data of biomass profiles in two particular conditions: moderate temperature (24 $°C$), high level of sugar (280 g/l) and moderate/high levels of nitrogen (approximately 220 mg/l and 551 mg/l respectively). The data set available in Mendes-Ferreira study ([79]) used the strain *Saccharomyces cerevisiae PYCC4072* that was supplied by the Portuguese Yeast Culture Collection. The paper describes experimental biomass, ethanol and sugar (and other indexes) results for two experimental conditions, fermentation maintained at 20 $°C$ with moderate initial sugar concentration (200 g/l) and initial nitrogen concentration high (267 mg/l) or low (66 mg/l). The acidity conditions were adjusted to pH 3.7, nitrogen is supplied by ammonium phosphate and sugar corresponds to glucose.

The observation of experimental data gave us some ideas about the profiles of fermentation variables. In the three variables (biomass, ethanol and sugar concentrations) we observed two phases, transient and stable. Before a particular time, that we call *stabilization time*, fermentation variables change exponentially over time. After stabilization these are statistically constant. We verified the exponential behavior of biomass profiles statistically by means of linearization and linear regression (Figure 5.1), the growth rate can be assumed to be constant over time. In the case of biomass, in the first phase it increases exponentially until the cells stop their growth. Ethanol concentration increases while the yeast cells are active, after the stabilization time the production stops. Sugar concentration decays in an exponential way until it is completely consumed. Different samples show different uptake (for biomass and ethanol) and consumption rates (for sugar). For the Coleman ([28]) and Pizarro ([85]) models, fermentation variables evolve in time according to uptake (biomass, ethanol and glycerol) or consumption (sugar, nitrogen) factors per concentration unity of yeast cell (biomass). They assume that these coefficients change in time depending of the fermentation or environmental variables and do not depend only on initial conditions. When solving these models one obtains exponential behaviors whose rates change over time, according to the value of the fermentation variables or environmental conditions, finishing in a stable phase. According to the sign of the factors we have exponential growth (positive sign) or decay (negative sign) followed by stabilization. Biomass profiles resemble the solutions of the logistic differential equation, which is widely used in ecology to model population growth. These types of differential equations were derived by Verhulst in 1838 to describe the self-limiting growth of a biological population (Verlhurst's model; see [82]). Population starts to grow in an exponential phase, as it gets closer to the carrying capacity the growth slows down and reaches a stable level. The equations (5.6-5.9) that define the Scaglia model ([94]) include logistic components but they are more complex, and one observes relations between one-order differential expressions of variables.

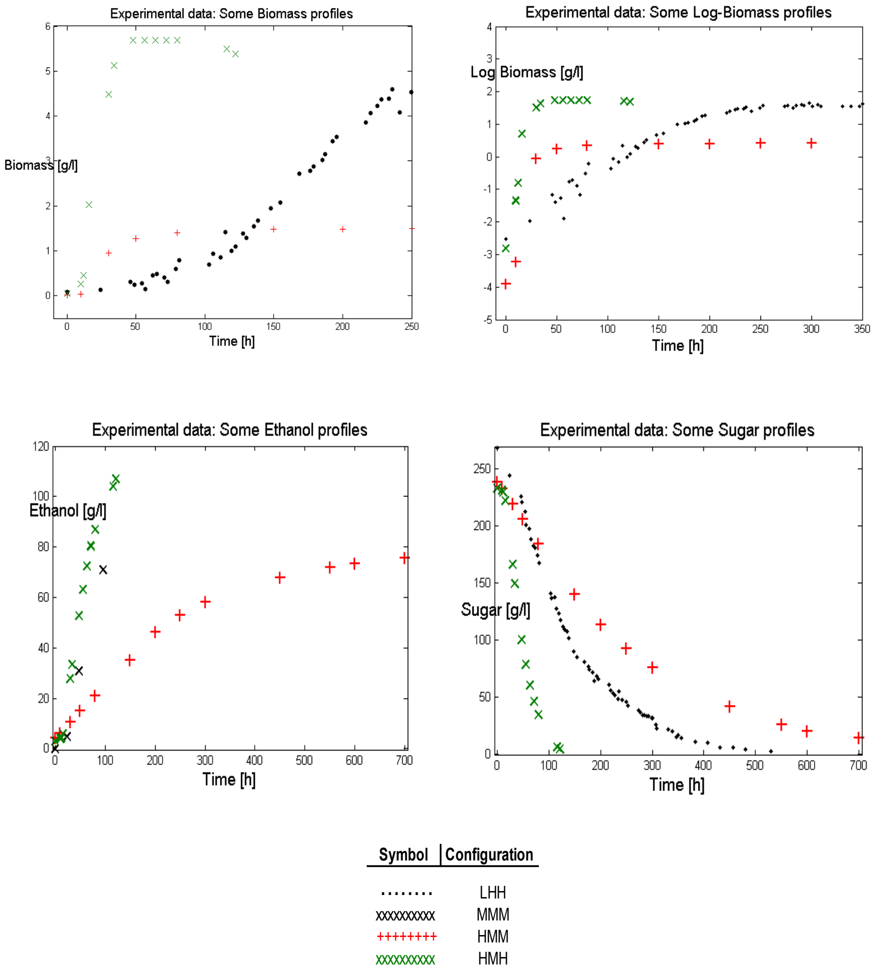


Figure 5.1: Fermentation variables profiles for some initial conditions configurations. For Biomass and Sugar one shows experimental results in LHH (low temperature: 12 $°C$, high initial sugar: 268 g/l, high initial nitrogen: 300 mg/l), HMM (28$°C$, sugar: 238 g/l, nitrogen: 50mg/l) and HMH (28 $°C$, sugar: 233 g/l, nitrogen: 300 mg/l) configurations. For Ethanol we show data for MMM configuration (20 $°C$, sugar: 200 g/l, nitrogen: 66 mg/l) instead of LHH. Log-Biomass profiles are shown too, we obtained linear correlations in transient phase of 0.98, 0.99 and 0.97 respectively.

Temperature	Sugar	Nitrogen	Biomass	Ethanol	Sugar
Low (0-19 °C)	Moderate (160-240 [g/l])	Moderate (50-240 [mg/l])			
		High (240-551 [mg/l])			🟦 1
	High (240-308 [g/l])	Moderate			
		High	🟦 2		🟦 3
Moderate (20-27 °C)	Moderate	Moderate	🟨 1	🟨 1	🟨 1
		High	🟨 1	🟨 1	🟦 6 🟨 1
	High	Moderate	🟪 1		
		High	🟪 1		🟦 2
High (28-35 °C)	Moderate	Moderate	🟦 1	🟦 1	🟦 1
		High	🟦 2	🟦 1	🟦 2
	High	Moderate			
		High			🟦 1

🟦	**Pizarro data ([22])**
🟪	**Malherbe data ([15])**
🟨	**Mendes-Ferreira data ([16])**

Table 5.2: Origin of experimental data. For each Initial Temperature-Sugar-Nitrogen configuration, and fermentation variable it is showed the origin of available data.

			Biomass		Ethanol		Sugar	
Temperature	**Sugar**	**Nitrogen**	Transient	Stable	Transient	Stable	Transient	Stable
Low (0-19 °C)	Moderate (180-240 [g/l])	Moderate (50-240 [mg/l])						
		High (240-551 [mg/l])					Pizarro	Pizarro
	High (240-308 [g/l])	Moderate						
		High	Indiferent	Pizarro			Scaglia/Pizarro	Scaglia/Pizarro
Moderate (20-27 °C)	Moderate	Moderate	Coleman/Pizarro	Coleman	Coleman	Indifferent	Coleman/Pizarro	Coleman
		High	Scaglia/Pizarro	Pizarro	Indifferent	Pizarro	Scaglia/Pizarro	Pizarro
	High	Moderate	Scaglia	Pizarro				
		High	Scaglia	Scaglia			Scaglia	Coleman
High (28-35 °C)	Moderate	Moderate	Coleman	Coleman	Pizarro	Coleman	Pizarro	Pizarro
		High	Coleman	Pizarro	Indifferent	Pizarro	Coleman	Pizarro
	High	Moderate						
		High					Pizarro	Scaglia/Pizarro

Quality **Worst** … Best

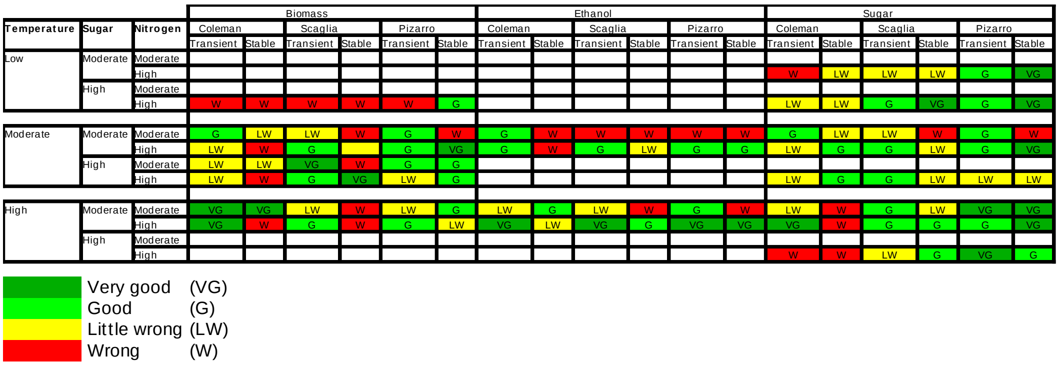
Table 5.3: Criterion of selection of best models in function of initial conditions. For each combination variable-phase is written the best model, colors represent the quality of the adjustment when comparing between all the initial conditions.

5.3 Statistical classification of the results

We considered as factors of the fermentation process the initial conditions and time. We studied the fermentation variables: X (biomass concentration), $EtOH$ (ethanol concentration) and S (sugar concentration). Initial conditions were determined by ranges of initial temperature, sugar and nitrogen concentration, and we divided in transient and stable phase. The levels of initial conditions were ranges of initial temperature, sugar and nitrogen concentrations. We considered that the temperature is *low* when it is lower than 19 $°C$, is *moderate* for values between 20 $°C$ and 27 $°C$, and *high* for larger values. Initial sugar concentration was called *moderate* for values less than 240 g/l and *high* for superior values. Initial nitrogen concentration was *moderate* for values less than 240 mg/l and *high* for those superior values. For each initial conditions configuration, we separated the profiles in *transient* and stable *phase* by analyzing the experimental results.

In Table 5.2 we show the origin of experimental data for different initial condition levels and fermentation variables. This classification allowed us to cover a wide range of configurations. In spite of this, for some combinations we do not have experimental data because conditions of fermentation are difficult. This is the case for *low* initial temperature, sugar and nitrogen concentrations. For *high* temperature and sugar with insufficient levels of nitrogen source, we have the same situation. Part of the data have superior statistical quality. While the number of samples that describe an initial condition configuration is larger, the quantity of information that validates our assertions about the adjustment of each model in this configuration is also larger. In particular the Pizarro sample for HMM (high temperature, moderate initial sugar level and moderate initial nitrogen level) and HMH configurations give us standard deviations for variable profiles. Mendes-Ferreira ([79]) samples for MHM (moderate temperature, high initial sugar level and moderate initial nitrogen level) and MHH supply means and standard deviations of measures too.

We evaluated the three studied models according to how well they agree with the experimental results (Figure 5.2). For each sample we reviewed the adjustment in the transient and the stable phase. For the local criterion, at each point we built confidence intervals of experimental results, and computed the *p-values* associated to the decision of considering simulated value equal to experimental result. Because we observe exponential behavior, for the global criterion we computed the correlation between the logarithm of simulations and the data over the time (Table 5.4). In general, an adjustment was considered *Very good* if the local criterion and the global one are very favorable ($p\text{-}value \geqslant 0.1$, $correlation \geqslant 0.98$); *Good* if a criterion is very favorable and the other one is only favorable ($0.05 \leqslant p\text{-}value < 0.1$ or $0.95 \leqslant correlation < 0.98$); *Little wrong* if a criterion is unfavorable ($p\text{-}value < 0.05$ or $correlation < 0.95$) and the other one is favorable or superior; and *Wrong* if both criteria are unfavorable. The cases near to the limits were checked especially. In case the local criterion is absolutely unfavorable ($p\text{-}value = 0$) we qualified in *Wrong*, if local criterion is unfavorable (but not absolutely) and global criterion is optimum ($correlation = 1$) we considered it *Good*.



Temperature	Sugar	Nitrogen	Biomass						Ethanol						Sugar					
			Coleman		Scaglia		Pizarro		Coleman		Scaglia		Pizarro		Coleman		Scaglia		Pizarro	
			Transient	Stable	Transient	Stable	Transient	Stable	Transient	Stable	Transient	Stable	Transient	Stable	Transient	Stable	Transient	Stable	Transient	Stable
Low	Moderate	Moderate																		
		High													W	LW	LW	LW	G	VG
	High	Moderate																		
		High	W	W	W	W	W	G							LW	LW	G	VG	G	VG
Moderate	Moderate	Moderate	G	LW	LW	W	G	W	G	W	W	W	W	W	G	LW	LW	W	G	W
		High	LW	W	G	🟨	G	VG	G	W	G	LW	G	G	LW	G	G	LW	G	VG
	High	Moderate	LW	LW	VG	W	G	G												
		High	LW	W	G	VG	LW	G							LW	G	G	LW	LW	LW
High	Moderate	Moderate	VG	VG	LW	W	LW	G	LW	G	LW	W	G	W	LW	W	G	LW	VG	VG
		High	VG	W	G	W	G	LW	VG	LW	VG	G	VG	VG	VG	W	G	G	G	VG
	High	Moderate																		
		High													W	W	LW	G	VG	G

🟩	Very good	(VG)
🟩	Good	(G)
🟨	Little wrong	(LW)
🟥	Wrong	(W)

Figure 5.2: Summary of results of adjustment according to initial conditions. Quality of adjustment of each model for each initial conditions configuration and phase (transient and stable).

5.4 Reconciled model of wine fermentation kinetics

The reconciliation process in [10] produces a model that describes the dynamics of X, $EtOH$ and S by the differential equations bellow:

$$\frac{dX}{dt} = \mu_A \cdot X_A + \mu^{(1)} \cdot X - \mu^{(2)} \cdot X^2 \tag{5.15}$$

$$\frac{d[EtOH]}{dt} = \epsilon_A \cdot X_A + \epsilon^{(1)} \cdot X + \epsilon_{CO_2} \cdot \frac{dCO_2}{dt} \tag{5.16}$$

$$\frac{dS}{dt} = -\left(\sigma_A \cdot X_A + \sigma^{(1)} \cdot X - \sigma^{(2)} \cdot X^2\right) \tag{5.17}$$

In such model one separates the possible values of temperature, sugar and nitrogen in three categories: low (L), moderate (M), and high (H) and choses the models with best adjustment to experimental data on the transient and the stable temporal phase of fermentation. Temperatures between $0°C$ and $19°C$ are considered to be low, between $20°C$ and $27°C$ is moderate, and between $28°C$ and $35°C$ is high. The concentration of sugar is considered to be moderate for values between $160[g/l]$ and $240[g/l]$, and high for values between $240[g/l]$ and $309[g/l]$. Finally, we say that the level of nitrogen is moderate if its value is between $50[mg/l]$ and $240[mg/l]$, and high if it is between $240[mg/l]$ and $551[mg/l]$.

These equations capture the three models and the variables X_A and CO_2 are those computed by the Coleman and Scaglia models respectively. The coefficients μ_A, ϵ_A and σ_A correspond to coefficients of the Coleman model to represent linear effect of X_A on X. The linear coefficient of X on $EtOH$, $\epsilon^{(1)}$, is associated to the Pizarro model; $\mu^{(1)}$ is composed by contributions of the Pizarro and Scaglia models, and quadratic effects (coefficients $\mu^{(2)}$ and $\sigma^{(1)}$) are obtained from the Scaglia model. The coefficients are active or not in function of initial configuration and time (Table 5.5) to fit the experimental data of three papers: [85], [75] and [79].

		Coleman model				Pizarro model				Scaglia model				Combined model			
		Transient		Stable		Transient		Stable		Transient		Stable		Transient		Stable	
Config.	Variable	C.1	C.2	C.1	C.2	C.1	C.2	C.1	C.2	C.1	C.2	C.1	C.2	C.1	C.2	C.1	C.2
MMM	X	0.006	1	0.006	0.92	0.109	1	0	1	0.012	0.96	0	0.77	0.033	1	0.006	0.92
	$EtOH$	0.001	1	0	0.97	0	1	0	1	0	1	0	0.93	0.001	1	0	0.97
	S	0.07	0.97	0.001	0.93	0.022	1	0	0.98	0.017	0.95	0	0.98	0.063	1	0.001	0.93
MMH	X	0.045	0.98	0.001	0.90	0.006	1	0.483	1	0.113	0.98	0.06	0.99	0.256	0.99	0.483	1
	$EtOH$	0.047	1	0	0.97	0.001	1	0.053	1	0.048	1	0.149	0.92	0.049	1	0.053	1
	S	0.129	0.93	0.152	0.98	0.140	0.98	0.27	0.98	0.064	0.99	0.015	0.97	0.129	0.99	0.27	0.98
HMM	X	0.388	0.99	0.476	0.97	0.102	0.83	0.027	1	0.082	0.99	0	0.87	0.388	0.99	0.476	0.97
	$EtOH$	0.049	0.95	0.155	0.97	0.129	0.95	0	1	0.08	0.92	0	-0.97	0.129	0.95	0.155	0.97
	S	0.171	0.79	0	-0.7	0.238	1	0.272	0.97	0.107	0.91	0.032	0.73	0.238	1	0.272	0.97
HMH	X	0.203	0.97	0	0.28	0.156	0.98	0.048	0.97	0.197	0.96	0	0.91	0.203	0.97	0.048	0.97
	$EtOH$	0.275	1	0.089	0.80	0.162	0.99	0.214	0.99	0.264	0.98	0.001	1	0.339	0.99	0.214	0.99
	S	0.327	1	0	0.59	0.135	0.98	0.167	0.99	0.197	0.95	0.001	1	0.327	1	0.167	0.99

Table 5.4: Statistical analysis of models. For each configuration of factors and fermentation variable, we show the average *p-value* for the local criterion (C.1), and *correlation* for global criterion (C.2). The lower the *p-value*, the bigger the local error in simulations. The bigger the *correlation*, the bigger the global similarity between data and simulations. Results for other configurations can be asked.

		Configuration of initial conditions								
Coefficient	Meaning	LMH	LHH	MMM	MMH	MHM	MHH	HMM	HMH	HHH
$\mu_A = \frac{max(\mu)\cdot N}{K_N+N}$	linear effect of X_A	-	$\varnothing$	$\forall t$	$\varnothing$	$\varnothing$	$\varnothing$	$\forall t$	$t \leqslant 30$	-
$\mu^{(1)} = FBA, \quad (F_\mu \cdot \mu + F_\tau \cdot \tau)$	linear effect of X	-	$t > 110$	$t \leqslant 96$	$\forall t$	$\forall t$	$\forall t$	$\varnothing$	$t > 30$	-
$\mu^{(2)} = F_\mu \cdot \beta$	quadratic effect of X	-	$\varnothing$	$\varnothing$	$t \leqslant 51$	$t \leqslant 27$	$\forall t$	$\varnothing$	$\varnothing$	-
$\epsilon_A = \frac{max(\nu_{EtOH})\cdot S}{K_S+S}$	linear effect of X_A	-	-	$t \leqslant 96$	$t \leqslant 51$	-	-	$t > 300$	$t \leqslant 30$	-
$\epsilon^{(1)} = FBA$	linear effect of X	-	-	$\varnothing$	$\forall t$	-	-	$t \leqslant 300$	$\forall t$	-
$\epsilon_{CO_2} = \frac{1}{Y_{CO_2/EtOH}}$	linear effect of $\frac{dCO_2}{dt}$	-	-	$\varnothing$	$t \leqslant 51$	-	-	$\varnothing$	$t \leqslant 30$	-
$\sigma_A = \frac{\nu_{EtOH}}{Y_{EtOH/S}}$	linear effect of X_A	$\varnothing$	$\varnothing$	$\forall t$	$\varnothing$	-	$t > 121$	$\varnothing$	$t \leqslant 30$	$\varnothing$
$\sigma^{(1)} = FBA, \quad 0.008 + \frac{max(\mu)\cdot S}{Y_{X/S}\cdot(S+K_S\cdot 93.023^{1.508})}$	linear effect of X	$\forall t$	$\forall t$	$t \leqslant 96$	$\forall t$	-	$t \leqslant 121$	$\forall t$	$t > 30$	$\forall t$
$\sigma^{(2)} = \frac{2\cdot 10^{-5}}{Y_{X/S}}$	quadratic effect of X	$\varnothing$	$\forall t$	$\varnothing$	$t \leqslant 51$	-	$t \leqslant 121$	$\varnothing$	$\varnothing$	$t > 105.5$

Table 5.5: Temporal intervals at which the coefficients of equations 5.15-5.17 are active for each configuration of initial conditions and the formulas. One writes $-$ if there are not experimental data to validate, FBA denotes the result by using Flux balance Analysis.

5.5 Results

We model reconciled models of wine fermentation kinetics ([10]) by using both approaches: coefficient switches and strong switches (Figure 3.3). The computation steps needed to solve these systems (Figure 4.1) were implemented with BioRica. We describe how to use the reconciled models to predict stuck and sluggish fermentations *in vivo*.

5.5.1 Reconciled models and simulations

In the reconciliation process, we considered four factors: initial levels of temperature, sugar and nitrogen, and temporal phase: transient and stable. The possible values of temperature, sugar and nitrogen are classified in three categories: low (L), moderate (M), and high (H). Coleman, Scaglia and Pizarro models ([28], [94], [85]) show different levels of precision in function of the factor values. We chose for each configuration of factors, the combination of models that obtains the best adjustments of experimental results of three papers: [85], [75] and [79]. The reconciled model is given by equations 5.15-5.17 with coefficients in Table 5.5 (section 5.1). We decide which model to use to start the model in function of the configuration, while at changing temporal phase we update this selection. In order to consider the experimental variability, we model the transition time as a random variable. That is to say, the time at which the system changes from transient to stable phase is simulated as a gaussian variable whose mean and standard deviation are computed for each configuration of temperature, sugar and nitrogen. For example, according to the computations summarized in Table 5.5, for configuration MHH the mean ($TTime\_MHH$) is 121 and the standard deviation 18.55, and for the configuration HHH these values are 105.5 and 19.125 respectively. Consequently, the resulting system is a hybrid system with stochastic mode transitions. According to which models are needed, we use coefficient switches or strong switches.

The differential equations systems of Coleman [28] and Scaglia [94] are solved in BioRica, but the Pizarro model is externally solved due to its superior complexity. The maximization problem makes not viable the direct implementation in BioRica at this moment, and there are more appropiate tools to solve this model. So, Coleman and Scaglia models use the *eq:diff* syntax and Pizarro model *eq:extern*. Consequently, if the reconciliation of models does not include the Pizarro model we use Hybrid systems with coefficient switches, but if the reconciliation considers it we use the approach with strong switches.

We begin with an example modeled with coefficient switches. Let us consider the configuration MHH: moderate temperature $23°C$, high sugar $308.6[g/l]$, and high nitrogen $280[mg/l]$. Here, we observed an initial effect of competition to consume resources. For the biomass concentration X, the model is given by Scaglia equations. For sugar consumption the Scaglia model is active in the transient phase, while the Coleman model is used in the stable phase. Pizarro model is not active. The time at which the system changes from transient to stable phase is modeled as a gaussian variable with mean $TTime\_MHH = 121$ and standard deviation $sd\_MHH = 18.55$ The code and simulation results are shown in Figure 5.3. The simulations for

(A)

```
const Xinit=0.2;XAinit=0.2;Sinit=308.6;CO2init=0;Einit=0;T=23;

const
      a=1.508;b=1.508;c=1.4445;d=1.0;e=1.4445;A=12.072;
      B=93.023;C=0.047;D=0.0001;E=0.0000215;F=0.0081;
      G=10.864;H=241.44;I=0.01;mu=0.111;KS=2.15;
      beta=0.0005;CO2_95=78;Y_XS=0.029;Y_CO2E=1.0443;
      max_mu=0.111;;K_S=10.278;Y_EtOHS=0.55;
      KN=exp(-4.73);mu_max=exp(log_mumax);
      kprim_d=exp(-9.810-0.00108*T+0.00478*T^2);

      sd_MHH=18.55;TTime_MHH=121;

formula
       max_nu_EtOH=exp(-2.30+0.0771*T);
       F1 = (exp(-(CO2-CO2_95))/(exp(CO2-CO2_95)+exp(-(CO2-
            CO2_95))))*A*mu*S/(S+KS*B^a);
       F2 = A*mu*S/((S+KS*B^a)*beta);
       F3 = 1-(exp(-(CO2-CO2_95))/(exp(CO2-CO2_95)+exp(-(CO2-
            CO2_95))));
       G1 = max_mu*S/(S+KS*B^b);
       H1 = G*mu*S/(S+KS*B^c);
       H2 = (S^2*(KS*B^e+KS*B^d)+2*S*(KS^2*B^d*B^e))/
            ((S+KS*B^d)^2*(S+KS*B^e)^2);
       H3 = (S^4+S^3*(KS*B^e+KS*B^d)+S^2*(KS*B^d*B^e))/
            ((S+KS*B^d)^2*(S+KS*B^e)^2);

       nu_S_Scaglia= (1/Y_XS)*G1;
       sigma_2= (1/Y_XS)*E;
       nu_EtOH= max_nu_EtOH*S/(K_S+S);
       nu_S_Coleman= nu_EtOH/Y_EtOHS;
       mu=mu_max*N/(KN+N);kd=kprim_d*E;
```

```
node MODEL
     state S,X,XA,CO2,E: FLOAT;
     flow mode_S,mode_C:FLOAT:in;
     eqdiff
          dS = -(mode_S*((nu_S_Scaglia+F)*X-
          sigma_2*X^2)+mode_C*nu_S_Coleman*XA);
          dXA= (mu-kd)*XA;
          dX = F1*X*(1-X/F2)+F3*(C*X*dS-D*X);
          dCO2 = H1*X+H*mu*X*(dS*H2+dX*H3)+dX;
          dEtOH = (1/Y_CO2E)*dCO2;
     init
          X:=Xinit,XA:=XAinit,S:=Sinit,CO2:=CO2init,
          EtOH:=Einit;
edon

node MODES
     state _mode_S,_mode_C: FLOAT; stable:BOOL;
     flow mode_S,mode_C:FLOAT:out;
     event change_phase;
     trans
          stable=False | change_phase
          -> stable=True,mode_S=0,mode_C=1;
     assert
          mode_S=_mode_S,mode_C=_mode_C;

     init mode_S:=1,mode_C:=0,stable:=False;

     extern law<change_phase>:Normal{TTime_MHH,sd_MHH};
edon

node RECONCILED_MODEL_OF_SUGAR_CONFIG_MHH
     sub M:MODEL;Mo:MODES;
     assert
        M.mode_S=Mo.mode_S,M.mode_C=Mo.mode_C;
edon
```

(B)

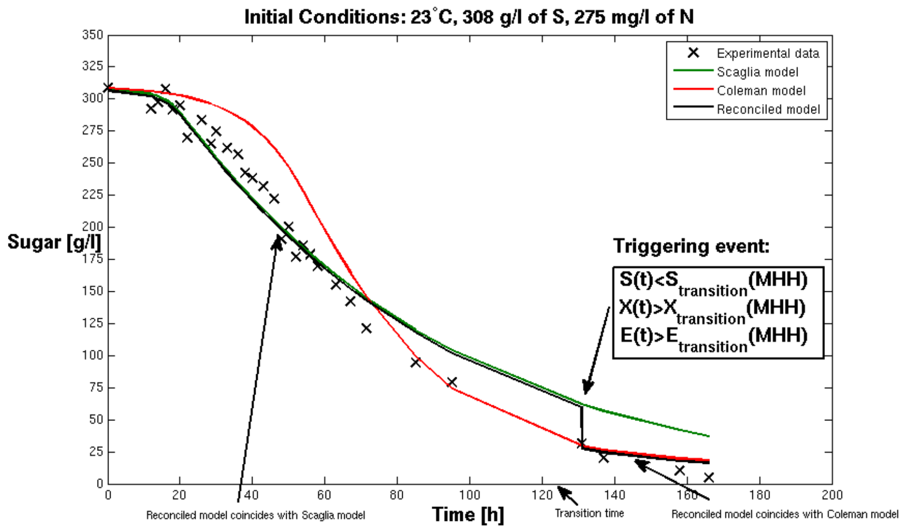


Figure 5.3: Reconciled model of wine kinetics for configuration MHH: temperature $23°C$, sugar $308.6[g/l]$, and nitrogen $275[mg/l]$. (A)Codes: MODEL describes the continuous part, MODES decides if the mode associated to Scaglia ($mode\_S$) or the associated to Coleman ($mode\_C$) is active, and RECONCILED MODEL OF SUGAR CONFIG MHH combines the results of the active models at each time. The change from transient to stable phase, triggered by a sufficiently low concentration of sugar and high concentrations of biomass and ethanol, is modeled to happen at time given by a gaussian variable with mean $TTime\_MHH = 121$ and standard deviation $sd\_MHH = 18.55$ (B)Simulation of sugar concentration.

the configuration MHH fit the biomass and sugar consumption dynamics correctly.

Equations 5.15, 5.17 take the form of the equations 5.18, 5.19. One observes that the main factor is the linear effect of X. In the transient phase there exists a quadratic effect of X, and X_A only affects the sugar consumption in the stable phase.

$$\frac{dX}{dt} = \mu^{(1)} \cdot X - \mu^{(2)} \cdot X^2 \tag{5.18}$$

$$\frac{dS}{dt} = -\big(1_{t\leqslant 121} \cdot (\sigma^{(1)} \cdot X - \sigma^{(2)} \cdot X^2) + 1_{t>121} \cdot \sigma_A \cdot X_A\big) \tag{5.19}$$

In this case, we use the structure of Figure 3.3*(A)*. The X $MODEL$ is coded in the node $MODEL$, with state variables X (biomass concentration), XA (active biomass), S (sugar concentration), $CO2$ (carbon dioxide) and $EtOH$ (ethanol). It interacts with the node $MODES$ that computes the values of the mode variables $mode\_S$ and $mode\_C$. These last ones are coefficients associated to the Scaglia (active only in the transient phase) and Coleman models that is activated at arriving to the stable phase. This corresponds to a hybrid system with coefficient switches. The continuous model is uniquely represented, and the effect of switching is modeled by changing the mode variables values present in $MODEL$. The node $RECONCILED\_MODEL\_OF\_SUGAR\_MHH$ links the continuous and discrete dynamics by using the outputs of the node $MODES$ (the modes) as inputs of $MODEL$. Whenever a mode variable is modified in $MODES$, it is modified in $MODEL$.

As illustrative example of strong switches, we show the reconciled model to predict sugar consumption if temperature and initial concentrations of sugar and nitrogen are high (Figure 5.4). That is to say, temperature between $28°C$ and $35°C$, sugar between $240[g/l]$ and $308[g/l]$, and nitrogen between $240[mg/l]$ and $551[mg/l]$. In this case, the time at which the system changes from transient to stable phase is modeled as a gaussian variable with mean $TTime\_HHH = 105.5$ and standard deviation $sd\_HHH = 19.125$. The simulations effectuated for the configuration HHH fit the sugar consumption dynamics correctly (Figure 5.4*(B)*).

For those conditions Coleman model is not active, Scaglia is considered active only in the stable phase (after 105.5 hours approximately) and Pizarro model is always active. The model of sugar consumption can be rewritten as the equation 5.20

$$\frac{dS}{dt} = -\big(\sigma^{(1)} \cdot X - 1_{t>105.5} \cdot \sigma^{(2)} \cdot X^2\big) \tag{5.20}$$

The approach with strong switches (Figure 3.3*(B)*) allows us to integrate the Pizarro model in the reconciled model for the configuration HHH. Each individual model is modeled in BioRica by a node, with S (sugar concentration) output variable. The Coleman and Scaglia models, with dynamics given by differential equation, use the syntax $eq : diff$ and are solved directly in BioRica. On the contrary, the Pizarro equations are externally solved and we used the syntax $eq : ext$. The node $MODES$ codes the dynamics of the mode variables $mode\_C$, $mode\_S$ and $mode\_P$ that define when each model is considered active to compute the

(A)

```
const Xinit=0.2;XAinit=0.2;Einit=0;Sinit=245.45;
      Ninit=0.28;CO2init=0;Glyinit=0;T=34;
      sd_HHH=19.125; TTime_HHH=105.5;
const KN=exp(-4.73);KS=exp(2.33);Y_ES=exp(-0.598);
     mu_max=exp(-3.92+0.0782*T);
     kprim_d=exp(-9.810-0.00108*T+0.00478*T^2);
     Y_XN=exp(3.50-3.61*10^(-3)*T);
     beta_max=exp(-2.30+0.0771*T);

formula mu;kd;beta; # SEE EXPRESSIONS IN COMPLETE CODE

node Coleman
        state X,XA,N,E,_S: FLOAT;
        flow
               X,E,S:FLOAT:out;
        eq:diff
                dX = mu*XA;
                dXA = (mu-kd)*XA;
                dN = -(mu/Y_XN)*XA;
                dE = beta*XA;
                dS = -(beta/Y_ES)*XA;
        assert
                S=_S;
        init
           X:=Xinit, XA:=XAinit, E:=Einit, _S:=Sinit,
           N:=Ninit;
edon

const a;b;c;d;e;A;B;C;D;E;F;G;H;I;mu,KS,beta,CO2_95;
      Y_XS;Y_CO2E;max_mu;  # VALUES IN COMPLETE CODE

formula max_nu_EtOH=exp(-2.30+0.0771*T);
     F1 = (exp(-(CO2-CO2_95))/(exp(CO2-CO2_95)+exp(-(CO2-
          CO2_95))))*A*mu*S/(S+KS*B^a);
     F2 = A*mu*S/((S+KS*B^a)*beta);
     F3 = 1-(exp(-(CO2-CO2_95))/(exp(CO2-CO2_95)+exp(-(CO2-
          CO2_95))));
     G1 = max_mu*S/(S+KS*B^b);
     H1 = G*mu*S/(S+KS*B^c);
     H2 = (S^2*(KS*B^e+KS*B^d)+2*S*(KS^2*B^d*B^e))/
          ((S+KS*B^d)^2*(S+KS*B^e)^2);
     H3 = (S^4+S^3*(KS*B^e+KS*B^d)+S^2*(KS*B^d*B^e))/
          ((S+KS*B^d)^2*(S+KS*B^e)^2);
```

```
node Scaglia
     state _S,X,CO2,E: FLOAT;
     flow X,E,S:FLOAT:out;
     eq:diff
          dS = (1/Y_XS)*(-X*(G1-E*X))-F*X;
          dX = F1*X*(1-X/F2)+F3*(C*X*dS-D*X);
          dCO2 = H1*X+H*mu*X*(dS*H2+dX*H3)+dX;
          dE = (1/Y_CO2E)*dCO2;
     assert
          S=_S;
     init X:=Xinit, _S:=Sinit, CO2:=CO2init, E:=Einit;
edon

node Pizarro
     state X,N,E,_S,Gly: FLOAT;
     flow S:FLOAT:out;
     eq:extern
     assert    S=_S;
     init
          X:=Xinit,N:=Ninit,E:=Einit,_S:=Sinit,Gly:=Glyinit;
edon

node MODES
     state _mode_C,_mode_S,_mode_P:FLOAT;stable:BOOL;
     flow
          mode_C,mode_S,mode_P:FLOAT:out;

     event change_phase;
     trans stable=False | change_phase
          -> _mode_S=1,stable=True;
     assert mode_C=_mode_C,mode_S=_mode_S,mode_P=_mode_P;
     init
          _mode_C:=0,_mode_S:=0,_mode_P:=1,stable:=False;

     extern law<change_phase>:Normal{TTime_HHH,sd_HHH};
edon

node RECONCILED_MODEL_OF_SUGAR_CONFIG_HHH

     flow S:FLOAT;
     sub Co:Coleman; Sc:Scaglia; Pi:Pizarro;Mo:MODES;
     assert
        S=(Co.S*Mo.mode_C+Sc.S*Mo.mode_S+Pi.S*Mo.mode_P)/
          (Mo.mode_C+Mo.mode_S+Mo.mode_P);
edon
```

(B)

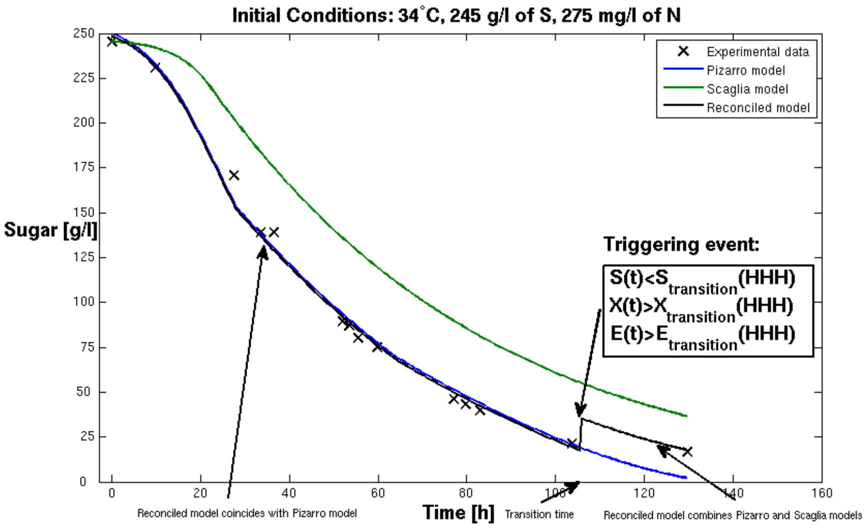


Figure 5.4: Reconciled model of wine kinetics for high initial conditions HHH: temperature $34°C$, sugar $245.45[g/l]$, and nitrogen $280[mg/l]$. (A)Codes: One BioRica node per original model, $MODES$ decides if each model is active or not, and $RECONCILED\_MODEL\_OF\_SUGAR\_CONFIG\_HHH$ combines the results of the active models at each time. The change from transient to stable phase, triggered by a sufficiently low concentration of sugar and high concentrations of biomass and ethanol, is modeled to happen at time given by a a gaussian variable with mean $TTime\_HHH = 105.5$ and standard deviation $sd\_HHH = 19.125$. (B)Simulation of sugar concentration.

sugar consumption. Initially only the Pizarro model is active, and in the stable phase Scaglia model becomes active by the transition *change_phase*. The node $RECONCILED\_MODEL\_OF\_SUGAR\_HHH$ links the continuous and discrete dynamics by computing S as a linear combination of the Coleman, Scaglia and Pizarro models with weights given by the mode variables.

As only the Scaglia model includes a competition effect in the sugar consumption equation ([10]), this reconciled system coincides with the expected behavior: initially the substrates are abundant, competition does not exist, but it appears when the resources become scarce in the stable phase.

5.5.2 Detecting sluggish and stuck fermentations

The reconciled models can be used to evaluate the fermentation process, according how the process *in vivo* agrees with the model. They can be used to detect sluggish (very slow) and stuck (with residual sugar) fermentations in time to save them. As is known, sluggish fermentations are promoted by insufficient temperatures and nitrogen levels, while the risk of stuck fermentations is increased by high levels of sugar and temperature ([17]). We consider that all the fermentations used to build the models do not present problems, excepting the configuration MMM which one observes sluggish fermentations because the moderate temperatures ($20°C$) and the limiting level of nitrogen ($66[mg/l]$) became the process slow. So, with this exception, our model simulates normal fermentations. We give some methods to evaluate the fermentation processes *in vivo* to detect sluggish and stuck fermentations. Insufficient levels of temperature and nitrogen promote slower fermentation rates and consequently the risk of sluggish fermentation, while the risk of stuck fermentation is increased when the levels of temperature and sugar concentrations are high.

If we consider fermentations with initial conditions in configuration MHH, the expected time to arrive to the stable phase is around 121 hours (Figure 5.3). Consequently, if before arriving to this transition time, the *in vivo* measures of sugar consumption are consistently slower than that expected in our model, we predict a risk of sluggish fermentation. This problem could be also detected by observing slow growth rates of the fermenting yeasts (biomass X). For configuration LHH or LMH one can do the same analysis to detect sluggish fermentations in time to save them.

On the contrary, for the configuration HHH (Figure 5.4), our dynamics estimations can be used to predict stuck fermentations *in vivo*. Since according to our results the transition to the stable fermentation phase happens at 105.5 hours (with standard deviation 19.125), one can evaluate the fermentation process by measuring the quantity of residual sugar. If the quantity of sugar even available is high when the process is arriving to the stable phase we have a risk of stuck fermentation. This problem could be detected in time to save the fermentations if one takes samples of the quantity of residual sugar to deduce that it becomes stable before the expected transition time.

5.6 Conclusions and discussion

Continuous dynamics are described by different types of models: differential equations, differential algebraic equations (DAEs), or other continuous models. The use of modules allows us to connect different models to describe complex behaviors by using input-output relations and mode transitions. To model discrete dynamics we considered transition systems theory. The discrete dynamics of the system are modeled by mode transitions, that can be deterministic, non-deterministic or stochastic. Both types of dynamics interact. So, changes in the continuous dynamics of the system are generated by mode transitions switching the continuous model, and the mode transitions can be provoked by specific conditions of the continuous variables. The mode transitions, transform the continuous model by changing only the values of coefficients (coefficient switches) or by modifying strongly the model (strong switches).

To model hybrid systems we used two abstraction schemas: the first one can models Hybrid systems with switches of coefficient values, and the second one models those with strong switches (Figure 3.3). Both are theoretically solved with the process in Figure 4.1, which is implemented with BioRica. We approached the reconciliation of competing model as an application case, and we illustrated it with the example of reconciliation of three wine fermentation kinetic models: Coleman ([28]), Scaglia ([94]) and Pizarro ([85, 92]).

We built a general method to combine models in function of configurations of factors. The method was applied to fermentation process modelling to explain the profiles of fermentation variables: concentration of yeast biomass, ethanol and sugar; by considering four factors: initial temperature, sugar and nitrogen, and growth phase. Our method starts with a symbolic step to homogenize the notation, for ODE models by rewriting into polynomial form and by identifying main and interaction effects. It continues with a statistical step to evaluate the models, as a function of experimental data (Pizarro *et al.* [85], Malherbe *et al.* [75], and Mendes-Ferreira *et al.* [79]). We defined discrete levels for each factor; for each configuration of factors and fermentation variable we statistically compared the results of the three kinetic fermentation models ([28, 85, 94]) with the experimental results and we obtained quality indexes of each model (Figure 5.2). After that we built a *combined model*, which selects the best resolution method for each fermentation variable and configuration of factors (Table 5.3). Finally, the reconciled models are described by Hybrid systems with coefficient switches for the configuration MHH (Figure 5.3) and strong switches for configuration HHH (Figure 5.4). This last reconciled model agrees with the expected result for sugar consumption with high temperature: initially the nutrients are abundant and competition does not exist, but it appears when the resources become scarce.

We described how to use the reconciled models to predict stuck and sluggish fermentations in time to save them. The models can be used as expected dynamics of normal fermentations, with the exception of configuration MMM that is considered sluggish fermentations. So, if for high temperatures and sugar concentrations (for example the configuration HHH) one detects that if the *in vivo* measures of residual sugar become stable before the expected transition time then one predicts stuck fermentations. On the contrary, for lower levels of temperature (such as configurations LHH, LMH and MHH), if before arriving to the expected transition time the *in vivo* measures of sugar consumption are consistently slower than the expected in our model (that is to say the residual sugar concentration is higher) then we predict sluggish fermentations.

Another challenge to obtain better approaches of the reality is to construct and to calibrate fermentation models that consider interacting yeast populations competing for resources. Although it has been observed that *Saccharomyces cerevisiae* is dominant in the majority of spontaneous alcoholic fermentations ([37, 93]) and it is the most popular yeast in inoculated cultures, there exist many strains and other yeasts as *Candida cantarellii* that participate in the process ([84, 116]) and that influence the aroma ([90]). The intervention of *lactic acid* and *acetic bacteria* in fermentations is also documented ([35, 34]). One can consider competition between individuals of the same population, modelled by logistic-like models, and interactions between different populations. A usual way to model the presence of two or more populations competing by resources is the *Lotka-Volterra-like models* ([98]). Different strains of *Saccharomyces cerevisiae* can present different levels of tolerance to ethanol, acidity, growth and death rate between other coefficients, Another fermenting yeast *Candida cantarellii* present different rates of growth, ethanol and glycerol yields [107]. In future works we will introduce the dependency of these rates with respect to yeast strains or species, and pH conditions.

Though BioRica syntax allowed to code in a clear form both types of Hybrid systems (Figures 5.3 and 5.4), our approach is independent of the implementation and it is possible to include models that are externally implemented.

Chapter 6

Application: modeling cell fate, particular case of bone precursors differentiation

In this chapter we explore cell fate decisions associated with the formation of specific cells. We want to model such decisions by considering the mechanisms that control them, to predict *in silico* the production of specific cell lineages and the formation of tissues. We focus in the case of bone and fat formation by modeling the dynamics of the differentiation of progenitor cells into osteoblasts and adipocytes.

Diseases such as osteoporosis affect a high percentage of the human population: one third of women and one twelfth of men over the age of 50 years suffer from osteoporosis, and the current treatments to increase bone mass or reduce resorption have many limitations and side effects ([54]). Moreover, obesity reduces bone density and is inversely associated with bone formation in osteoporosis ([26]), and there is a notorious decrease of the bone/fat formation ratio with the aging ([102, 19]). In this scenario, understanding the regulatory signaling pathways that are relevant during control of bone formation (*e.g.* Wnt-mediated signaling) has emerged as a critical component for treating in the future this and other bone disorders ([26, 69, 99, 70, 59, 54]). Moreover, it has become necessary to define their contribution within the regulatory processes that control the cell fate decisions responsible for going from bone precursor cells to bone tissue.

Many processes interact to control cell division, to regulate apoptosis, and to decide which cell lineages are produced. Despite the existing models for each individual process, models for the cross-talks and functional interactions between them have not been developed.

Here we describe *in silico* interactions between different regulatory processes leading to bone and fat tissue formation. Our combined model allow us to predict changes in bone or fat formation by stimulating (or inhibiting) the Wnt signaling pathway, the $PPAR\gamma$ pathway, the division of progenitor cells, and the apoptosis of progenitor or osteoblast cells. Based on this, we can analyze *in silico* the physiological responses to treatments of bone mass disorders based on the Wnt signaling pathway, and to explore the efficiency of new medical strategies before testing them in animal models.

Motivated by the recent model proposed by [96] and the results of [26], we defined the expression of $RUNX2$ as associated with the osteogenic differentiation ([69]), while $PPAR\gamma$ as associated with adipogenesis ([26]). Both transcription factors are mutually exclusive and autoregulated. This inter-regulated system is described by our main osteo-adipo switch model.

We describe the differentiation from osteo-adipo progenitor cells into osteoblasts and adipocytes by associating the main osteo-adipo switch model with a well-described model of the Wnt/β-catenin pathway ([62]) to stimulate the osteoblast lineage, and with a probabilistic model that describes the activation of the $PPAR\gamma$ pathway during stimulation of the adipocyte differentiation ([69], [26]). To accomplish this, we consider stimuli coefficients of the main osteo-adipo switch model as functions of the pathways activation. Finally, we include well-established validated models that reflect how cell division [108]) and apoptosis ([63]) are controlled.

Individual validated models are based on gene regulatory networks ([39, 31]), while stimulus interactions are described by the switched systems theory ([18, 100, 104]). Composition allows us to combine and reuse validated models, to give more complete descriptions of cell fate decisions, and to study the responses of the complete model to local variations, such as pathways activations or changes in cell death rates.

We implement our approach by using an available SBML models ([56]) of cell division ([108]), apoptosis ([63]), and Wnt pathway ([62]).

We use the BioRica framework to implement differentiation and apoptosis regulation of osteoblast cells. BioRica allowed us to translate SBML models into BioRica, to simulate switched systems, and to compose the different models in a non-ambiguous semantics. BioRica defines a global semantics for the composition of interacting modules, which it is preserved with respect to flow relations and event synchronization. Local clocks and solvers allow us to consider diverse types of models.

In the last section we discuss other models to consider. We describe the differentiation between red and white blood cells by reusing the model by Huang *et al.* ([55]) with BioRica, and composing them with a separated model that controls the stimulation of each cell lineage over time. The analysis of this system has applications in control of diseases and sports medicine.

6.1 Mechanisms of regulation

In multi-cellular organisms, inter and intra-cellular processes control metabolism ([44, 20]). The basic functioning of a cell can be explained by four essential processes: growth, division, differentiation and apoptosis ([2]). These processes are responsible for going from an unique cell to several specialized cell lineages. The cell decision between self-renewal, differentiation and apoptosis defines the cell fate ([110]).

The cell cycle is comprised by several events that leads to cell division itself (mitosis and cytokinesis). It is regulated by several proteins named *cyclins* and *cyclin dependent kinases* ([2]).

The process by which an undifferentiated cell acquires specialized functions is called *differentiation*. During this process the undifferentiated precursor cell, called progenitor, can differentiate into different specific lineages. Thus, hematopoietic (blood) cells differentiate from a common progenitor into red blood cells, to transport oxygen, or into white blood cells that have defensive functions in the organism ([55]). Osteoblasts (bone cells) and adipocytes (fat cells) share a common precursor derived from the bone marrow stromal cells ([26]).

The Wnt/β-catenin pathway plays an important role in the stimulation of bone formation ([69]). It promotes osteoblat differentiation, proliferation and mineralization, and blocks apoptosis and osteoclastogenesis. Consequently, it is fundamental during bone remodeling and repair, constituting a potential target for the treatment of bone mass disorders such as osteoporosis or to reduce adiposity or fracture risk ([59, 54]). As example, it has been shown that loss of function of *LRP4* and *LRP5* (Wnt receptors, [69]) is associated with osteoporosis ([71, 41]). The presence of some Wnt ligands activates the canonical Wnt pathway induces the accumulation of β-catenin in the nucleus of the cell, which interacts with a *TCF/LEF* transcription factor to activate the expression of the so called Wnt target genes ([53]). Some proteins that stimulate bone formation, such as *Runx2* considered here, are Wnt targets ([69]).

Apoptosis, programmed cell death, can occur during cell-cycle or of differentiation. It is controlled by a diverse range of cell signals, which may originate after either intrinsic or extrinsic inducers. Intracellular apoptosis begins in response to a stress such as heat, radiation, nutrient deprivation, viral infection, or membrane damage ([57]). Extracellular lethal signals include toxins, hormones, growth factors, nitric oxide or cytokines. In the case of bone cells, it has been shown that *homocysteine* strongly induces apoptosis in bone precursors and osteoblasts via the mitochondria pathway ([63]).

6.2 Gene Regulatory Networks and Switched Systems

In the Gene Regulatory Networks approach, for a given gene, we associate logic relations to define what other genes promote its expression and which additional genes inhibit it. These modulations depend on the expression of all of these genes: if the expression of a gene is sufficiently high (the gene is active) it promotes (or inhibits) the expression of its target gene ([39, 31]).

Here, we use ordinary differential equations to model Gene Regulatory Networks ([39], [31]). Many approximations are needed to use this type of model: mRNA molecules and proteins are represented by the same variables, and post-transcriptional regulations (common in eukaryotes) are not included. Nevertheless, this approach allows the study of a wide range of systems and to integrate microarray results.

Promotion and inhibition functions are described by Hill functions ([51]), sigmoidal curves used to describe influences between elements of a system. They measure the influence of an element on a target, depending on the concentration of the affecting element x, an exponent m to control the curve steepness, and on the mean point of influence θ ($h^+(x, \theta, m) = \frac{x^m}{x^m+\theta^m}$ for activation and $h^-(x, \theta, m) = 1 - h^+(x, \theta, m)$ for inhibition). Although Hill functions have their origin in biochemistry to describe the cooperative binding of oxygen to hemoglobin ([51]), they have applications in diverse areas from pharmacology ([43]) to thermodynamics ([72]).

Currently the metabolic control of gene regulatory networks is being considered as a way to obtain indirect interaction between genes, to obtain better descriptions of the underlying behaviors. In [12], it is divided between enzymatic effects and complex formations that are fast processes, and proteins synthesis and degradation are considered slow. So, the effect of metabolic processes is included within the gene regulatory networks by implicit functions that describe the fast dynamics of the metabolites and enzymes. In the theory, this idea works fine, but in practice one can obtain only boolean answers: at each time genes are expressing or not. With our approach, we look for connecting the activation of metabolic pathways with gene regulatory networks to obtain continuous models of the genes activity. One of the key elements of the model is how to include the activation of the pathways with the regulatory model, we consider stimulus coefficients that change their values to stimulate the expression of specific target genes as answer to the activation of the associated pathway.

We describe regulatory systems by composing continuous models, given by ordinary differential equations, with discrete models that control instantaneous coefficients changes. To include possible stochastic or non-deterministic changes of the differential equations and to allow the interaction between different regulatory systems, we use Switched systems (section 3). Continuous dynamics are described by differential equations ([18, 100, 104]), while stochastic transition systems ([29]) generate changes on the continuous model by changing coefficients associated with the mode variables into the continuous models.

The dependent variables of the model are called *state variables* (in analogy with Transition Systems), while continuous and discrete factors are considered *controllers*. These systems are described using a mixture of continuous, discrete dynamics and logical relations to allow multiple interacting components. In switched systems, the continuous dynamics are described by ordinary differential equations whose solution over time depends on the initial conditions. The discrete dynamics are given by the evolution of the *mode* variables. The changes in their values are called *mode transitions*. These are transitions in the sense of Transition Systems theory ([7]) and can be deterministic, non-deterministic or stochastic.

6.3 Approach

Our approach is based on reusing and composing validated models of cell division, differentiation and apoptosis to build a consolidated model that explains interactions that lead to bone and fat formation.

6.3.1 Reused models

We reuse a very well known model that simulates how cell division is regulated ([108]). In this model, cell division is triggered when a sufficiently high concentration of the active complex *MPF* (maturation promoting factor) complex is reached in the cell. As shown in 6.4, division occurs when the variable M (that represents the concentration of the active complex) reaches its threshold values.

With respect to apoptosis, we reuse the model proposed by [63] that explains how *homocysteine* strongly induces apoptosis of bone precursors and osteoblasts in a mitochondria-mediated manner.

In the case of cell differentiation, we reuse the osteo-chondro switch model proposed by [96] (Figure 6.1), but replacing chondrocyte lineage variable by adipocyte lineage. To simulate the induction of differentiation, we combined this model with other specific models that describe an engagement to the osteoblast lineage (by activation of the Wnt/β catenin pathway) or to the adipocyte lineage (by activation of the PPAR/γ oathway). Hence, we call the *main osteo-adipo switch model* the differentiation model before specifying the stimulus models, and the *osteo-adipo switch model* the model that includes the specific inductions to osteoblast and adipocyte lineages.

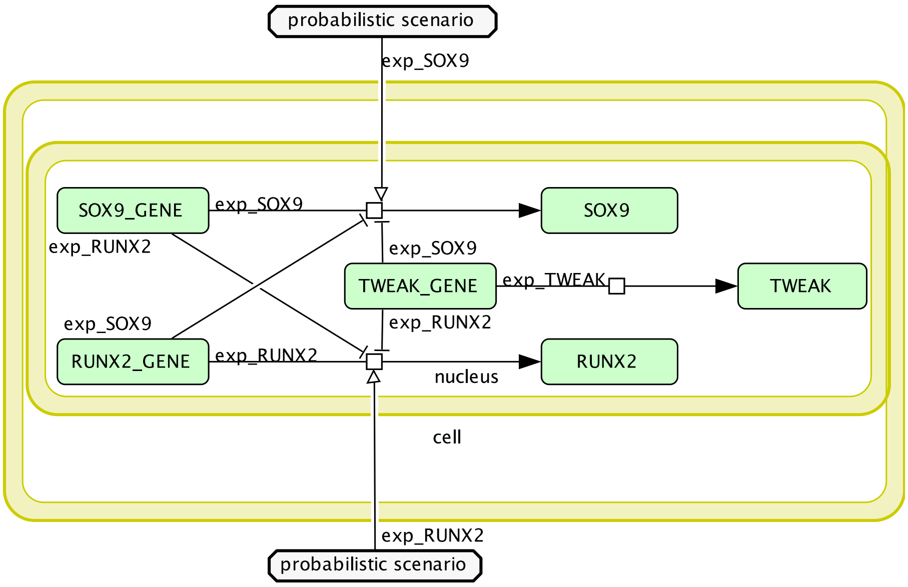


Figure 6.1: The osteo-chondro switch differentiation model by [96]. The expression of two mutually exclusive genes (*RUNX2* and *SOX9*) is associated with the bone and cartilage formation, respectively. The gene *Tweak* is the bio-marker for the progenitor state. Initially, an unknown stimuli that favors bone or cartilage formation is considered. Then, the outcomes are modeled by stochastic switches in the coefficients values of the model.

To simulate the activation of the Wnt/β-catenin pathway, we reuse the model by [62] performing the analyses under both control conditions and considering the activation of the Wnt pathway for a period of $500 - 1000$ minutes. On the other hand, the activation of the $PPAR\gamma$-pathway is stochastically simulated.

6.3.2 The main osteo-adipo switch model

Based on the osteo-chondro differentiation model by [96] (Figure 6.1) and the results described in [26], we developed the osteo-adipo switch system shown in Figure 6.2.

Progenitor cells differentiate into osteoblasts (bone cells) or adipocytes (fat cells). We associate two mutually exclusive genes (*Runx2*) and ($PPAR\gamma$) with engagement to the osteogenic or the adipogenic differentiation, respectively. Expression of progenitor bio-markers (*e.g.* *OCT4*, *SOX2* or both, [73]) is associated with maintenance in the uncommitted state that prevents the expression of *Runx2* and $PPAR\gamma$. The concentration of $mRNA$ associated with the progenitor state is denoted x_P, the $mRNA$ concentration of the osteogenic state is denoted x_O and that associated with the adipogenic differentiation is denoted x_A.

To incorporate the extracellular pro-differentiation, pro-osteogenic and pro-adipogenic stimuli, we considered three inputs: z_D, z_O and z_A with positive value. The effects that promote or inhibit expressions of regulated genes are incorporated by using variants of the common Hill functions. The model is represented by the equations 6.1-6.3, shown below. We have not considered the white noise introduced in the osteo-chondro switch model reported by [96].

$$\dot{x}_P(t) = \frac{a_P \cdot x_P^n + b_P}{m_P + z_D + c_{PP} \cdot x_P^n} - k_P \cdot x_P, \tag{6.1}$$

$$\dot{x}_O(t) = \frac{a_O \cdot x_O^n + b_O + z_O}{m_O + c_{OO} \cdot x_O^n + c_{OA} \cdot x_A^n + c_{OP} \cdot x_P^n} - k_O \cdot x_O, \tag{6.2}$$

$$\dot{x}_A(t) = \frac{a_A \cdot x_A^n + b_A + z_A}{m_A + c_{AA} \cdot x_A^n + c_{AO} \cdot x_O^n + c_{AP} \cdot x_P^n} - k_A \cdot x_A, \tag{6.3}$$

With $n = 2$, $a_P = 0.2$, $b_P = 0.5$, $m_P = 10$, $c_{PP} = 0.1$, $k_P = 0.1$, $a_O = a_A = 0.1$, $b_O = b_A = 1$, $m_O = m_A = 1$, $c_{OO} = c_{AA} = c_{OA} = c_{AO} = 0.1$, $c_{OP} = c_{AP} = 0.5$, $k_O = k_A = 0.1$ known parameters.

6.3.3 The osteo-adipo switch model: including the Wnt pathway as a stimulus for bone cell differentiation

Activation of the Wnt/β-catenin pathway stimulates the expression of *Runx2* ([69]) and therefore, bone cell differentiation. We introduced in the main osteo-adipo switch model (equations 6.1-6.3) the activation of the canonical Wnt pathway as positive stimulus for bone formation (Figure 6.1) and the activation of the $PPAR\gamma$ pathway that favors the adipocyte lineage. We build a composed model in which the coefficient z_O represents the activated state (it takes value 1) when the concentration of β-catenin is sufficiently high, and z_A represents the activated state when the $PPAR\gamma$ pathway is turned on.

As shown in Figure 6.2, the Wnt pathway can be externally activated by incubation of cells with lithium as well as by other treatments ([21, 54, 9]).

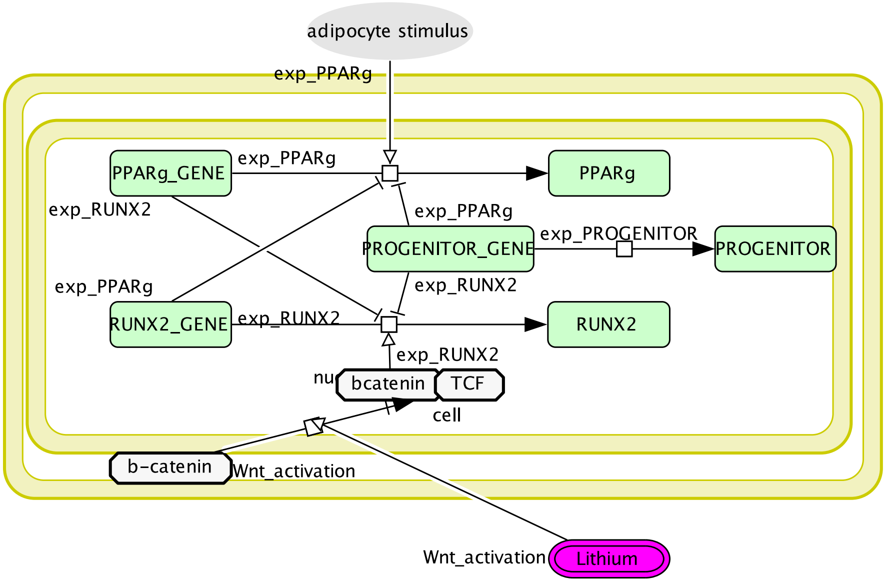


Figure 6.2: Osteo-adipo switch differentiation model. The expressions of two mutually inhibiting genes ($RUNX2$ and $PPAR\gamma$) are associated to the osteoblast and adipocyte lineage decision respectively, and a third bio-marker ($OCT4$ or $SOX2$) detects osteo-adipo progenitor cells. If the canonical Wnt pathway is active, stimulated by lithium, $\beta-$catenin goes to the nucleus of the cell and promotes the expression of $RUNX2$, which favors the bone formation. The activation of the $PPAR\gamma$ stimulates the formation of adipocytes.

6.3.4 Composing models

The differentiation stimuli into specific lineages are separately modeled and connected with the main osteo-adipo switch model by considering stimuli coefficients of the main differentiation model as functions of the activation of the pathways. We build the osteo-adipo switch model connecting by composition the canonical Wnt pathway model and a probabilistic model for the $PPAR\gamma-$pathway with the main osteo-adipo switch model, hence simulating in the system the presence of a biological stimulus that produces either bone or fat cells (Figure 6.2). We considered that the concentration of the β-catenin/TCF complex is sufficiently high to promote osteoblast lineage when it reaches 1.1 times its normal value ($8.81nM^{-1}$). That is to say, if the concentration of the β-catenin/TCF overcomes that threshold, then $z_O = 0.8$ in the main differentiation model.

Finally, we propose a consolidated model that describes cell fate decisions of bone precursor cells by considering division, differentiation and apoptosis models. We designate this as *the cell fate decisions model* (Figure 6.7). It connects the dynamic models of division and differentiation described here, with lineage-stimuli and apoptosis models (Figure 6.3). We consider as apoptosis stimulus the increase of *homocysteine* (*Hcy*) that leads to apoptosis of progenitors and osteoblasts ([63]). According to results obtained in *HS-5* osteoblastic lineage and primary human bone marrow stromal cells (both lines from ATCC, Manasas, VA, USA), when the concentration of *Hcy* is $10\mu M$ the apoptosis rates of osteoblasts (k_O) and precursors

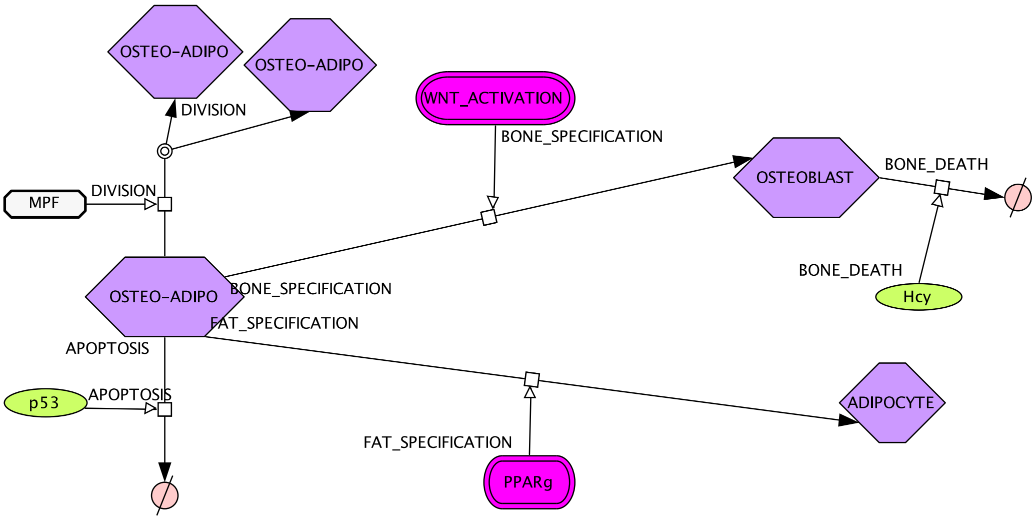


Figure 6.3: Composed model of cell fate. A progenitor cell can divide into new progenitor cells, or can die (apoptosis), or can differentiate into more specific lineages. Each one of these possibilities is controlled by a regulatory system. Apoptosis is stimulated by inducers, here quantified by increases of Hcy protein. Division is induced if the cell carries a sufficiently high level of maturation (called MPF, [108] model). Differentiation is stimulated by inhibitions of the progenitor maintenance role (associated to specific genes), after which that the cell decides its lineage. Each specific lineage is stimulated by signals: the activation of the canonical Wnt pathway (model in [62]) stimulates bone formation (osteoblast lineage), and the pathway of $PPAR\gamma$ the formation of fat (adipocyte lineage).

(k_P) are increased in 47% and 41% respectively on the osteo-adipo switch model (equations 6.1-6.3).

Therefore, our model for osteo-adipo progenitor cell differentiation is the result of the interaction of multiple regulatory models. The *DIFFERENTIATION MODEL* defines the dynamics in the concentrations of progenitor x_P, osteoblast x_O, and adipocytes x_A by a system of ordinary differential equations. The *STIMULI MODEL* defines the switches of the differentiation model by giving the values of the input parameters to *DIFFERENTIATION MODEL* (Figure 6.7).

Transitions in the *STIMULI MODEL* are controlled by the different models that regulate cell division, apoptosis and differentiation considered here. Each one of these regulatory models describes the dynamics of the signals that trigger these decisions. First, the cell division rate of progenitor cells a_P (input of the main osteo-adipo differentiation model) is obtained by simulating the cell division models and computing the period between divisions. Apoptosis of osteo-adipo progenitor cells and osteoblasts are stimulated by increases of Hcy ($homocysteine$) that promote apoptosis of osteoblasts by switching k_O to $1.47 \cdot k_O$, and of progenitor cells by switching k_P to $1.41 \cdot k_P$. Differentiation begins after an exponential time with mean of 100 minutes (z_D is switched to 1). The activation of the canonical Wnt pathway stimulates osteoblast lineage in the period of $500 - 1000$ minutes after the process is initiated (z_O is switched to 0.8), and the activation of the pathway of $PPAR\gamma$ (exponential distribution with mean 1333 minutes) stimulates the adipocyte lineage (z_A switched to 0.8).

6.3.5 Adjusting the model to human cells

In our cell fate decisions model (Figure 6.7), we adjusted parameters to represent the behavior of adult human bone precursor cells by assuming normal conditions according to literature. We considered that the doubling time of precursor cells is 86 hours ([32]), for osteoblasts is 72 hours (observed in *HS-5* cells, [97]) and for adipocytes is 78 hours (observed in liposuctioned adipose tissue, [32]). With respect to apoptosis coefficients, we considered that precursor cells have a lifespan between 6 and 9 months, osteoblasts have lifespan of 3 months ([76]) and the relative rate of adipocytes death is 10% per year (mean age of 10 years, [5]).

6.4 Results

We present implementations and simulations of regulatory models for cell division and differentiation. We analyze the results of our osteo-adipo switch model and cell fate model explained previously.

```
const k1=0.015; k2=0.0; k3=200; k4=180; k4prim=0.018; k5=0.0;k6=1.0;k7=0.6;
const  k8=1000000.0;k9=1000.0;
formula CT= C2+CP+pM+M; YT=Y+yP+pM+M; F=k4prim+k4*(M/CT)^2;

node main
         state C2,CP,pM,M,Y,yP:FLOAT;
         flow yt:FLOAT:out;
  eqdiff
    dC2 = k6*M -k8*C2 + k9*CP;
    dCP = -k3*CP*Y + k8*C2 -k9*CP;
    dpM = k3*CP*Y + -pM*F + k5*M;
    dM  = pM*F -k5*M -k6*M;
    dY  = k1 - k2*Y -k3*CP*Y;
    dyP = k6*M -k7*yP;

assert
         yt=YT;
init
         C2:= 0.0,M:= 0.0,CP:= 0.75,pM:= 0.25,Y:= 0.0,yP:= 0.0;
edon
```

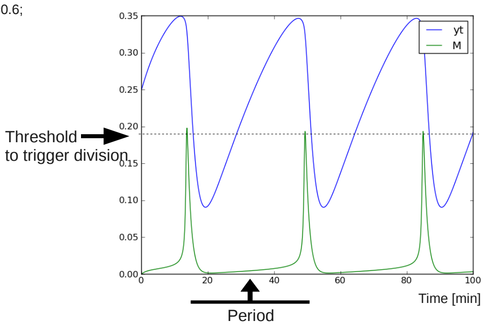


Figure 6.4: BioRica code and simulation of [108] model of regulation of cell division. Code is obtained from SBML models. Cell mitosis are predicted to happen when the maturation promoting factor MPF (variable M) reaches its threshold. Total *cyclin yt* and M are expressed as relative concentrations with respect to the total *cdc2* CT. Period is the temporal difference between two divisions.

6.4.1 Cell division

Our results (Figure 6.4) are in agreement with those obtained in the original publication ([108]). Cell division occurs when the variable M (that represents the concentration of the active complex) reaches its threshold values. We observed regular oscillations of the variables according to the repetitions of the cell-cycle. According to the original parameter values, the time period for a mitosis is around 35 minutes.

```
const n=2;ap=0.2;bp=0.5;mp=10;cpp=0.1;kp=0.1;ao=0.1;ac=0.1;bo=1;
const n=2;ap=0.2;bp=0.5;mp=10;cpp=0.1;kp=0.1;ao=0.1;ac=0.1;bo=1;
const bc=1;mo=1;mc=1;coo=0.1;ccc=0.1;coc=0.1;cco=0.1;cop=0.5;
const ccp=0.5;ko=0.1;kc=0.1;

node DIFF
      state xp,xo,xc:FLOAT;
      flow zd,zo,zc:[0,1]:in;
      eqdiff
            dxp=(ap*xp^n+bp)/(mp+zd+cpp*xp^n)-kp*xp;
            dxo=(ao*xo^n+bo+zo)/(mo+coo*xo^n+coc*xc^n+cop*xp^n)
                  -ko*xo;
            dxc=(ac*xc^n+bc+zc)/(mc+ccc*xc^n+cco*xo^n+ccp*xp^n)
                  -kc*xc;
      init
            xp:=12,xo:=0,xc:=0;
edon

node STIMULUS
      state _zd,_zo,_zc:FLOAT;on_d,on_o,on_c:BOOL;
      flow zd,zo,zc:FLOAT:out;
      event to_diff,to_osteo,to_chondro;
      trans
            on_d=False |-to_diff -> _zd=1,on_d=True;
            on_o=False |-to_osteo -> _zo=0.8,on_o=True;
            on_c=False |-to_chondro -> _zc=0.8,on_c=True;
      init
      _zd:=0,_zo=0,_zc=0,on_d:=False,ond_o=False,on_c=False;
     extern
            law<to_diff>:Exponential{0.01};
            law<to_osteo>:Exponential{0.002};
            law<to_chondro>:Exponential{0.001};
     assert
            zd=_zd;
            zo=_zo;
            zc=_zc;
edon

node MAIN
     sub
            S:STIMULUS;
            D:DIFF;
     assert
            D.zd=S.zd;
            D.zo=S.zo;
            D.zc=S.zc;
edon
```

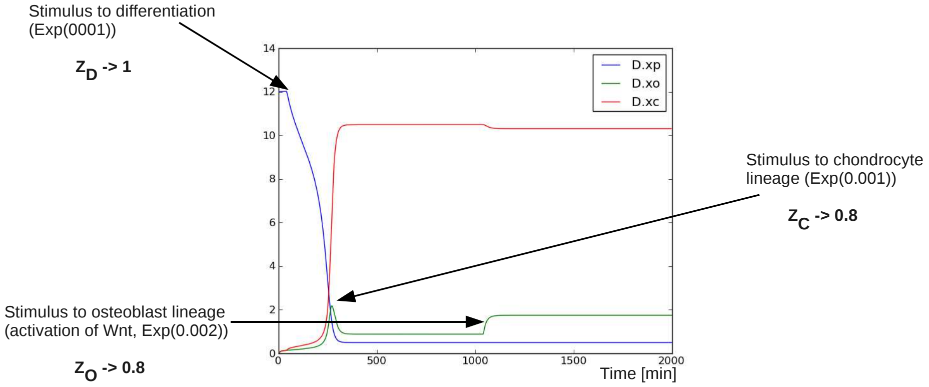


Figure 6.5: BioRica code and simulations of the osteo-chondro switch differentiation model in [96]). Stimuli are included by a probabilistic scenario externally modeled. Variable $D.xp$ denotes the concentration of the progenitor bio-marker, $D.xo$ of the osteoblast lineage and $D.xc$ of the chondrocytes lineage. The pro-differentiation stimulus happens at time exponential with expected value $t = 100$ minutes, the pro-osteogenic stimulus happens at time exponential with expected value $t = 500$ minutes, and the pro-chondrogenic stimulus with expected value $t = 1000$ minutes.

6.4.2 Cell differentiation

The osteo-chondro switch and the osteo-adipo switch models (Figures 6.5 and 6.6) were implemented by defining a BioRica node to describe the differential model ($DIFF$) and another one to describe the stimuli ($STIMULUS$) by computing the values of the parameters z. The node $MAIN$ describes the input-output connections between both nodes.

For the scenarios analyzed in the osteo-chondro switch model described in [96] (Figure 6.1), we obtained similar results, but the BioRica representation provides flexibility to the model. For example, in Figure 6.5 we consider another scenario, where pro-differentiation, pro-osteogenic and pro-chondrogenic stimuli occur with exponential probabilities over time (Poisson process, [112]). This corresponds to a switched system, in which delay times have random behaviors. We considered three possible switches. The system can be stimulated to favor the differentiation of progenitor cells, and to decide one or another lineage specification. In the simulation (Figure 6.5), near 100 minutes after initiated the process differentiation is activated.

```
node WNT

# FROM SBML CODE
# SEE SUPPLEMENTARY MATERIAL

edon

node DIFF

# FROM FIGURE 5, SCHITTLER SWITCH MODEL

edon



node STIMULUS
       state _zd,_zo,_za:FLOAT;
        flow zo:FLOAT:in; zd,za:FLOAT:out;
        event to_diff,to_adipo;
        trans

              _zd=0 |-to_diff -> _zd=1;
               _za=0 |-to_adipo -> _za=0.8;
        init
               _zd:=0,_za=0;

                 extern
                         law<to_diff>:Exponential{0.01};
                         law<to_adipo>:Exponential{0.001};
                assert
                      zd=_zd;
                      za=_za;


               edon

 node MAIN
         sub
             S:STIMULUS;
             D:DIFF;
             W:WNT;
                 assert
                      D.zd=S.zd;
                      S.zo=WNT_ACTIVE;
                      D.zo=S.zo;
                      D.za=S.za;

edon
```

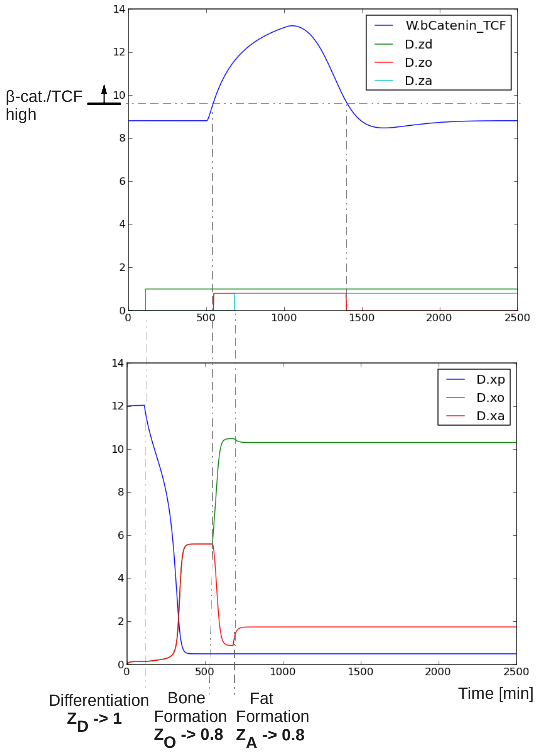


Figure 6.6: BioRica code and simulations of our osteo-adipo switch model. Node *DIFF* is the same of the osteo-chondro switch model, replacing x_C by x_A. The activation of the canonical Wnt pathway (model by [62]) is a stimulus to the osteoblast lineage, The activation of the $PPAR\gamma$ pathway, stimulates the adipocyte lineage, and it is simulated by an exponential probability distribution with mean 1000 minutes. In case of Wnt activation, the pro-osteoblast differentiation coefficient z_O increases to 0.8. For $PPAR\gamma$ it increases z_A.

The concentration of progenitor cells decreases and the concentrations of osteoblasts and chondrocytes increase. At approximately 250 minutes the formation of chondrocytes is stimulated, the concentration of *SOX9* (x_C) strongly increases, decreases the formation of osteoblasts (x_O) and the concentration of osteoblasts becomes stable. After approximately 1000 minutes the formation of osteoblasts is stimulated, its concentration increases and decreases the concentration of chondrocytes. After these transitions, the concentrations became stable.

We incorporate to the osteo-adipo switch model, with equations 6.1-6.3, the activation of the Wnt pathway as stimulus to bone formation (Figure 6.2). We automatically translate the SBML model by [62] (BioRica node *WNT*) to use the activation of the canonical Wnt pathway as stimulus to express $Runx2$ to favor differentiation into osteoblasts. We detect its activation by measuring the concentration of nuclear β-catenin/TCF. We consider that the pathway is active if it is sufficiently high (1.1 times its normal value of $8.81nM^{-1}$). The activation of the $PPAR\gamma$-pathway is simulated by a Poisson process ([112]) with mean 1000 minutes.

As shown in Figure 6.6, the system responds to the differentiation stimuli by

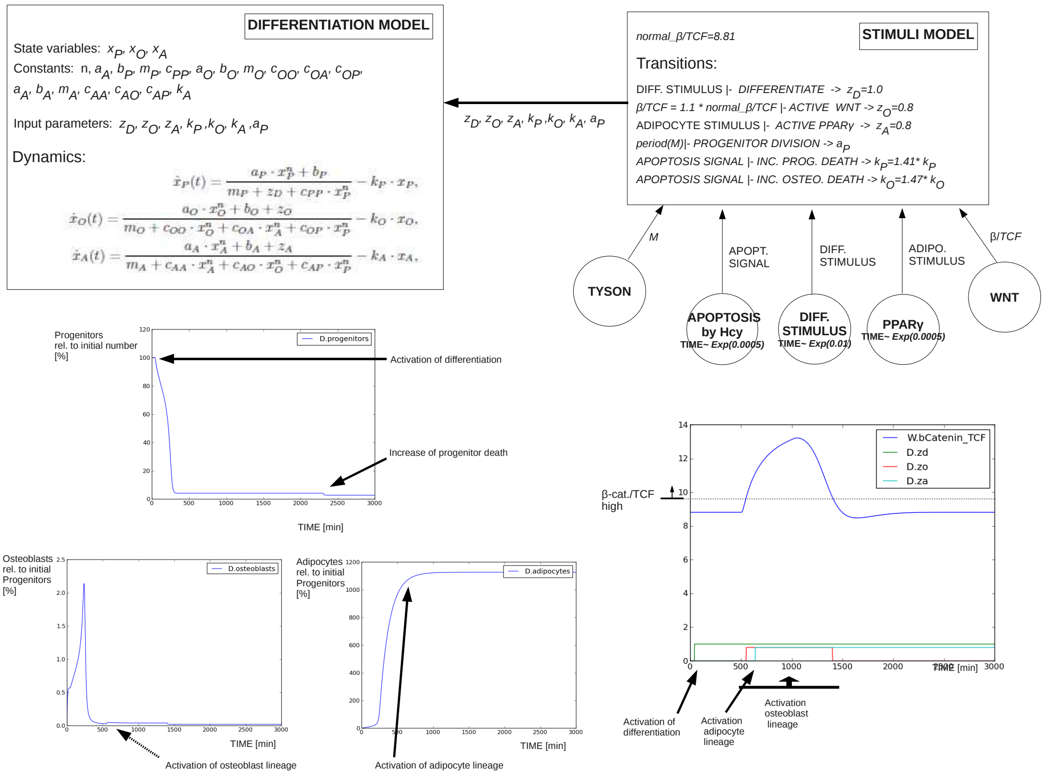

DIFFERENTIATION MODEL
State variables: x_P, x_O, x_A
Constants: n, $a_A, b_P, m_P, c_{PP}, a_O, b_O, m_O, c_{OO}, c_{OA}, c_{OP}, a_A, b_A, m_A, c_{AA}, c_{AO}, c_{AP}, k_A$
Input parameters: $z_D, z_O, z_A, k_P, k_O, k_A, a_P$
Dynamics:
$\dot{x}_P(t) = \frac{a_P \cdot x_P^n + b_P}{m_P + z_D + c_{PP} \cdot x_P^n} - k_P \cdot x_P,$
$\dot{x}_O(t) = \frac{a_O \cdot x_O^n + b_O + z_O}{m_O + c_{OO} \cdot x_O^n + c_{OA} \cdot x_A^n + c_{OP} \cdot x_P^n} - k_O \cdot x_O,$
$\dot{x}_A(t) = \frac{a_A \cdot x_A^n + b_A + z_A}{m_A + c_{AA} \cdot x_A^n + c_{AO} \cdot x_O^n + c_{AP} \cdot x_P^n} - k_A \cdot x_A,$
$z_D, z_O, z_A, k_P, k_O, k_A, a_P$
normal_β/TCF=8.81
STIMULI MODEL
Transitions:
DIFF. STIMULUS |- DIFFERENTIATE -> z_D=1.0
β/TCF = 1.1 * normal_β/TCF |- ACTIVE WNT -> z_O=0.8
ADIPOCYTE STIMULUS |- ACTIVE PPARγ -> z_A=0.8
period(M)|- PROGENITOR DIVISION -> a_P
APOPTOSIS SIGNAL |- INC. PROG. DEATH -> k_P=1.41* k_P
APOPTOSIS SIGNAL |- INC. OSTEO. DEATH -> k_O=1.47* k_O
M
APOPT. SIGNAL
DIFF. STIMULUS
ADIPO. STIMULUS
β/TCF
TYSON
APOPTOSIS by Hcy TIME~ Exp(0.0005)
DIFF. STIMULUS TIME~ Exp(0.01)
PPARγ TIME~ Exp(0.0005)
WNT
Progenitors rel. to initial number [%]
D.progenitors
120
100
80
60
40
20
0
500
1000
1500
2000
2500
3000
TIME [min]
Activation of differentiation
Increase of progenitor death
Osteoblasts rel. to initial Progenitors [%]
D.osteoblasts
2.5
2.0
1.5
1.0
0.5
0.0
TIME [min]
Activation of osteoblast lineage
Adipocytes rel. to initial Progenitors [%]
D.adipocytes
1200
1000
800
600
400
200
0
TIME [min]
Activation of adipocyte lineage
β-cat./TCF high
W.bCatenin_TCF
D.zd
D.zo
D.za
14
12
10
8
6
4
2
0
TIME [min]
Activation of differentiation
Activation adipocyte lineage
Activation osteoblast lineage

Figure 6.7: The cell fate decisions model: composed model of cell fate for osteo-adipo progenitor cells. Boxes and circles represent models here used, coded by BioRica nodes. Edges represent input-output relations. The *DIFFERENTIATION MODEL* defines the dynamics of x_P, x_O and x_A describing progenitors, osteoblasts, and adipocytes respectively. The *STIMULI MODEL* defines the transitions of the differentiation model by giving the update values to *DIFFERENTIATION MODEL*. The models in circles represent the models to regulate division and apoptosis. Apoptosis of precursor and osteoblast cells is stimulated by increasing *Hcy* ([63]) at exponential time with mean 2000 minutes. Division happens if the cell carries the maximal level of maturation, maximizing M ([108]). Differentiation is simulated to happen after an exponential time with mean 100 minutes. The activation of the canonical Wnt pathway (model in [62]) stimulates osteoblast lineage in the period $500 - 1000$ minutes, and the activation of the pathway of $PPAR\gamma$ (simulated by an exponential distribution of mean 1000 minutes) the adipocyte lineage. In simulations we show the percentages of each type of cells with respect with the initial quantity of progenitors.

changing its behavior. Initially, the expression patterns of progenitor-, osteoblast- and adipocyte-state bio-markers increase (variables x_P, x_O, x_A). After near 100 minutes the differentiation is stimulated: x_P decreases until becoming stable while x_O and x_A increase with the same rate. After that (near 500 minutes) the Wnt signal (β-catenin/TCF complex sufficiently high) is received to stimulate the bone formation: x_O increases its growth while x_A decreases. Finally, near 700 minutes the $PPAR\gamma$-pathway is activated: x_A increases while x_O decreases until both concentration becoming stable.

6.4.3 Consolidated models

We next estimated the dynamics of osteo-adipo progenitor cells, osteoblasts and adipocytes, in response to division stimuli, favoring each lineage specification (bone or fat formation), as well as to apoptosis stimuli. We show percentages of osteoblasts, adipocytes and progenitors relatives to the initial quantity of progenitors (Figure 6.7).

It was considered that initially there is a high presence of progenitors and absence of adipocytes and osteoblasts. As shown in Figure 6.7, the decrease of progenitor cells due to the differentiation stimulus is favored, arriving to the end, by the apoptotic action of *Hcy*. Adipocytes exhibit higher growth rates than osteoblasts, but bone formation is favored by the activation of the Wnt pathway during 500 minutes. The formation of fat is stimulated following around 600 minutes of the $PPAR\gamma$ pathway activity.

6.5 Conclusions and discussion

We have modeled the dynamics of cell fate decisions when going from osteo-adipo progenitor cells to bone (osteoblasts) and fat (adipocytes) cells. Bone and fat formation is controlled by many complex processes. Progenitor cells divide, differentiate into osteoblasts or adipocytes, or die, depending on regulatory processes. With the models here presented we can predict the changes in bone and fat formation by stimulating (or inhibiting) the Wnt pathway, the $PPAR\gamma$ pathway, the division of progenitor cells, and the apoptosis of progenitor or osteoblast cells. This allows us to simulate the physiological responses to treatments of bone mass disorders in silico, and to explore the efficiency of new medical strategies before testing them in mice or other animals *in vitro* or *in vivo*.

With respect to the theoretical basis, we used validated models of individual processes based on regulatory networks, switched systems to include stimuli effects, and composition between components to describe the interactions between the regulatory processes. We code, simulate and compose models using the BioRica framework (BioRica). BioRica allows us to describe switched systems, to reuse models described by the popular *SBML* format ([56]) providing higher flexibility to make future improvements and to reuse models.

To begin with cell differentiation, we have proposed the *osteo-adipo switch model*. Here, we consider the gene *RUNX2* as bio-marker for the osteoblast lineage and

$PPAR\gamma$ for the adipocyte lineage (Figures 6.2 and 6.6). Moreover, we introduced the canonical Wnt pathway to externally simulate a stimulus to the osteoblast lineage commitment, while the adipocyte lineage was stochastically simulated. As a result, we obtained a way to estimate the stimulatory effect of Wnt/β-catenin and $PPAR\gamma$ on bone and fat formation *in silico*. By means of these stimuli, we can switch the differentiation process to favor bone or fat formation and to analyze treatments developed to prevent or reduce bone mass disorders such as osteoporosis. Importantly, we can adjust this model to simulate the effect of other factors, such as aging, on the differentiation processes. While children show high tendency to regenerate bone, this process is slower in adults, who in turn form fat tissue with higher facility ([19, 102]). Therefore, this paradox could be assessed by incorporating the aging factor in the control functions z of our main osteo-adipo switch model.

Finally, we have considered a more complete description of the process of bone formation: the cell fate decisions model. We propose (Figures 6.3 and 6.7) a composed model that includes four essential cell processes: cell growth, cell division, cell differentiation and apoptosis. Our consolidated model integrates the [108] model that explains how cell division is regulated, the Wnt pathway-mediated (([62]) stimulation of the osteoblast lineage commitment, stochastic models to stimulate generic differentiation of the adipocyte lineage, and programmed cell death ([63]).

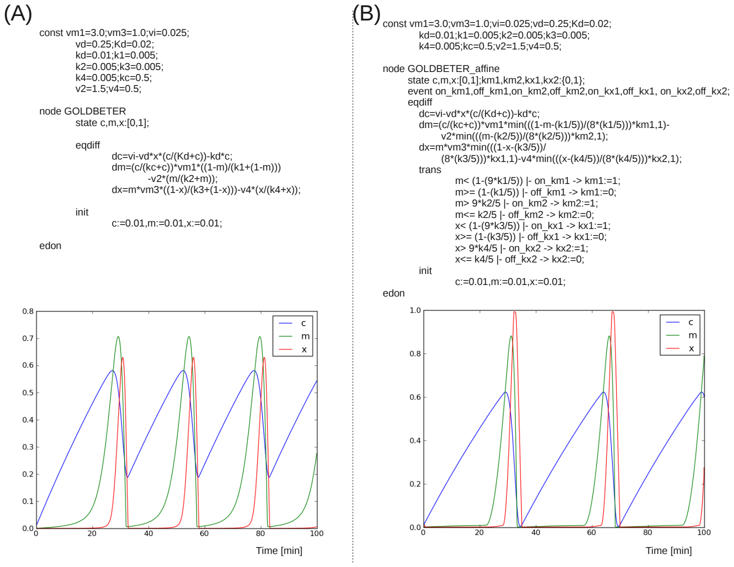
As a primary step, we have initially considered only a limited set of regulatory events during cell fate commitment. There are many other factors that affect life and death of bone cells ([76, 16]). Also, in the case of apoptosis, it is well known that factors such as the protein *sclerostin* (a bone morphogenetic protein that functions as a *BMP* antagonist) produce marked increase in caspase activity and apoptosis of human mesenchymal stem cells [103]. The tumor suppressor protein *p53* is involved in apoptosis too. In addition, we do not consider in our simulations the process of bone remodeling. Osteoclasts, in contrast to osteoblasts, are responsible of bone resorption. Osteoblasts form bone by differentiating into osteocytes or lining cells ([76]). As reported earlier ([69]) the canonical Wnt pathway has a also a regulatory role in all these aspects of bone remodeling.

Macarthur and colleagues ([73]) have presented an alternative model to explain differentiation towards osteoblasts, chondrocytes and adipocytes. They define a pluripotency circuit (provided by *OCT4*, *SOX2* and *NANOG*) to regulate precursor cells, which interacts with a mutually inhibiting network (provided by *RUNX2*, *SOX9* and $PPAR\gamma$) to regulate each lineage. In that case, stimuli are simulated by switching coefficients of the model and it is possible the same inclusion of stimuli models performed here.

In our model, we adjusted parameter values for adult human cells according to data in the literature ([76, 32, 97, 5]). Hence, for specific culture conditions of osteo-adipo progenitor cells, it may be necessary to re-adjust these parameters with stricter accuracy to allow the interacting models effectively describe its behavior. We may have to adjust the normal proportion of cells that differentiate into osteoblasts with respect to adipocytes and the ability of the Wnt pathway to selectively stimulate the expression of *Runx2*. Also, by taking advantage of reagents like *RGZ* (*rosiglitazone*) that have been shown to activate the $PPAR\gamma$ pathway and induce adipocyte formation ([111]), it will be possible cross-link the effects generated by the activation of both signaling pathways.

6.6 Other cell division and differentiation models

We describe, code and simulate another models to control cell division and differentiation not used in our application for bone formation: the [40] model to regulate cell division, and the model by [55] for differentiation of blood cells.



(A)

```
const vm1=3.0;vm3=1.0;vi=0.025;
        vd=0.25;Kd=0.02;
        kd=0.01;k1=0.005;
        k2=0.005;k3=0.005;
        k4=0.005;kc=0.5;
        v2=1.5;v4=0.5;

node GOLDBETER
        state c,m,x:[0,1];

        eqdiff
                dc=vi-vd*x*(c/(Kd+c))-kd*c;
                dm=(c/(kc+c))*vm1*((1-m)/(k1+(1-m)))
                        -v2*(m/(k2+m));
                dx=m*vm3*((1-x)/(k3+(1-x)))-v4*(x/(k4+x));

        init
                c:=0.01,m:=0.01,x:=0.01;

edon
```

(B)

```
const vm1=3.0;vm3=1.0;vi=0.025;vd=0.25;Kd=0.02;
         kd=0.01;k1=0.005;k2=0.005;k3=0.005;
         k4=0.005;kc=0.5;v2=1.5;v4=0.5;

node GOLDBETER_affine
       state c,m,x:[0,1];km1,km2,kx1,kx2:{0,1};
       event on_km1,off_km1,on_km2,off_km2,on_kx1,off_kx1, on_kx2,off_kx2;
       eqdiff
         dc=vi-vd*x*(c/(Kd+c))-kd*c;
         dm=(c/(kc+c))*vm1*min(((1-m-(k1/5))/(8*(k1/5)))*km1,1)-
               v2*min(((m-(k2/5))/(8*(k2/5)))*km2,1);
         dx=m*vm3*min(((1-x-(k3/5))/
               (8*(k3/5)))*kx1,1)-v4*min(((x-(k4/5))/(8*(k4/5)))*kx2,1);
         trans
                  m< (1-(9*k1/5)) |- on_km1 -> km1:=1;
                  m>= (1-(k1/5)) |- off_km1 -> km1:=0;
                  m> 9*k2/5 |- on_km2 -> km2:=1;
                  m<= k2/5 |- off_km2 -> km2:=0;
                  x< (1-(9*k3/5)) |- on_kx1 -> kx1:=1;
                  x>= (1-(k3/5)) |- off_kx1 -> kx1:=0;
                  x> 9*k4/5 |- on_kx2 -> kx2:=1;
                  x<= k4/5 |- off_kx2 -> kx2:=0;
         init
                  c:=0.01,m:=0.01,x:=0.01;
edon
```

Figure 6.8: BioRica codes and simulations of the Goldbeter model ([40]) of regulation of cell division. (A) The Goldbeter model. (B) The Goldbeter model simplified by piecewise-affine approximations of Hill functions ($h^+(C, K_d, 1)$ and $h^+(C, K_c, 1)$ are the original ones) by using the thresholds $\theta_1 = \frac{\theta}{5}$ and $\theta_2 = \frac{9\theta}{5}$. Cell mitosis are predicted to happen each time that the variable M (concentration of the maturation promoting factor MPF) reaches the threshold value. The observed period between mitosis is 25 minutes approximately.

6.6.1 Regulation of cell division by Goldbeter model

The Goldbeter model is a contemporary alternative to [108]. It ([40]) gives a minimal mathematical model for the so-called *mitotic oscillator* describing the interactions between *cyclin* and *cdc2 kinase* that control the maturation process arriving to the cell division. The model is formed by a regulatory network (equations 6.4-6.6) between C the *cyclin* concentration, M the active *cdc2 kinase* concentration, X the active *cyclin protease* concentration, $1 - M$ the inactive (*i.e.* phosphorylated)*cdc2 kinase* concentration and $1-X$ the inactive (*i.e.* dephosphorylated) *cyclin protease* concentration.

$$\dot{C}(t) = v_i - v_d \cdot X \cdot h^+(C, K_d, 1) - k_d \cdot C, \tag{6.4}$$

$$\dot{M}(t) = h^+(C, K_c, 1) \cdot V_{M1} \cdot h^+(1 - M, K_1, 1) - V_2 \cdot h^+(M, K_2, 1), \tag{6.5}$$

$$\dot{X}(t) = M \cdot V_{M3} \cdot h^+(1 - X, K_3, 1) - V_4 \cdot h^+(X, K_4, 1), \tag{6.6}$$

with original parameters $v_i = 0.025$, $v_d = 0.25$, $K_d = 0.02$, $k_d = 0.01$, $K_c = 0.5$, $V_{M1} = 3$, $K_1 = K_2 = K_3 = K_4 = 0.005$ and $V_{M3} = 1$ known parameters, and initial conditions $C = M = X = 0.01$. Hill functions given by $h^+(C, K_d, 1) = \frac{C}{K_d+C}$, $h^+(C, K_c, 1) = \frac{C}{K_c+C}$, $h^+(1-M, K_1, 1) = \frac{1-M}{K_1+(1-M)}$, $h^+(M, K_2, 1) = \frac{M}{K_2+M}$, $h^+(1-X, K_3, 1) = \frac{1-X}{K_3+(1-X)}$, and $h^+(X, K_4, 1) = \frac{X}{K_4+X}$. Here we show the simulation results with BioRica, considering the original parameter values (Figure 6.8). One observes that the period between mitosis is around 25 minutes.

Moreover, we use a piecewise-affine reduction to approximate $h^+(1-M, K_1, 1)$, $h^+(M, K_2, 1)$, $h^+(1-X, K_3, 1)$, and $h^+(X, K_4, 1)$. With this approximation, the concentration levels change but not the period (Figure 6.8(B)). Though, in comparison with [108] (Figure 6.4) this model gives an smaller period between mitosis (25 minutes compared with 35 minutes), one can modify the parameters values of both models to obtain the same period. Only the concentration levels are different in essence.

6.6.2 Cell differentiation of blood cells

Another differentiation system is that in which progenitor cells decide between white and red blood cells (Figure 6.9). The motivation to analyze this process is the control of diseases and sports medicine. One requires a bigger production of white blood cells to combat diseases, and an increase of oxygen transport (red blood cells) to realize sports activities.

Here we considered the differentiation model in [55], that describes the differentiation by using the bio-markers $GATA1$ and $PU.1$, for red blood cells and white blood cells formation respectively. The regulatory system is described by considering ordinary differential equations with Hill functions (equations 6.7-6.8). The mRNA concentration of the red blood cells marker is denoted x_R and the associated to the white blood cells is denoted x_W.

$$\dot{x}_R(t) = a_R \cdot h^+(x_R, \theta_{a_R}, n) + b_R \cdot h^-(x_W, \theta_{b_R}, n) - k_R \cdot x_R, \tag{6.7}$$

$$\dot{x}_W(t) = a_W \cdot h^+(x_W, \theta_{a_W}, n) + b_W \cdot h^-(x_R, \theta_{b_W}, n) - k_W \cdot x_W, \tag{6.8}$$

in which the coefficients a_R and a_W are related to the auto-stimulation rates, b_R and b_W introduce cross-inhibition, k_R and k_W are decay activity coefficients. The exponent $n = 4$, and the θs coefficients are influence thresholds. We model some scenarios in which one introduces an asymmetric dependence over time of each auto-stimulation rate (a_R for red blood cells and a_W for white blood cells). So, the stimuli for each lineage are given by modeling these parameters with the equations 6.9, 6.10.

$$\dot{a}_R(t) = -\lambda_R \cdot a_R, \tag{6.9}$$

$$\dot{a}_W(t) = -\lambda_W \cdot a_W, \tag{6.10}$$

with initial auto-stimulation rate $a_R(0) = a_W(0) = 1.0$, rates of exponential decrease λ_R and λ_W different according to the studied scenarios (Figure 6.9). For $\lambda_R = 0.7$ and $\lambda_W = 0.5$ (Figure 6.9, case (C)), the auto-stimulation rate of white blood cells becomes bigger than whose of red blood cells. That is to say, one introduces a stimulus to produce red blood cells (Figure 6.9).

```
const n=4; kR=1.0; kW=1.0; thetaW=0.5; thetaR=0.5; bW=1.0;
const bR=1.0; aW_0=1.0; aR_0=1.0;
const lambdaR=0.5; lambdaW=0.5; # case (A)

# const lambdaR=0.5; lambdaW=0.7; # case (B)
# const lambdaR=0.7; lambdaW=0.5; # case (C)

formula hposR=pow(xR,n)/(pow(thetaR,n)+pow(xR,n));
formula hnegR=pow(thetaR,n)/(pow(thetaR,n)+pow(xW,n));
formula hposW=pow(xW,n)/(pow(thetaW,n)+pow(xW,n));
formula hnegW=pow(thetaW,n)/(pow(thetaW,n)+pow(xR,n));
node DIFF
        state xR,xW:FLOAT;
        flow aR,aW:FLOAT:in;GATA1_REL,PU1_REL:FLOAT:out;
        eqdiff
                dxR=aR*hposR+bR*hnegR-kR*xR;
                dxW=aW*hposW+bR*hnegW-kW*xW;
        init
                xR:=1.5,xW:=0.2;
        assert
                GATA1_REL=xR/1.5;
                PU1_REL=xW/0.2;
edon

node STIMULUS
        state _aR,_aW:FLOAT;
        flow aR,aW:FLOAT:out;
        eqdiff
                d_aR=-lambdaR*_aR;
                d_aW=-lambdaW*_aW;
        init
                _aR:=1.0,_aW:=1.0;
        assert
                aR=_aR;
                aW=_aW;
edon

node main
        sub
                S:STIMULUS;
                D:DIFF;
        assert
                D.aR=S.aR;
                D.aW=S.aW;
edon
```

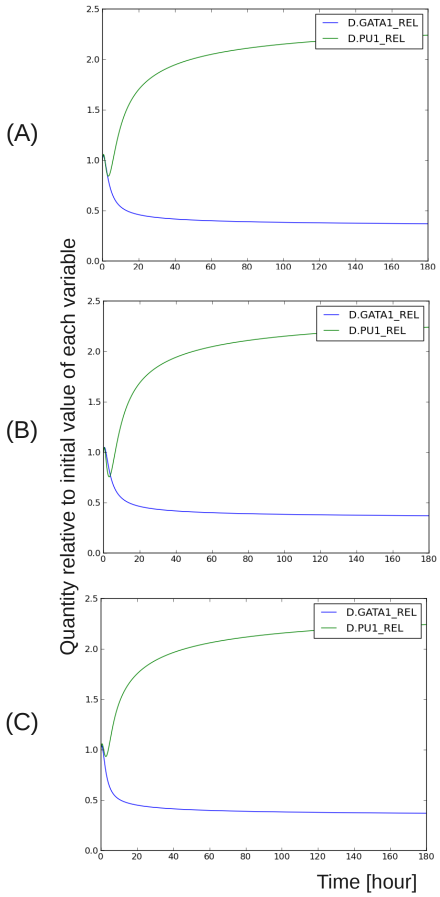


Figure 6.9: BioRica code and simulations of an red-white blood cells differentiation model in [55]). The pro-differentiation stimuli for each cell lineage (red and white blood cells) are modeled by considering the auto-stimulation coefficients as functions over time whose rate of exponential decrease λ differ. In case (A) both values are equal $\lambda_R = \lambda_W = 0.5$, in (B) $\lambda_R = 0.5$ and $\lambda_w = 0.7$, and in (C) $\lambda_R = 0.7$ and $\lambda_w = 0.5$. Parameters are $n = 4$, $k_R = k_W = b_R = b_W = 1$, and the θ's coefficients have value 0.5. As in [55], we show the relative quantities of x_R $(GATA1)$ and x_W $(PU.1)$ over time.

Chapter 7

Conclusions and discussion

7.1 Conclusions

In this work we have used the theory of Hybrid systems and composition to model biological systems, and we have presented abstractions capable of modeling different levels of changes in the behavior laws of the system, reusing and reconciling models. Continuous models describe the dynamics of variables of the system with gradual changes over time, and switches are used to change the laws of the behavior or for combining different models.

We formalized the theory of hybrid systems by including stochastic and non-deterministic model changes, and defining two types of model transformations: with coefficient switches and with strong switches (section 3.3). During this work we have contributed to the development of the new implementation of BioRica, which includes important improvements to be capable of modeling real biological systems. We detected the problems of the previous versions, proposed and checked possible solutions. As result, with the new version we are capable of describing biological systems as hybrid systems that integrates continuous, discrete, deterministic, stochastic and non-deterministic behaviors. As result, we developed an approach with theoretical substance and good results in practice that allows modeling, implementation and simulation of real biological applications such as the case of wine fermentation kinetics, and cell fate decisions in bone formation.

We described three steps of our approach: modeling, solving and implementing. We summarize them here.

In the modeling step, we began by formalizing some aspects of the hybrid systems theory. The continuous dynamics are described by different types of models: differential equations, differential algebraic equations (DAEs), or other continuous models. The use of modules allows us to connect different models to describe complexes behaviors by using input-output relations and the mode transitions. To model discrete dynamics we considered Transition Systems theory. The discrete dynamics of the systems are modeled by mode transitions that can be deterministic, non-deterministic or stochastic. Both types of dynamics interact. So, changes in the continuous dynamics of the system are generated by mode transitions switching the continuous model, and the mode transitions can be provoked by specific conditions of the continuous variables. Mode transitions transform the continuous model by changing only the values of coefficients (coefficient switches) or by modifying strongly the model (strong switches). With this theory one can consider different levels of model changes according to the modeling decisions, one uses coefficient switches and modifies the models to include coefficients associated to mode transitions, or one uses strong switches and the models are directly reused.

In the solving step we presented the computation steps needed to solve hybrid systems (Figure 4.1). After preprocessing the initial conditions and parameters of the system, begin a solving/updating process of the model. At any given time the model is solved, and the model is updated (by modifying coefficients values or strong switches) when detecting the guard condition $GUARD(x, mode, time)$. The process finishes when the condition $END(x, mode, time)$ is verified. In the third step we implemented and simulated the models with BioRica.

These considerations allow us to model a wide range of biological systems: from processes where the information levels are high to those in which it is very poor, from continuous models without changes over time to other ones in which the behavior changes force to change completely of model type. From the theoretical point of view, we obtained some assume-guarantee rules with respect to properties of Hybrid systems and compositions of them. The stability of Switched systems is assured given the conditions of Theorem 6 in [18]. Compositions of Hybrid systems are Hybrid systems, and refinements of a module forming part of a composed system result in refinements of the composed system. With our approach and the implementation in BioRica we assure the consistency of composed models with respect to the values of shared variables.

The development of the theoretical basis of our approach was complemented with the improving of the BioRica framework. We identified problems of the previous BioRica version, and proposed solutions that contributed to the development of important improvements included in the current version with the aim of modeling real biological systems, composing models and allowing hybrid systems. We identified and modeled biological systems no-implementable with the previous BioRica version. For discrete systems, we established the need to define transitions computations using not only the basic algebraic operations and integer variables. To describe more general systems was necessary to include continuous elements, which carried to the use of hybrid systems. We gave a BioRica specification of Hybrid Systems and the reconciliation of competing models that assures the integrity of individual models, obtained a general implemented of Switched Systems by using two types of model changes: coefficient switches and strong switches, and proposed a new BioRica syntax to code general Hybrid Systems.

We approach themes and application of current interest: composition and reusing of models, and coupling of metabolic pathways and Gene Regulatory Networks. Composition of validated models is a known way to build complete descriptions of complex biological systems ([109, 33]). Our approach and the BioRica framework allow us to solve this need, to combine different types of models by implementing the notion of composition and importing SBML models. The connection between metabolic changes and Gene Regulatory Networks ([12]) has been recently considered as a way to obtain indirect interaction between genes, to obtain better descriptions of underlying behaviors. Here we approach that problem to model cell fate decisions leading to bone formation by considering a Gene Regulatory Network and specific pathways to stimulate the expression of target genes. We developed two biological applications: reconciliation of competing models in case of wine fermentation, and modeling of cell fate decisions leading bone and fat formation.

We approached the reconciliation of competing model as an application case of Hybrid System, and we illustrated it with the example of reconciliation of three wine fermentation kinetic models: Coleman ([28]), Scaglia ([94]) and Pizarro ([85, 92]). We built a general method to combine models in function of configurations of factors. The method was applied to fermentation process modeling to explain the profiles of fermentation variables: concentration of yeast biomass, ethanol and sugar; by considering four factors: initial temperature, sugar and nitrogen, and growth phase. We described how to use the reconciled models to predict stuck and sluggish fermentations in time to save them. The models can be used as expected dynamics of normal fermentations, with the exception of configuration MMM that is considered sluggish fermentations.

Switched systems and gene regulatory networks allowed us to build good descriptions of basic cell processes by defining interactions between components and the possibility of diverse types of behaviors. We showed how to build extension of models by using the composition of the cell differentiation models with models that control the stimuli, such as the canonical Wnt pathway to favor the osteoblast lineage. BioRica allowed us to describe switched systems, coding differential equations to model the continuous dynamics and transitions for the discrete dynamics, and integrating models in the *SBML* format ([57]). We proposed (Figure 6.3) a composed model to integrate four essential cell processes: growing, division, differentiation and apoptosis. It integrates the Tyson model to regulate the division and generic pathways to stimulate formation of cell lineages. In the case of osteo-adipo progenitor cells, we use this model and considers the Wnt pathway to stimulates the osteoblast lineage.

7.2 Future work

To continue we discuss some possible extensions and improvements of our approach. Extensions are related with the type of Hybrid Systems that we can model and simulate, and the Transition Systems theory. The improvements are related with the BioRica implementation and the numerical solution of differential equations, currently we use Runge-Kutta $(4, 5)$ implemented by the *Scipy* library of *Python*, and the inclusion of other continuous models.

Our approach is based on the use of Stochastic Transition Systems ([29]) to modeling the mode transitions (the changes in the form of the model). The recent years the theory of Transition Systems has been extended to include continuous spaces of events and more general types of schedulers ([25, 113]). These works give insights about how extend our approach (section 2.2). As we explained here, the approach is valid for general Hybrid Systems, with different types of continuous models. To implement them with BioRica, we propose to review some specific Python libraries to solve differential algebraic equations and partial differential equations. The inclusion of bridges to other software such as Matlab has not showed satisfactory results.

There are two known simulating problems that we considered: *stiffness* and *mode transitions not happening at the considered times.* For differential equations that appear in Switched Systems as continuous models, the classic methods to numerically approximate their solutions present problems with stability in stiff equations. One requires many iterations, incrementing the computation time, to obtain good approximations of the solutions. We solved in part this problem by considering separation into modules. So, stiff models can be isolated to spend more time or to use strong solvers just in their solutions. However the selection of the best solvers to avoid these problems is a key point.

Moreover, the presence of the mode variables *mode* adds another problem: What do we do if at time t it changes the value of *mode* but t does not belong to $\{t_0, t_1, \ldots, t_k, t_{k+1}, \ldots\}$? It is not the aim of this work to face these simulation problems with BioRica. Between the classic methods, we use *Runge-Kutta* $(4, 5)$ that has good responses to the stiffness problem and we did not observe such problems. To consider mode transitions that happen between the discrete times, BioRica interpolates the state variable values at the transition time, do the transition and restart the continuous dynamics. In future works we will explore the QSS approach ([68]), that promises to avoid both problems taking advantage of piecewise definitions.

To finish we want to discuss about the basis, the goals and the scopes of our approach. One of the arguments of our approach was 'Biology is modular', that supports the use of modules and composition of them to model biological systems. We argued that this is an open question, but practice shows that it would be true. However, if that is true or not is not really important. The fact is that the human understanding of Biology, the description of biological systems, is modular.

The main goal of this work was to build an approach, theoretical, sustainable, and usable. We considered applicability of the approach as the final aim. So, we use the theory here presented as basis and the BioRica framework as tool to be capable of modeling, implementing and simulating real biological systems, to do *in silico* analysis that captures emerging properties not observed before. Composition and hybrid systems theory gave us the theoretical sustenance to integrate diverse types of models describing a biological systems with a non-ambiguous semantics. BioRica allowed us to implement these descriptions, to simulate the systems behaviors and to analyze them.

We showed that our approach has real applications in Biotechnology and Medicine. We built usable models to describe the wine fermentation kinetics and the cell fate decisions leading to bone and fat formation form precursor cells.

In the case of wine fermentation, we reused and reconciled known models that describe over time the responses of yeasts cells to different temperatures, quantities of resources and toxins. We chose, depending on the initial conditions and fermentation variable, the model with best adjustment to experimental data. The importance of this application is that the resulting model can be applied to predict process problems as stuck and sluggish fermentations, to avoid them ([10]). It is clear that this idea can continue being developed to consider more models with better adjustments in different conditions such as different types of fermenting yeasts.

With respect to our application on modeling cell fate decisions leading to bone and fat formation, the idea is very ambitious. By considering accurate models to predict the bone and fat formation in response to activation of pathways (such as the Wnt pathway), and factors affecting these functions (such as increments of Homocysteine), one can analyze *in silico* the responses to treatments for osteoporosis and other bone mass disorders. We think that we are giving a first step to do that. With a more extensive inclusion of factors associated to bone and fat formation and stricter experimental adjustments, continuing the work here exposed, it is possible to obtain accurate models that give *in silico* evaluations of medical treatments of bone mass disorders before testing them *in vitro* or *in vivo*.

With respect to future works, we think that an important point to explore is modeling metabolic processes as hybrid systems, and combining them with gene regulatory networks. The dynamics of metabolic processes could be the key to obtain more precise models of biological systems. In the case of wine fermentation kinetics, the glycolysis of yeast has a key role to produce ethanol. The dynamics of this process depends on the type of yeast (species, strain), and environmental conditions such as levels of sugars, ethanol, carbon dioxide and temperature. As we explained before (section 6.2, [12]), metabolic control of gene regulatory networks is used to obtain indirect interaction between genes, to obtain better descriptions of the underlying behaviors. We think that our approach, as applied in section 6, is an appropriate way to include interactions between metabolic processes and genic regulation.

Two known approaches to model the dynamics of metabolic processes are Dynamic Flux Balance Analysis (DFBA, [14, 74]), and the use of discrete event formalisms ([13, 27]). DFBA approach considers time steps, and starting with the initial conditions at each time uses FBA to predict the growth, nutrient uptake and by-product secretion rates to calculate the new nutrient concentrations that limit the next FBA solution. In [4] the discrete event formalisms DEVS ([115]) is used to model glycolysis considering reactions as event transitions. We wish to use our approach to incorporate important aspects to these approaches: randomness, non-determinism, effect of environmental conditions and composition of processes.

Bibliography

[1] Kazuyuki Aihara and Hideyuki Suzuki. Theory of hybrid dynamical systems and its applications to biological and medical systems. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 368(1930):4893 –4914, November 2010.

[2] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. *Molecular Biology of the Cell, Fourth Edition*. Garland Science, 4 edition, March 2002.

[3] R Alfieri, E Bartocci, E Merelli, and L Milanesi. Modeling the cell cycle: From deterministic models to hybrid systems. *Bio Systems*, March 2011. PMID: 21453748.

[4] Thierry Antoine-Santoni, Fabrice Bernardi, and François Giamarchi. General Methodology for Metabolic Pathways Modeling and Simulation using Discrete Event Approach. Example on glycolysis of Yeast. In *Proceedings of BIOCOMP 2007*, volume 1, pages 58–64, Las Vegas, États-Unis, June 2007. CSREA Press. 7 pages.

[5] Erik Arner, Pal O. Westermark, Kirsty L. Spalding, Tom Britton, Mikael Rydén, Jonas FrisÃ©n, Samuel Bernard, and Peter Arner. Adipocyte turnover: Relevance to human adipose tissue morphology. *Diabetes*, 59(1):105–109, January 2010. PMID: 19846802 PMCID: 2797910.

[6] André Arnold. Synchronized products of transition systems and their analysis. In *Application and Theory of Petri Nets 1998*, pages 26–27. Springer, 1998.

[7] André Arnold, Didier Bégay, and Paul Crubillé. *Construction and analysis of transition systems with MEC*. World Scientific Publishing Co., Inc., 1994.

[8] André Arnold, Gérald Point, Alain Griffault, and Antoine Rauzy. The AltaRica formalism for describing concurrent systems. *Fundam. Inf.*, 40(2-3):109–124, 1999.

[9] Macarena S. Arrázola, Lorena Varela-Nallar, Marcela Colombres, Enrique M. Toledo, Fernando Cruzat, Leonardo Pavez, Rodrigo Assar, Andrés Aravena, Mauricio González, Martín Montecino, Alejandro Maass, Servet Martínez, and Nibaldo C. Inestrosa. Calcium/calmodulin-dependent protein kinase type IV is a target gene of the <I>Wnt</I>/beta-catenin signaling pathway. *Journal of Cellular Physiology*, 221(3):658–667, 2009.

[10] Rodrigo Assar, Felipe Vargas, and David James Sherman. Reconciling competing models: a case study of wine fermentation kinetics. In Katsuhisa Horimoto, Masahiko Nakatsui, and Nikolaj Popov, editors, *Algebraic and Numeric Biology 2010*, volume 6479 of *Springer LNCS*, pages 68–83, Hagenberg, Austria, August 2010.

[11] Guillaume Bal and Leonid Ryzhik. Stability of time reversed waves in changing media. *DISC. CONT. DYN. SYST. A*, 12:793–815, 2005.

[12] Valentina Baldazzi, Delphine Ropers, Yves Markowicz, Daniel Kahn, Johannes Geiselmann, and Hidde de Jong. The carbon assimilation network in escherichia coli is densely connected and largely Sign-Determined by directions of metabolic fluxes. *PLoS Comput Biol*, 6(6):e1000812, June 2010.

[13] F.J. Barros, M.T. Mendes, and B.P. Zeigler. Variable DEVS-variable structure modeling formalism: an adaptive computer architecture application. In *Fifth Annual Conference on AI, and Planning in High Autonomy Systems*, pages 185–191, Gainesville, FL, USA, 1994.

[14] Scott A Becker, Adam M Feist, Monica L Mo, Gregory Hannum, Bernhard O Palsson, and Markus J Herrgard. Quantitative prediction of cellular metabolism with constraint-based models: the COBRA toolbox. *Nature Protocols*, 2(3):727–738, 2007. PMID: 17406635.

[15] Frank Beichelt. *Stochastic processes in science, engineering, and finance.* Chapman & Hall/CRC, 2006.

[16] John P. Bilezikian, Lawrence Gideon Raisz, and Gideon A. Rodan. *Principles of bone biology.* Elsevier, 2002.

[17] Linda F. Bisson. Stuck and sluggish fermentations. *Am. J. Enol. Vitic.*, 50(1):107–119, March 1999.

[18] Michael S. Branicky. Stability of switched and hybrid systems. *Decision and Control, 1994., Proceedings of the 33rd IEEE Conference on Decision and Control*, 4:3498–3503, December 1994.

[19] H. Brockstedt, M. Kassem, E.F. Eriksen, L. Mosekilde, and F. Melsen. Age- and sex-related changes in iliac cortical bone mass and remodeling. *Bone*, 14(4):681–691, July 1992.

[20] Feliksas Bukauskas. *Intercellular communication.* Manchester University Press ND, May 1991.

[21] Yongping Cai, Alexander B Mohseny, Marcel Karperien, Pancras C W Hogendoorn, Gengyin Zhou, and Anne-Marie Cleton-Jansen. Inactive wnt/beta-catenin pathway in conventional high-grade osteosarcoma. *The Journal of Pathology*, 220(1):24–33, January 2010. PMID: 19882675.

[22] Werner Callebaut and Diego Rasskin-Gutman. *Modularity: Understanding the Development and Evolution of Natural Complex Systems*. The MIT Press, illustrated edition edition, June 2005.

[23] J. G. Carbonell, editor. *Machine learning: paradigms and methods*. Elsevier North-Holland, Inc., 1990.

[24] Franck Cassez. Vérification qualitative – Model-Checking et Logiques Temporelles. Actes de l'école d'été ETR'03, 2003. Toulouse.

[25] Stefano Cattani, Roberto Segala, Marta Kwiatkowska, and Gethin Norman. Stochastic transition systems for continuous state spaces and non-determinism. *FOSSACS 2005 LNCS 3441*, pages 125–139, 2005.

[26] Jin-Ran Chen, Oxana P. Lazarenko, Xianli Wu, Yudong Tong, Michael L. Blackburn, Kartik Shankar, Thomas M. Badger, and Martin J. J. Ronis. Obesity reduces bone density associated with activation of ppargamma and suppression of wnt/beta-catenin in rapidly growing male rats. *PLoS ONE*, 5(10):e13704, October 2010.

[27] A. C.-H. Chow. Parallel DEVS: a parallel, hierarchical, modular modeling formalism and its distributed simulator. *Trans. Soc. Comput. Simul. Int.*, 13(2):55–67, 1996.

[28] Matthew C. Coleman, Russell Fish, and David E. Block. Temperature-Dependent kinetic model for Nitrogen-Limited wine fermentations. *Applied and Environmental Microbiology*, 73(18):5875–5884, September 2007. PMID: 17616615 PMCID: 2074923.

[29] Luca De Alfaro. Stochastic transition systems. *Proceedings CONCUR 98*, 1466:423—438, 1998.

[30] Luca De Alfaro, Thomas A Henzinger, and Ranjit Jhala. Compositional methods for probabilistic systems. *Proceedings CONCUR 2001*, 2001.

[31] Hidde De Jong. Modeling and simulation of genetic regulatory systems: A literature review. *JOURNAL OF COMPUTATIONAL BIOLOGY*, 9:67–103, 2002.

[32] Daniel A De Ugarte, Kouki Morizono, Amir Elbarbary, Zeni Alfonso, Patricia A Zuk, Min Zhu, Jason L Dragoo, Peter Ashjian, Bert Thomas, Prosper Benhaim, Irvin Chen, John Fraser, and Marc H Hedrick. Comparison of multi-lineage cells from human adipose tissue and bone marrow. *Cells, Tissues, Organs*, 174(3):101–109, 2003. PMID: 12835573.

[33] SBML developers. Sbml composition workshop 2007. `http://sbml.info/Events/Other_Events/SBML_Composition_Workshop_2007`, 2007.

[34] G. S. Drysdale and G. H. Fleet. Acetic acid bacteria in winemaking: A review. *Am. J. Enol. Vitic.*, 39(2):143–154, June 1988.

[35] G. H. Fleet, S. Lafon-Lafourcade, and P. Ribéreau-Gayon. Evolution of yeasts and lactic acid bacteria during fermentation and storage of bordeaux wines. *Applied and Environmental Microbiology*, 48(5):1034–1038, November 1984. PMID: 16346661 PMCID: 241671.

[36] David V Foster, Jacob G Foster, Sui Huang, and Stuart A Kauffman. A model of sequential branching in hierarchical cell fate determination. *Journal of Theoretical Biology*, 260(4):589–597, October 2009. PMID: 19615382.

[37] Valerie Frezier and Denis Dubourdieu. Ecology of yeast strain saccharomyces cerevisiae during spontaneous fermentation in a bordeaux winery. *Am. J. Enol. Vitic.*, 43(4):375–380, December 1992.

[38] C W. Gear. The simultaneous numerical solution of differential-algebraic equations. *IEEE Trans. Circuit Theory*, 18(1):89–95, 1971.

[39] J. Gebert, N. Radde, and G.-W. Weber. Modeling gene regulatory networks with piecewise linear differential equations. *European Journal of Operational Research*, 181(3):1148–1165, 2007.

[40] A Goldbeter. A minimal cascade model for the mitotic oscillator involving cyclin and cdc2 kinase. *Proceedings of the National Academy of Sciences of the United States of America*, 88(20):9107–9111, October 1991.

[41] Y Gong, R B Slee, N Fukai, G Rawadi, S Roman-Roman, A M Reginato, H Wang, T Cundy, F H Glorieux, D Lev, M Zacharin, K Oexle, J Marcelino, W Suwairi, S Heeger, G Sabatakos, S Apte, W N Adkins, J Allgrove, M Arslan-Kirchner, J A Batch, P Beighton, G C Black, R G Boles, L M Boon, C Borrone, H G Brunner, G F Carle, B Dallapiccola, A De Paepe, B Floege, M L Halfhide, B Hall, R C Hennekam, T Hirose, A Jans, H JÃ$\frac{1}{4}$ppner, C A Kim, K Keppler-Noreuil, A Kohlschuetter, D LaCombe, M Lambert, E Lemyre, T Letteboer, L Peltonen, R S Ramesar, M Romanengo, H Somer, E Steichen-Gersdorf, B Steinmann, B Sullivan, A Superti-Furga, W Swoboda, M J van den Boogaard, W Van Hul, M Vikkula, M Votruba, B Zabel, T Garcia, R Baron, B R Olsen, and M L Warman. LDL receptor-related protein 5 (LRP5) affects bone accrual and eye development. *Cell*, 107(4):513–523, November 2001. PMID: 11719191.

[42] A N Gorban and O Radulescu. Dynamical robustness of biological networks with hierarchical distribution of time scales. *IET Systems Biology*, 1(4):238–246, July 2007. PMID: 17708431.

[43] Sylvain Goutelle, Michel Maurin, Florent Rougier, Xavier Barbaut, Laurent Bourguignon, Michel Ducher, and Pascal Maire. The hill equation: a review of its capabilities in pharmacological modelling. *Fundamental & Clinical Pharmacology*, 22(6):633–648, December 2008. PMID: 19049668.

[44] Iva Greenwald. Lin-12/notch signaling: lessons from worms and flies. *Genes & Development*, 12(12):1751 –1762, June 1998.

[45] A C Guyton. Brief reviews: A systems analysis approach to understanding Long-Range arterial blood pressure control and hypertension. http://circres.ahajournals.org/cgi/content/citation/35/2/159, August 1974.

[46] A C Guyton, T G Coleman, and H J Granger. Circulation: overall regulation. *Annual Review of Physiology*, 34:13–46, 1972. PMID: 4334846.

[47] N. Halbwachs. A synchronous language at work: the story of lustre. In *Proceedings of the 2nd ACM/IEEE International Conference on Formal Methods and Models for Co-Design*, pages 3–11. IEEE Computer Society, 2005.

[48] Jing-Dong J. Han, Nicolas Bertin, Tong Hao, Debra S. Goldberg, Gabriel F. Berriz, Lan V. Zhang, Denis Dupuy, Albertha J. M. Walhout, Michael E. Cusick, Frederick P. Roth, and Marc Vidal. Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature*, 430(6995):88–93, July 2004.

[49] L H Hartwell, J J Hopfield, S Leibler, and A W Murray. From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–52, December 1999. PMID: 10591225.

[50] Thomas A. Henzinger. The theory of hybrid automata. Technical Report UCB/ERL M96/28, EECS Department, University of California, Berkeley, 1996.

[51] A.V. Hill. The possible effects of the aggregation of the molecules of hemoglobin on its dissociation curves. *J.Physiol*, 40:iv–vii, January 1910.

[52] Arend Hintze and Christoph Adami. Evolution of complex modular biological networks. *PLoS Comput Biol*, 4(2):e23, February 2008.

[53] Christian Hodar, Rodrigo Assar, Marcela Colombres, Andres Aravena, Leonardo Pavez, Mauricio Gonzalez, Servet Martinez, Nibaldo Inestrosa, and Alejandro Maass. Genome-wide identification of new wnt/beta-catenin target genes in the human genome using CART method. *BMC Genomics*, 11(1):348, 2010.

[54] Luke H. Hoeppner, Frank J. Secreto, and Jennifer J. Westendorf. Wnt signaling as a therapeutic target for bone diseases. *Expert opinion on therapeutic targets*, 13(4):485–496, April 2009. PMID: 19335070 PMCID: 3023986.

[55] Sui Huang, Yan-Ping Guo, Gillian May, and Tariq Enver. Bifurcation dynamics in lineage-commitment in bipotent progenitor cells. *Developmental Biology*, 305(2):695–713, May 2007.

[56] Michael Hucka, Michael Hucka, Frank Bergmann, Stefan Hoops, Sarah Keating, Sven Sahle, and Darren Wilkinson. The systems biology markup language (SBML): language specification for level 3 version 1 core (Release 1 candidate). *Nature Precedings*, 2010.

[57] Alexander Hunziker, Mogens Jensen, and Sandeep Krishna. Stress-specific response of the p53-Mdm2 feedback loop. *BMC Systems Biology*, 4(1):94, 2010.

[58] Yehea I Ismail, Eby G Friedman, and Jose L Neves. Figures of merit to characterize the importance of On-Chip inductance. *PROCEEDINGS OF THE IEEE/ACM DESIGN AUTOMATION CONFERENCE*, 7:560–565, 1998.

[59] Paul S. Issack, David L. Helfet, and Joseph M. Lane. Role of wnt signaling in bone remodeling and repair. *HSS Journal*, 4(1):66–70, February 2008. PMID: 18751865 PMCID: 2504275.

[60] Karl Henrik Johansson. Hybrid control systems. Technical Report EOLSS,6.43.28, Dept. of Signals, Sensors & Systems, Royal Institute of Technology, 1044., 2003.

[61] Cliff Joslyn and Luis M Rocha. Towards semiotic Agent-Based models of Socio-Technical organizations. *IN: PROC. AI AND SIMULATION 2000, ED. HS SARJOUGHIAN ET AL*, pages 70—79, 2000.

[62] D Kim, O Rath, W Kolch, and K-H Cho. A hidden oncogenic positive feedback loop caused by crosstalk between wnt and ERK pathways. *Oncogene*, 26(31):4571–4579, January 2007.

[63] Duk Jae Kim, Jung-Min Koh, Oksun Lee, Na Jung Kim, Young-Sun Lee, Yang Soon Kim, Joong-Yeol Park, Ki-Up Lee, and Ghi Su Kim. Homocysteine enhances apoptosis in human bone marrow stromal cells. *Bone*, 39(3):582–590, September 2006. PMID: 16644300.

[64] Hiroaki Kitano. Computational systems biology. *Nature*, 420(6912):206–210, November 2002.

[65] Hiroaki Kitano. Biological robustness. *Nature Reviews. Genetics*, 5(11):826–837, November 2004. PMID: 15520792.

[66] Hiroaki Kitano, Kanae Oda, Tomomi Kimura, Yukiko Matsuoka, Marie Csete, John Doyle, and Masaaki Muramatsu. Metabolic syndrome and robustness tradeoffs. *Diabetes*, 53 Suppl 3:S6–S15, December 2004. PMID: 15561923.

[67] Ernesto Kofman. A Second-Order approximation for DEVS simulation of continuous systems. *SIMULATION*, 78(2):76–89, February 2002.

[68] Ernesto Kofman. Quantization-Based simulation of differential algebraic equation systems. *SIMULATION*, 79(7):363–376, July 2003.

[69] Venkatesh Krishnan, Henry U Bryant, and Ormond A Macdougald. Regulation of bone mass by wnt signaling. *The Journal of Clinical Investigation*, 116(5):1202–1209, May 2006. PMID: 16670761.

[70] Takuo Kubota, Toshimi Michigami, and Keiichi Ozono. Wnt signaling in bone metabolism. *Journal of Bone and Mineral Metabolism*, 27(3):265–271, 2009.

[71] Jitender Kumar, Maria Swanberg, Fiona McGuigan, Mattias Callreus, Paul Gerdhem, and Kristina Akesson. LRP4 association to bone properties and fracture and interaction with genes in the wnt- and BMP signaling pathways. *Bone*, In Press, Accepted Manuscript, 2011.

[72] Irving Langmuir. THE CONSTITUTION AND FUNDAMENTAL PROPERTIES OF SOLIDS AND LIQUIDS. PART i. SOLIDS. *Journal of the American Chemical Society*, 38(11):2221–2295, November 1916.

[73] Ben D. MacArthur, Colin P. Please, and Richard O. C. Oreffo. Stochasticity and the molecular mechanisms of induced pluripotency. *PLoS ONE*, 3(8):e3086, 2008.

[74] Radhakrishnan Mahadevan, Jeremy S Edwards, and Francis J Doyle. Dynamic flux balance analysis of diauxic growth in escherichia coli. *Biophysical Journal*, 83(3):1331–1340, September 2002. PMID: 12202358 PMCID: 1302231.

[75] S. Malherbe, V. Fromion, N. Hilgert, and J.-M. Sablayrolles. Modeling the effects of assimilable nitrogen and temperature on fermentation kinetics in enological conditions. *Biotechnology and Bioengineering*, 86(3):261–272, 2004.

[76] Stavros C. Manolagas. Birth and death of bone cells: Basic regulatory mechanisms and implications for the pathogenesis and treatment of osteoporosis. *Endocrine Reviews*, 21(2):115 –137, April 2000.

[77] Lucia Marucci, Stefania Santini, Mario Bernardo, and Diego Bernardo. Derivation, identification and validation of a computational model of a novel synthetic regulatory network in yeast. *Journal of Mathematical Biology*, 62(5):685–706, 2010.

[78] Carsten Maus, Mathias John, Mathias Rohl, and Adelinde Uhrmacher. Hierarchical modeling for computational biology. In *Formal Methods for Computational Systems Biology*, pages 81–124. SpringerLink, 2008.

[79] A. Mendes-Ferreira, M. del Olmo, J. Garcia-Martinez, E. Jimenez-Marti, A. Mendes-Faia, J. E. Perez-Ortin, and C. Leao. Transcriptional response of saccharomyces cerevisiae to different nitrogen concentrations during alcoholic fermentation. *Appl. Environ. Microbiol.*, 73(9):3049–3060, May 2007.

[80] Monod. La technique de culture continue; thÃ©orie et applications. *Ann Ist Pasteur Lille*, 79:390–410, 1950.

[81] Douglas C. Montgomery. *Design and Analysis of Experiments, 5th Edition.* Wiley, 5 edition, June 2000.

[82] James D. Murray. *Mathematical Biology: I. An Introduction (Interdisciplinary Applied Mathematics).* Springer, 3rd edition, December 2007.

[83] Vincent Noel, Sergei Vakulenko, and Ovidiu Radulescu. Piecewise smooth hybrid systems as models for networks in molecular biology. http://www.jobim2010.fr/sites/default/files/actes_jobim_2010.pdf.

[84] Canan Nurgel, Huseyin Erten, Ahmet Canbas, Turgut Cabaroglu, and Serkan Selli. Yeast flora during the fermentation of wines made from vitis viniferaL. cv. emir and kalecik karasi grown in anatolia. *World Journal of Microbiology and Biotechnology*, 21(6):1187–1194, October 2005.

[85] Francisco Pizarro, Cristian Varela, Cecilia Martabit, Claudio Bruno, J. Ricardo Pérez-Correa, and Eduardo Agosin. Coupling kinetic expressions and metabolic networks for predicting wine fermentations. *Biotechnology and Bioengineering*, 98(5):986–998, 2007.

[86] Gérald Point. AltaRica : Contribution à l'unification des méthodes formelles et de la sûreté de fonctionnement. http://tel.archives-ouvertes.fr/tel-00353284/fr/, 2000.

[87] Ovidiu Radulescu, Alexander Gorban, Andrei Zinovyev, and Alain Lilienbaum. Robust simplifications of multiscale biochemical networks. *BMC Systems Biology*, 2(1):86, 2008.

[88] Ovidiu Radulescu, Alexander N. Gorban, Sergei Vakulenko, and Andrei Zinovyev. Hierarchies and modules in complex biological systems. http://www.cabdyn.ox.ac.uk/complexity_PDFs/ECCS06/ Conference_Proceedings/PDF/p114.pdf, 2006.

[89] Alexander W. Rives and Timothy Galitski. Modular organization of cellular networks. *Proceedings of the National Academy of Sciences of the United States of America*, 100(3):1128 –1133, February 2003.

[90] MarÃa Eugenia Rodríguez, Christian A Lopes, Raúl J Barbagelata, Nora B Barda, and Adriana C Caballero. Influence of candida pulcherrima patagonian strain on alcoholic fermentation behaviour and wine aroma. *International Journal of Food Microbiology*, 138(1-2):19–25, March 2010. PMID: 20116878.

[91] Kiichi Sagawa. Critique of a large-scale organ system model: Guytonian cardiovascular model. *Annals of Biomedical Engineering*, 3(4):386–400, 1975.

[92] Javier Sainz, Francisco Pizarro, J. Ricardo Pérez-Correa, and Eduardo Agosin. Modeling of yeast metabolism and process dynamics in batch fermentation. *Biotechnology and Bioengineering*, 81(7):818–828, 2003.

[93] Pilar SantamaríÂa, Patrocinio Garijo, Rosa López, Carmen Tenorio, and Ana Rosa Gutiérrez. Analysis of yeast population during spontaneous alcoholic fermentation: Effect of the age of the cellar and the practice of inoculation. *International Journal of Food Microbiology*, 103(1):49–56, August 2005.

[94] Gustavo J.E. Scaglia, Pablo M. Aballay, Carmen A. Mengual, Martha D. Vallejo, and Oscar A. Ortiz. Improved phenomenological model for an isothermal winemaking fermentation. *Food Control*, 20(10):887–895, 2009.

[95] E. Schachinger and B. Schnizer. General solution of the two-dimensional heat equation for two concentric domains of different material. *Warme- und Stoffubertragung*, 14(1):7–13, 1980.

[96] D. Schittler, J. Hasenauer, F. Allgöwer, and S; Waldherr. Cell differentiation modeled via a coupled two-switch regulatory networks. *Chaos*, 20(4), December 2010.

[97] R Schmidmaier, P Baumann, and G Meinhardt. Cell-cell contact mediated signalling - no fear of contact. *Experimental Oncology*, 28(1):12–15, March 2006. PMID: 16614701.

[98] James F. Selgrade. Dynamical behavior of a competitive model with genetic variation. *Applied Mathematics Letters*, 2(1):49–52, 1989.

[99] Mohammad Shahnazari, Wei Yao, Maripat Corr, and Nancy E. Lane. Targeting the wnt signaling pathway to augment bone formation. *Current Osteoporosis Reports*, 6(4):142–148, 2008.

[100] Robert Shorten, Fabian Wirth, Oliver Mason, Kai Wulff, and Christopher King. Stability criteria for switched and hybrid systems. *SIAM REVIEW*, 49:545–592, 2005.

[101] Hayssam Soueidan, David James Sherman, and Macha Nikolski. BioRica: A multi model description and simulation system. In *Foundations of Systems Biology and Engineering*, pages 279–287, Allemagne, 2007.

[102] Karin Stenderup, Jeannette Justesen, Christian Clausen, and Moustapha Kassem. Aging is associated with decreased maximal life span and accelerated senescence of bone marrow stromal cells. *Bone*, 33(6):919–926, December 2003. PMID: 14678851.

[103] May Kung Sutherland, James C Geoghegan, Changpu Yu, Eileen Turcott, John E Skonier, David G Winkler, and John A Latham. Sclerostin promotes the apoptosis of human osteoblastic cells: a novel regulation of bone formation. *Bone*, 35(4):828–835, October 2004. PMID: 15454089.

[104] L Tavernini. Differential automata and their discrete simulators. *Nonlinear Analysis*, 11(6):665–683, 1987.

[105] Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern Recognition, Second Edition*. Academic Press, 2 edition, April 2003.

[106] S. Randall Thomas, Pierre Baconnier, Julie Fontecave, Jean-Pierre Francoise, Francois Guillaud, Patrick Hannaert, Alfredo Hernández, Virginie Le Rolle, Pierre Maziére, Fariza Tahi, and Ronald J White. SAPHIR: a physiome core

model of body fluid homeostasis and blood pressure regulation. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1878):3175 –3197, 2008.

[107] M.E. Toro and F. Vazquez. Fermentation behaviour of controlled mixed and sequential cultures of candida cantarellii and saccharomyces cerevisiae wine yeasts. *World Journal of Microbiology and Biotechnology*, 18(4):347–354, January 2002.

[108] J J Tyson. Modeling the cell division cycle: cdc2 and cyclin interactions. *Proceedings of the National Academy of Sciences of the United States of America*, 88(16):7328 –7332, 1991.

[109] Adelinde M. Uhrmacher, Daniela Degenring, Jens Lemcke, and Mario Krahmer. Towards reusing model components in systems biology. In *Computational Methods in Systems Biology*, pages 192–206. SpringerLink, 2005.

[110] A J Wagers, J L Christensen, and I L Weissman. Cell fate determination from stem cells. *Gene Therapy*, 9(10):606–612, May 2002. PMID: 12032706.

[111] Wei Wei, Xueqian Wang, Marie Yang, Leslie C Smith, Paul C Dechow, Junichiro Sonoda, Ronald M Evans, and Yihong Wan. PGC1beta mediates PPARgamma activation of osteoclastogenesis and rosiglitazone-induced bone loss. *Cell Metabolism*, 11(6):503–516, June 2010. PMID: 20519122.

[112] Darren James Wilkinson. Stochastic transition systems for continuous state spaces and non-determinism. http://cat.inist.fr/?aModele=afficheN&cpsidt=16895179.

[113] Nicolás Wolovick and Sven Johr. A characterization of meaningful schedulers for continuous-time markov decision processes. In *FORMATS 2006. LNCS*, volume 4202, pages 352–367, 2006.

[114] Stefan Wuchty, Erszébet Ravasz, and Albert-László BarabÃ¡si. The architecture of biological networks. *COMPLEX SYSTEMS IN BIOMEDICINE*, 2003.

[115] Bernard P. Zeigler, Herbert Praehofer, and Tag Gon Kim. *Theory of Modeling and Simulation, Second Edition.* Academic Press, 2 edition, January 2000.

[116] Katharina Zott, Cecile Miot-Sertier, Olivier Claisse, Aline Lonvaud-Funel, and Isabelle Masneuf-Pomarede. Dynamics and diversity of non-Saccharomyces yeasts during the early stages in winemaking. *International Journal of Food Microbiology*, 125(2):197–203, July 2008. PMID: 18495281.