UNIVERSITÀ DEGLI STUDI DI TRENTO

Facoltà di Scienze Matematiche, Fisiche e Naturali

Corso di Laurea in Informatica Specialistica

TESI

MODELLING MOLECULAR SYSTEMS WITH DISCRETE CONCENTRATION LEVELS IN THE CONTEXT OF PROCESS ALGEBRA PEPA: STOCHASTIC AND DETERMINISTIC INTERPRETATIONS

RELATORE: Prof. Corrado Priami

LAUREANDO: Marco Stenico

CORRELATORE: Dott.ssa Jane Hillston

ANNO ACCADEMICO 2005/2006

“One day he remarked, without lifting his head, ’In the interior you will no doubt meet Mr.Kurtz’”

from Heart of Darkness by Joseph Conrad

Abstract

Process algebras are formalisms widely used to represent concurrent systems such as biological systems (Regev et al., 2001, Priami et al., 2001). An algebraic specification of a system is composed by processes that interact and communicate between them or are synchronized over a set of actions. Algebraic laws describe how processes are defined and how they can be modified.

In this thesis we present a novel approach for modelling molecular systems (Calder et al., 2004) based on discrete levels of concentration. Here each molecule species is represented by a process and each process has an index that denotes the current level of the corresponding molecule. Actions model reactions as usual and rates are computed following some rules. Our analysis is performed in the context of PEPA process algebra (Hillston, 1995).

An algebraic model can have several mathematical interpretations. We consider Markov chains, ordinary differential equations and reactions for stochastic simulations (Gillespie, 1976). It is well known that, when the number of molecules is *sufficiently* large, stochastic simulations converge to a deterministic limit distribution (Kurtz, 1970). Instead it is not clear the relation between differential equations and Markov chains where states represent concentration levels instead of individual molecules. The scope of this thesis is to throw some light on these relationships.

In the first part we illustrate this modelling style on the real problem of modelling circadian clocks (Goldbeter, 2002) and we present some experiments useful to understand the relationships between stochastic and deterministic simulations. Our approach yields results similar to the literature (Gonze et al., 2002a,b).

In the second part we analyse the relationships between Markov chains with discrete levels

and differential equations. We prove that if a model has a particular structure, then the underlying Markov chain converges to the deterministic interpretation as the number of levels increases.

Finally we use our results to prove that the representation of the ERK signalling pathway (Cho et al., 2003) with discrete levels of concentrations yields results similar to the deterministic model when the number of levels is *sufficiently* large.

Acknowledgements

I carried out this research project during my visit at the University of Edinburgh as Socrates/Erasmus student. I am very grateful to Dr.Jane Hillston (University of Edinburgh), who advised me and helped me in various aspects of my work. I thank her for her patience and encouragement and for reviewing this thesis. I thank also Prof.Corrado Priami (University of Trento) who introduce me to Jane. Last but not least, I thank all the people and animals who were there when I needed them most.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Motivations

Over the last decades high throughput technologies have generated a large amount of information on the cell. Now basic interactions among genes, proteins, RNA and other molecules are well known and several metabolic and signal pathways have been indentified. At the same time, however, the complexity of biological systems has been increased and it is almost impossible to understand their behaviour when considered as a whole. Therefore mathematical modelling and computer simulations are needed to unravel the dynamics of biological processes. Systems biology is the branch of bioinformatics which studies these techniques. From experimental data systems biologists propose hypotheses to explain a system's behaviour. These hypotheses are used to mathematically model the system. Models are used to predict the behaviour of the system and to formulate new experimentally verifiable hypotheses iteratively.

Process algebras are formalisms widely used to represent concurrent systems. In a process algebra several independent subsystems, called processes, interact and communicate between them or are synchronized over a set of actions. Algebraic laws describe how processes are defined and how they can be modified. For example the expression $\alpha.P$ models a process which can undertake action α and then becomes P. A labelled transition system is a graph representing all possible

states and transitions. This graph can be used to verify some properties of the system. Sometimes actions are associated with rates; in this case the notion of "time" is embedded inside models and quantitative analysis can be performed using an underlying mathematical interpretation.

Recently process algebras have been used to model biological systems (Regev et al., 2001, Priami et al., 2001). Process algebras have several advantages in modelling biological systems with respect to other traditional formalisms, such as differential equations. Modelling is focused on the high level description of system entities and their interactions rather than directly on the underlying mathematical interpretations. Different algebraic formulations of the same system can be compared, e.g. through bisimulation (Calder et al., 2004). Moreover process algebras are compositional and allow abstraction to hide complexity or incomplete knowledge.

While in literature (Regev et al., 2001) processes model individual molecules, Calder et al. (2004) proposed a novel and alternative style of modelling where processes represent discrete levels of concentration. The scope is to reduce the size of the state space and to deal with incomplete information. Each molecule is represented by a process and each process has an index that represents the current level of the corresponding molecule. Actions model reactions as usual and rates are computed following some rules.

It is challenging to discover the relationships between the possible mathematical interpretations of an algebraic model based on discrete levels of concentration. In Figure 1.1 we compare some possible mathematical interpretations.

A Markov chain $X_N(t)$ is extracted assigning a state to each node of the labelled transition system and defining transitions for each arc (Hillston, 1995). Index N is the greatest possible level for each molecule. A state is represented by a vector $X_N(t) = (x_1(t), \ldots, x_n(t)) \in \mathbb{N}^n$ where $x_i(t)$ is the level of the i-th molecule species at time t. M is the maximal concentration and $M/NX_N(t)$ is the discrete concentration vector. Transition rates depend on the current state; however when some product levels of a reaction are N, transition rates corresponding to the reaction are zero.

An ODE system is derived from the syntax of the model building an activity graph (Calder et al.) that represents increasing and descreasing of molecular concentrations in reactions. When

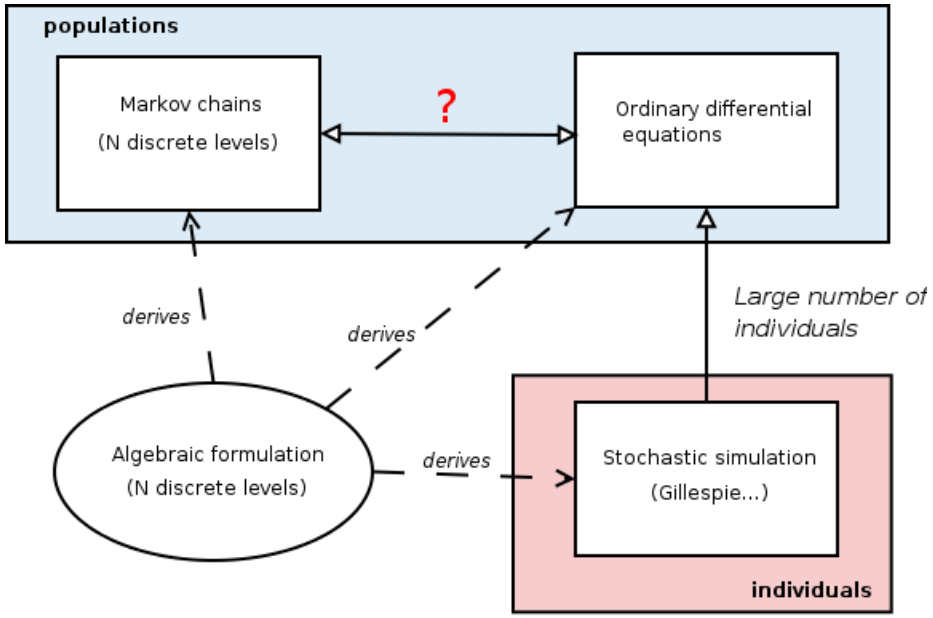


Figure 1.1: **Relationships between different mathematical interpretations of the same algebraic specification.** Markov chains and ODEs treat molecules in populations or concentrations. Instead stochastic simulations consider molecules individually. Markov chains and stochastic simulations have discrete state space on the contrary of ODEs. All the models are continous time. Stochastic simulations *converge* to a limit deterministic distribution when the number of individuals is *sufficiently large*. At the moment it is not clear which relations exist between Markov chains and ODEs.

we consider only two levels of concentration (e.g. high and low), the algebraic specification contains enough information for extracting differential equations of this form,

$$\frac{dX(t)}{dt} = F(X(t))$$

Here, the state of the system is given by $X(t) = (x_1(t), \ldots, x_n(t))$ where $x_i(t)$ denotes the concentration of the i-th molecule at time t whereas F is a function that describes the dynamical behaviour of the system following the Mass Action Law.

From the activity graph a stochastic simulation is derived, too. A set of reaction equations and the corresponding occurence probabilities define a model that is used as input for Gillespie's

algorithm (Gillespie, 1976).

In stochastic simulations each molecule is treated individually, whereas in ODEs and Markov chains with discrete levels molecules are considered in concentrations. It is well known that, when the number of molecules is sufficiently large, stochastic simulations converge to a deterministic limit distribution. Instead it is not clear the relation between differential equations and Markov chains. There is some evidence that increasing the number of levels N the avarage behaviour of Markov chains seems to converge to the solution of the differential equations (Calder et al., 2005), however it is not well understood how this happens and if it is always true. The scope of this thesis is to throw some light on these relationships.

1.2 Methods and Tools

Our analysis is performed in the context of PEPA. PEPA is a stochastic process algebra invented by Jane Hillston (Hillston, 1995) for modelling computer and communication systems. Systems are formed by several components which can perform activities. Each activity has a duration and an action type. For example the expression $(\alpha, r).P$ models a system which can undertake action α with rate r and becomes P.

Several tools were used during the development of this work. Here, I list some of them. Readers intertested in details can look at the referenced documents.

PEPA Workbench (Gilmore, 2001) is a Java application for PEPA models. It can parse models, extract Markov chains in different formats, find steady state solutions and other features.

PRISM (Parker et al., 2006) is a probabilistic model checker written in Java for modelling and analysing probabilistic systems. It was developed at the University of Birmingham. It supports continuous time Markov chain models and implements CSL model checking (Aziz et al., 1996), a logic to express properties of steady state and transient behaviour of Markov processes.

Dizzy (Ramsey, 2006) is a chemical kinetics simulation software package written in the Java programming language. It allows to define models as systems of chemical reactions. It performs several kinds of stochastic and deterministic simulations (e.g. Gillespie).

GNU Octave/Matlab (Eaton, 2005) is a high-level language for solving linear and nonlinear problems numerically using a language compatible with Matlab. It was used to solve systems of ordinary differential equations using Runge-Kutta5 method.

1.3 Organization

In the first part of this thesis we present some background material useful to understand the rest. We give an informal description of PEPA semantics, we describe basic notions about chemical reactions, nonlinear dynamic systems, Markov chains and simulation techniques. The reader who already knows these topics can overlook this chapter.

In the second part we describe formally the modelling style based on discrete concentration levels. We show how to extract differential equations and stochastic simulations from models automatically. An activity graph is a graph that represents molecules and their interactions. An activity graph is built up performing some syntax analysis on a PEPA model. Stochastic simulations and differential equations are derived from the activity graph. We illustrate this modelling style on the real problem of modelling circadian clocks (Goldbeter, 2002). We show some drawbacks of our approach and some *ad hoc* solutions. Finally we present some experiments useful to understand the relationships between stochastic and deterministic simulations. Our approach yields results similar to the literature (Gonze et al., 2002a,b).

In the third part we analyse the relationships between Markov chains with discrete levels and differential equations. Firstly, we propose some simple examples, we solve the corresponding Markovian and determinstc models analytically and we prove that for these particular models the average Markovian behaviour converges to the solution of the deterministic systems as the number of levels increases. Secondly, we try to generalize previous results.

A density dependent family of Markov chains X_v is a sequence $\{X_v\}$ of Markov processes defined by a parameter v. The states of the Markov chains are normalized with respect to v and the state spaces of this sort of densities are considered instead. The transition rates depend on the densities, hence the name density dependent Markov chains. Kurtz's theorem (Kurtz, 1970) says that as parameter v grows *arbitrarily large*, the sequence of stochastic processes $v^{-1}X_v(t)$ *converges* to a deterministic process $X(t)$ which is solution of a system of ordinary differential equations. While systems with finite v are discrete processes, the limiting system is continuous.

The Markovian representation of a PEPA model with discrete levels is a parametrized Markov chain $X_N(t)$, where N is the number of levels. However, the sequence $\{X_N(t)\}$ is not always density dependent. In fact when some product levels of a reaction are N, transition rates corresponding to the reaction are zero. We formulate a sufficient and necessary structural condition on the state space of $X_N(t)$ in such a way that the sequence $\{X_N(t)\}$ is a sequence of dentity dependent Markov chains and Kurtz's theorem can be applied. We propose an effective way to verify if a model satifies this condition. Roughly speaking we use a generalization of the notion of activity graph (essentially a *Petri Net*) to represent the state spaces of all the Markov chains. Then if the activity graph presents some structural properties (e.g. boundedness), the corresponding sequence is density dependent.

We use our results to prove that the representation of the ERK signalling pathway (Cho et al., 2003) with discrete levels of concentrations yields results similar to the deterministic model when the number of levels is *sufficiently* large.

Chapter 2

Introduzione

2.1 Motivazioni

Negli ultimi decenni l'insorgere di nuove tecnologie ad alte prestazioni ha permesso di disporre di una grande quantità di informazioni sulla cellula. Ora conosciamo molto bene le interazioni fondamentali tra i geni, le proteine, l'RNA e le altre molecole ed abbiamo individuato molti dei principali meccanismi biologici. Al tempo stesso, però, la complessità dei sistemi biologici è cresciuta ed è impensabile di poter capire il loro comportamento quando presi in considerazione nel loro insieme. Per questo la modellazione matematica e le simulazioni al computer sono necessarie per scoprire le dinamiche dei processi biologici. La biologia dei sistemi è un settore della bioinformatica che studia queste tecniche. Dai dati sperimentali i biologi propongono delle ipotesi per spiegare il comportamento di un sistema. Queste ipotesi sono utilizzate per modellare matematicamente il sistema. I modelli sono utilizzati per predirre il comportamento del sistema e per formulare nuove ipotesi da verificare empiricamente in modo iterativo.

Le algebre di processo sono formalismi molto utilizzati per rappresentare dei sistemi concorrenti. In un algebra di processo alcuni sotto-sistemi indipendenti, chiamati processi, interagiscono e comunicano tra loro o sono sincronizzati su un insieme di azioni. Delle leggi algebriche descrivono come i processi sono definiti e come possono essere modificati. Per esempio l'espressione $\alpha.P$

modella un processo che può portare a termine un'azione α e diventare poi P. Un *labelled transition system* è un grafo che rappresenta tutti i possibili stati e transizioni di un sistema. Questo grafo può essere usato per verificare alcune proprietà del sistema. Talvolta le azioni sono associate con delle velocità; in questo caso la nozione di "tempo" è intrinseca al modello e un'analisi quantitativa può essere effettuata utilizzando qualche sottostante interpretazione matematica.

Di recente le algebre di processo sono state utilizzate per modellare sistemi biologici (Regev et al., 2001, Priami et al., 2001). Le algebre di processo hanno diversi vantaggi nella modellazione di sistemi biologici rispetto agli altri tradizionali formalismi, come le equazioni differenziali. La modellazione infatti si focalizza su la descrizione ad alto livello delle entità del sistema e delle loro interazione anzichè direttamente sulle interpretazioni matematiche sottostanti. Differenti formulazioni algebriche dello stesso sistema possono essere confrontate, ad esempio con la bisimulazione (Calder et al., 2004). Inoltre le algebre di processo sono composizionali e permettono l'astrazione per nascondere la complessità o le conoscenze incomplete.

Mentre in letteratura (Regev et al., 2001) i processi modellano le singole molecole, Calder et al. (2004) hanno proposto un nuovo ed alternativo stile di modellazione dove i processi rappresentano livelli discreti di concentrazione. Lo scopo è ridurre la dimensione dello spazio degli stati e gestire il caso di informazione incompleta. Ogni molecola è rappresentata da un processo e ogni processo ha un indice che rappresenta il livello corrente della molecola corrispondente. Le azioni modellano le reazioni come al solito e le velocità sono calcolate seguendo certe regole.

È interessante scoprire le relazioni tra le possibili interpretazioni matematiche di un modello algebrico basato su livelli discreti di concentrazione. In Figura 1.1 confrontiamo alcune possibili interpretazioni matematiche.

Una catena di Markov $X_N(t)$ è estratta assegnando uno stato ad ogni nodo dell'*labelled transition system* e definendo delle transizioni per ogni arco (Hillston, 1995). L'indice N è il più grande livello possibile per ogni molecola. Uno stato è rappresentato da un vettore $X_N(t) = (x_1(t), \ldots, x_n(t)) \in \mathbb{N}^n$ dove $x_i(t)$ è il livello dell'i-esima molecola al tempo t. M è la concentrazione massima e $M/NX_N(t)$ è il vettore delle concentrazioni discrete. Le velocità di transizione dipendono dallo stato corrente,

però quando i livelli di alcuni prodotti di una reazione sono N, le velocità di transizione corrispondenti alla reazione sono zero.

Un sistema di ODE è derivato dalla sintassi del modello costruendo un grafo delle attività (Calder et al.) che rappresenta gli incrementi e i decrementi delle concentrazioni molecolari nelle reazioni. Quando consideriamo solamente due livelli di concentrazione (cioè alto e basso), la specificazione algebrica contiene abbastanza informazione per estrarre delle equazioni differenziali di questa forma,

$$\frac{dX(t)}{dt} = F(X(t))$$

Qui, lo statto del sistema è dato da $X(t) = (x_1(t), \ldots, x_n(t))$ dove $x_i(t)$ denota la concentrazione della i-esima molecola al tempo t mentre F è una funzione che descrive il comportamento dinamico del sistema seguendo la Legge di Massa.

Dal grafo delle attività è derivata anche una simulazione stocastica. Un insieme di reazioni e le corrispondenti probabilità di occorrenza definiscono un modello che è poi usato come input per l'algoritmo di Gillespie (Gillespie, 1976).

Nelle simulazioni stocastiche ogni molecola è trattata individualmente, mentre nelle ODE e nelle catene di Markov con livelli discreti le molecole sono considerate in concentrazioni. È ben noto che, quando il numero di molecole è sufficientemente grande, le simulazioni stocastiche convergono ad una distribuzione limite deterministica. Invece non è chiara la relazione tra le equazioni differenziali e le catene di Markov. In alcuni casi incrementando il numero di livelli N il comportamento medio delle catene di Markov sembra convergere alla soluzione delle equazioni differenziali (Calder et al., 2005), però non si capisce bene come questo accada e se sia sempre vero. Lo scopo di questa tesi è di gettare un po' di luce su queste relazioni.

2.2 Metodologie e strumenti

La nostra analisi è eseguita nel contesto di PEPA. PEPA è un'algebra di processo stocastica inventata da Jane Hillston (Hillston, 1995) per modellare sistemi di computer. I sistemi sono formati da diverse componenti che possono eseguire delle attività. Ogni attività ha una durata e un tipo di azione. Per esempio l'espressione $(\alpha, r).P$ modella un sistema che esegue un'azione α con velocità r per poi diventare P.

Durante lo sviluppo di questo lavoro abbiamo utilizzato diversi strumenti. Ne elenchiamo di seguito alcuni. I lettori interessati ai dettagli possono andare a vedere la documentazione citata.

PEPA Workbench (Gilmore, 2001) è un'applicazione Java per modelli PEPA. Parserizza i modelli, estrae le catene di Markov in diversi formati, trova le soluzioni stazionare e così dicendo.

PRISM (Parker et al., 2006) è un *model checker* probabilistico scritto in Java per modellare ed analizzare sistemi probabilistici. È stato sviluppato presso l'Università di Birmingham. Supporta modelli di catene di Markov a tempo continuo ed implementa un *model checking* per CSL (Aziz et al., 1996), una logica che permette di esprimere proprietà del comportamento stazionario e transitorio dei processi di Markov.

Dizzy (Ramsey, 2006) è un pacchetto software per la simulazione di reazioni chimiche scritto nel linguaggio di programmazione Java. Permette di definire modelli come sistemi di reazioni chimiche. Esegue diversi tipi di simulazioni stocastiche e deterministiche (ad esempio Gillespie).

GNU Octave/Matlab (Eaton, 2005) è un linguaggio ad alto livello per risolvere problemi lineari e non lineari numericamente usando un linguaggio compatibile con Matlav. È stato utilizzato per risolvere sistemi di equazioni differenziali ordinarie usando il metodo di Runge-Kutta 5.

2.3 Organizzazione

Nella prima parte di questa tesi presentiamo del materiale introduttivo utile per capire il resto. Diamo una descrizione informale della semantica di PEPA, descriviamo alcune nozioni basilari di chimica, di teoria dei sistemi dinamici non lineari, di catene di Markov e tecniche di simulazione. Il lettore che già conosce questi argomenti può saltare questo capitolo.

Nella seconda parte desciviamo formalmente lo stile di modellazione basato su livelli discreti di concentrazione. Mostriamo come estrarre le equazioni differenziali e le simulazioni stocastiche dai modelli in modo automatico. Un grafo delle attività è un grafo che rappresenta le molecole e le loro interazioni. Un grafo delle attività è costruito eseguendo delle analisi sintattiche sul modello PEPA. Le simulazioni stocastiche e le equazioni differenziali sono derivate dal grafo delle attività. Illustriamo questo stile di modellazione con il problema reale di modellare i cicli circadiani (Goldbeter, 2002). Mostriamo alcuni svantaggi del nostro approccio e alcune soluzioni ad hoc. Infine presentiamo alcuni esperimenti utili per capire le relazioni tra le simulazioni stocastiche e deterministiche. Il nostro approccio produce risultati simili alla letteratura (Gonze et al., 2002a,b).

Nella terza parte analizziamo le relazioni tra le catene di Markov con livelli discreti e le equazioni differenziali. In primo luogo proponiamo alcuni semplici esempi, risolviamo i corrispondenti modelli markoviani e deterministici in modo analitico e proviamo che per questi particolari modelli il comportamento medio delle catene di Markov converge verso la soluzione dei sistemi deterministici quando il numero di livelli cresce. In secondo luogo, cerchiamo di generalizzare i risultati precendenti.

Una famiglia di catene di Markov X_v dipendenti dalla densità è una sequenza $\{X_v\}$ di processi di Markov definiti da un parametro v. Gli stati delle catene di Markov sono normalizzati rispetto a v e viene considerato lo spazio degli stati di questa specie di densità. Le velocità di transizione dipendono dalle densità, da cui il nome catene di Markov dipendenti dalla densità. Il teorema di Kurtz (Kurtz, 1970) dice che quando il parametro v diventa arbitrariamente grande, la sequenza di processo stocastici $v^{-1}X_v(t)$ converge al processo deterministico $X(t)$ che è soluzione di un sistema

di equazioni differenziali ordinarie. Mentre i sistemi con v finito sono processi discreti, il sistema limite è continuo.

La rappresentazione markoviana di un modello PEPA con livelli discreti è una catena di Markov parametrizzata $X_N(t)$, dove N è il numero di livelli. Però, la sequenza $\{X_N(t)\}$ non è sempre dipendente dalla densità. Infatti quando alcuni prodotti di una reazione hanno livello N, le velocità di transizione corrispondenti alla reazione sono zero. Formuliamo una condizione strutturale sufficiente e necessaria sullo spazio degli stati di $X_N(t)$ in modo tale che la sequenza $\{X_N(t)\}$ diventi una sequenza di catene di Markov dipendenti dalla densità e il teorema di Kurtz possa essere applicato. Proponiamo un metodo effettivo per verificare se un modello soddisfa questa condizione. In parole povere utilizziamo una generalizzazione della nozione di grafo delle attività (in pratica una rete di Petri) per rappresentare gli spazi degli stati di tutte le catene di Markov. Infine, se il grafo delle attività presenta alcune proprietà strutturali (ovvero è *bounded*), la corrispondente sequenza è dipendente dalla densità.

Utilizziamo i nostri risultati per provare che la rappresentazione dell'ERK signalling pathway (Cho et al., 2003) con livelli discreti di concentrazione produce risultati simili al modello deterministico quando il numero di livelli è sufficientemente grande.

Chapter 3

Background

3.1 Introduction

In this chapter we explain some background material useful to understand the rest of the thesis. Readers who already know these subjects can overlook the chapter. We give a brief overview of process algebras and PEPA, Markov chains, nonlinear dynamic systems and reaction kinetics. Readers should have elementary knowledge of calculus and probability theory. In Appendix C a glossary of biological terms used in this thesis can be found.

3.2 Process Algebras

In computer science process algebras are formalisms to model concurrent systems. In a process algebra several independent subsystems, called processes or components, interact and communicate between them or are synchronized over a set of actions. Algebraic laws describe how processes are defined and how they can be modified. Formal specification allows us to reason about process properties like equivalence (i.e. bisimulation). Examples of process algebras are *CCS* (Milner, 1980), π-calculus (Milner, 1999) and PEPA (Hillston, 1995). In this dissertation we will consider just the last one.

3.2.1 Performance Evaluation Process Algebra

PEPA is a stochastic process algebra invented by Jane Hillston (Hillston, 1995) for modelling computer and communication systems. As in all process algebras, systems are formed by several components which can perform actions. Each action has a duration. For example the expression $(\alpha, r).P$ models a system which can undertake action α with rate r and becomes P. The duration usually is represented by a random variable with a negative exponential distribution. In other words, r is the parameter of the distribution $F(t) = 1 - e^{-rt}$.

We present an informal description of the language below. In Appendix A we discuss the operational semantics of PEPA. A more detailed explanation can be found in Hillston (1995). PEPA has five combinators: prefix, choice, constant, hiding and cooperation.

Prefix is the basic component to build up complex systems, the process $(\alpha, r).P$ carries out action α at rate r and then it behaves as P. The prefix (α, r) is termed activity while α is the action type and r the rate of the activity.

Choice models competition between two processes: the component $P + Q$ represents a system which may evolve either into P or Q.

Constant allows us to assign names to components, for example $X \stackrel{def}{=} (\alpha, r).P$ means that variable X behaves as process $(\alpha, r).P$.

Hiding is a mechanism to abstract away some aspects of a component's behaviour. For instance, the process $P \setminus \{\alpha\}$ hides the action α and prevents other processes from joining in.

Cooperation allows two processes to be synchronized over a set of actions. In expression $P \underset{L}{\bowtie} Q$ processes P and Q must cooperate on actions contained in the set L, but other enabled actions are carried out independently and concurrently. When L is empty, we write $P \| Q$ instead of $P \underset{L}{\bowtie} Q$.

When a component enables an activity whose action type is in the cooperation set, it will be stuck until the other component enables an activity of that type. The rates of shared activities depend on the rate of both cooperands' rates. In other words, the apparent of a shared activity is the rate of the slower component.

Sometimes a component may be passive with respect to an action in a cooperation set. Consider

for example a client waiting for a service. In these cases the rate of the activity is unspecified (symbol $\top$) and it depends on the rate of the activity of the other cooperand. All passive actions must be synchronized in the final model.

3.3 Continuous Time Markov Chains

A continuous time Markov chain (CTMC) is a stochastic process $\{X(t) : t \geq 0\}$ that has the Markov property and takes values from a discrete set called the state space. The Markov property states that at any times $s > t > 0$, the conditional probability distribution of the process at time s given the whole history of the process up to time t, depends only on the state of the process at time t. Formally a stochastic process $X(t)$ is a Markov process if and only if for all $s, t \geq 0$ and states j, i and l,

$$P(X(s+t) = j | X(s) = i, X(u) = l \; 0 \leq u < s) = P(X(s+t) = j | X(s) = i)$$

A Markov process is time homogeneous if the transition rates are independent of the time of occurrence of each transition, i.e. $P(X(s+t) = j | X(s) = i) = P(X(t) = j | X(0) = i)$. In this work we will consider only time homogeneous Markov chains.

Continuous time Markov chains are described by the infinitesimal generator matrix Q. An element $q_{ij} \geq 0$ with $i \neq j$ of Q is the transition rate between states i and j. Instead diagonal elements q_{ii} are defined as $-\sum_{i \neq j} q_{ij}$.

The time dependent probability is the solution of the following differential equations, called Chapman-Kolmogorov equations,

$$\frac{\partial \pi(t)}{\partial t} = \pi(t) Q$$

The stationary probability distribution π is the solution of the linear system $\pi Q = 0$ subject to

normalization condition $\sum_j \pi_j = 1$. A Markov chain is irreducible if all states can be reached from all other states. A Markov chain is positive recurrent if starting in any state the expected time to return to that state is finite. A steady state probability distribution is the probability of being in a state in the long run. If a Markov chain is irreducible and positive recurrent the steady state corresponds to the stationary probability, more formally,

$$\lim_{t\to\infty} \pi(t) = \pi$$

3.4 Kinetics of chemical reactions

A chemical reaction is a process where one or more chemical substances, called reactants, yield one or more products. The rate of a reaction describes how the concentration of the involved substances changes in time. Chemical kinetics study the reaction rates in a chemical reaction. Reaction rates depend on several factors (e.g. temperature, concentration of reactants, pressure and so on), however we will assume most of them to be constant. Since several chemical kinetics occur in this work, we will give a brief overview in this section.

3.4.1 Mass Action Law

The Mass Action Law states that the rate of a chemical reaction is proportional to the probability of finding all the reacting molecules in a small space. Because we assume that the event of finding one molecule in a small space is independent of finding another molecule in the same space, the probability of finding both of them in the same space is given by the product of their individual probabilities. The probability of finding a molecule in a small volume is proportional to its concentration. Hence, the rate of a reaction is proportional to the product of the concentrations of each reactant molecules (Cox and Nelson, 2005).

As an example, consider the following simple reaction where molecules A and B bind toghether

and form complex C.

$$A + B \quad \rightarrow \quad C$$

The rate of change of C concentration is represented by a differential equation,

$$\frac{d[C]}{dt} \quad = \quad k[A][B]$$

where k is a constant, $[A]$, $[B]$ and $[C]$ are the concentrations of A, B and C respectively. By abuse of notation, we will often drop concentration brackets $[\cdot]$ when it is clear from the context we are dealing with concentrations.

The basic assumption behind mass action is that individual molecules act randomly, but if taken as a whole, they will tend to a deterministic law.

3.4.2 Michaelis-Menten

Michaelis-Menten kinetics describe the rate of enzyme driven reactions when the concentration of enzyme is much less than the concentration of substrate.

An enzymatic reaction consists of the following elementary reaction steps,

reaction	name	rate
$E + S \rightarrow ES$	binding	k_1
$ES \rightarrow E + S$	unbinding	k_{-1}
$ES \rightarrow E + P$	catalysis	k_2

Here, E, S, ES and P represent the enzyme, the substrate, the enzyme-substrate complex and the final product respectively. These reactions can be modelled as a set of differential equations using the mass action law. However it is often useful to simplify them with a single equation given by,

$$\frac{dP}{dt} = V_{MAX}\frac{S}{K_M + S} \tag{3.1}$$

where $K_M = \frac{k_{-1}+k_2}{k_1}$ is the Michaelis-Menten constant and V_{MAX} the maximum rate of reaction. Equation 3.1 is based on a steady state approximation. In other words after an initial period the concentration of complex ES is assumed to be constant, that is $dES/dt \simeq 0$. While both K_M and V_{MAX} can be determined experimentally, sometimes the elementary rates are not available so that it is not always possible to develop a system into elementary steps.

3.4.3 Inhibition

In some sense an enzyme is an activator of a reaction. Instead some molecules act as repressors. For instance, let us consider a protein P which represses the transcription of a gene. Given the maximal transcription rate constant V_{MAX} and an affinity constant K_I, the rate of change of mRNA concentration M is modelled by the following differential equation,

$$\frac{dM}{dt} = V_{MAX}\frac{K_I}{K_I + P} \tag{3.2}$$

We observe that the right hand side of Equation 3.2 is the complement of that in Equation 3.1 where K_I is used instead of K_M.

3.4.4 Cooperative binding and Hill Coefficient

A macromolecule is said to have cooperative binding if the affinity of the ligand for the molecule depends on the amount of ligands already bound. The cooperativity is positive if the binding of the ligand increases affinity for ligand, negative otherwise. A macromolecule can be also noncoperative, in this case the amount of ligands does not change the binding affinity.

The degree of cooperativity is quantified by the Hill Coefficient. A coefficient of one indicates

completely independent binding. Values greater than one indicate positive cooperativity, while numbers less than one indicate negative cooperativity.

As an example, if binding of repressor protein to enzyme in the inhibition model is cooperative with degree n, Equation 3.2 becomes,

$$\frac{dM}{dt} = V_{MAX}\frac{K_I^n}{K_I^n + P^n} \tag{3.3}$$

In Figure 3.1 we plot Equation 3.3 for several values of n. For $n = 1$ the curve is a typical hyperbolic plot and there is no cooperative effect. For $n > 1$ the graph is sigmoidal and shows positive cooperation; in fact the more the concentration of protein P is, the faster the rate decreases and transcription is inhibited. On the contrary if $n < 1$ the curve shows negative cooperativity: it has a faster initial fall, but it tends towards zero less sharply. As one can easily prove, the point of intersection of all curves is K_I.

3.5 Dynamical systems

A dynamical system describes a system which evolves in time. The state of the system is represented by a collection of real numbers. A deterministic rule says what future state follows the current state. For a more detailed tratement on this subject see Kuznetsov (2004).

3.5.1 Phase analysis

We consider a system of ordinary differential equations,

$$\frac{dY(t)}{dt} = F(Y(t), \Theta)$$

At any time t the state of the system is given by the vector $Y(t) = (y_1(t), \ldots, y_n(t))$. In this thesis $Y(t)$ is always in $\mathbb{R}^n$ and function $y_i(t)$ represents concentration of the i-th molecule kind at

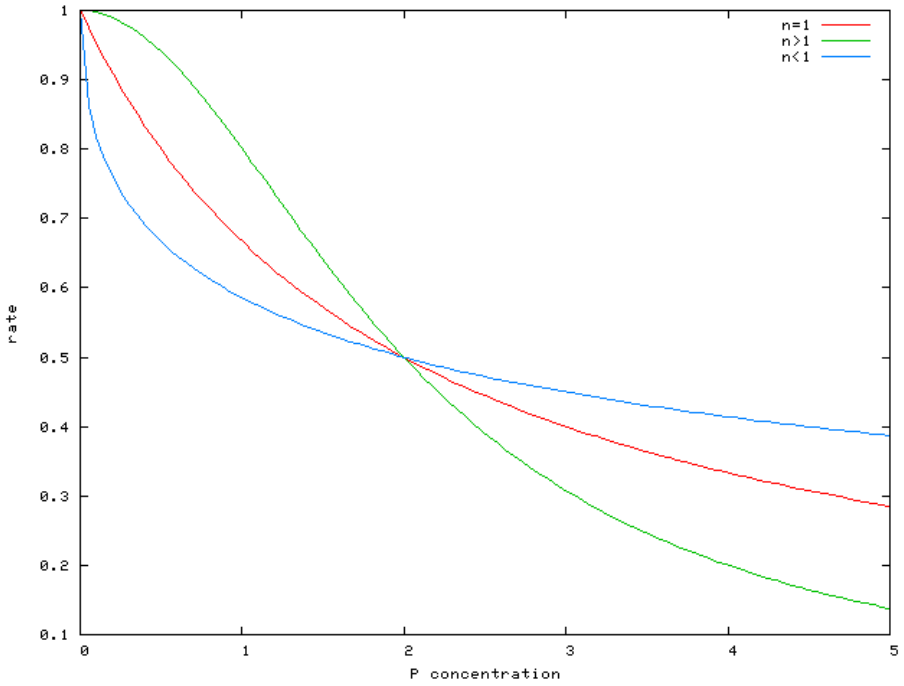


Figure 3.1: Cooperative inhibition for different degree n.

time t. The phase space is the n-dimensional space $Y \subseteq \mathbb{R}^n$ consisting of all the possible values of $(y_1(t), \ldots, y_n(t))$. When we consider just two dimensions, we call it phase plane. A trajectory or orbit is the sequence of points through which the system passes as it evolves.

Function $F : Y \times P \rightarrow \mathbb{R}^n$ describes the dynamic behaviour of the system where $Y \subseteq \mathbb{R}^n$ denotes the state space and P a set of parameter subsets (e.g., constant rates). Given a set of parameter values $\Theta \in P$ the current state $Y(t)$ is associated with the system rates of change.

The vector $dY(t)/dt = (y_1'(t), \ldots, y_n'(t))$ is sometimes called the velocity vector and describes how the system evolves given its current state. The vector field is the phase space where every point has associated its velocity vector. A flow is the set of all possible trajectories. A flow and a vector field give us an idea about the structure of the solution set of the ODEs and they are useful tools to unravel the dynamics of nonlinear systems.

An equilibrium point (also known as stagnation point, steady state[1], fix point or singular point)

[1]The term “steady state” is used also in markov chain theory to denote a different concept. In order to avoid ambiguity

is an element of the phase space where the velocity vector is zero.

A (stable) attractor is a set of points toward which ”neighboring” points approach in the course of time evolution of the system.

A y_i-nullcline is the set of points $(y_1, \ldots, y_n)$ of the phase space which satisfy $y_i' = f_i(y_1, \ldots, y_n) = 0$. The intersection of all the nullclines is an equilibrium point.

3.5.2 (Linear) Stability analysis

Equilibria are not always stable. The following table is a powerful method to check if equilibrium $\overline{y}$ is stable. Here, $J(\overline{y})$ is the Jacobian matrix,

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial y_1} & \frac{\partial f_2}{\partial y_2} & \cdots & \frac{\partial f_n}{\partial y_n} \\ \frac{\partial f_1}{\partial y_1} & \frac{\partial f_2}{\partial y_2} & \cdots & \frac{\partial f_n}{\partial y_n} \\ \vdots & \vdots & & \vdots \\ \frac{\partial f_1}{\partial y_1} & \frac{\partial f_3}{\partial y_3} & \cdots & \frac{\partial f_n}{\partial y_n} \end{pmatrix}$$

evaluated in the equilibrium point $\overline{y}$.

eigenvalues of $J(\overline{y})$	fixed point
complex with positive real part	unstable focus
complex with negative real part	stable focus
real and positive	unstable node
real and negative	stable node
positive and negative	saddle point

The case of one or more zero eigenvalues is much more complicated. There exist other methods to check stability; for instance Liapunov functions are useful if one wants to understand how points far from an equilibrium behaves with respect to the equilibrium.

---

we will use equilibrium or fix point when we talk about differential equations and steady state for the markov chains.

3.5.3 Bifurcation analysis

Most dynamical systems contain parameters. In our example Θ is a set of parmeters on which the solutions of the system depend. Changes in the parameter values may produce qualitative changes in the phase space and the dynamical system is said to have gone through a bifurcation. Bifurcation theory studies the system behaviour as a function of one or more parameters.

In this work we are interested in Hopfs bifurcation points. In a Hopfs bifurcation, as one increases the value of some parameters, a stable focus becomes unstable and the attractor becomes a limit cycle. A limit cycle is a periodic solution described by a closed curve in the phase plane or by sustained oscillations in time dependent graphs.

3.6 Stochastic simulation

We consider a system of $m_1, \ldots, m_l$ molecule species and $r_1, \ldots, r_k$ reactions between them. We assume the system is well-stirred, in thermal equilibrium and limited in a constant volume Ω. $X_i(t)$ denotes the number of molecules m_i in the system at time t and $X(t) = (X_1(t), \ldots, X_l(t))$. Given an initial state $X(t_0) = x_0$, we want to compute the probability $P(x, t_0 + dt|x_0, t_0)$ that $X(t)$ will be equal to x at time $t_0 + dt$ given that $X(t_0) = x_0$. The time evolution of this function is described by the Chemical Master Equation (CME), defined as,

$$\frac{\partial P(x, t|x_0, t_0)}{\partial t} = \sum_{j=1}^{k} [a_j(x - \nu_j)P(x - \nu, t|x_0, t_0) - a_j(x)P(x, t|x_0, t_0)]$$

Here, ν_j is a vector $(\nu_{j1}, \ldots, \nu_{jl})$ where ν_{ji} represents the change in the population of molecule species m_i caused by a reaction r_j. The propensity function $a_j(x)$ is defined so that $a_j(x)dt$ is the probability that a reaction r_j will occurr in time interval $[t, t + dt)$ given that $X(t) = x$.

The CME can be solved analytically or numerically for only few cases. Therefore we need another method. The idea is to derive a *numerical realization* of $X(t)$, namely a possible trajectory

of $X(t)$ versus time. Information on the underlying distribution can be inferred from averaging the results of many realizations or simulations. The Gillespie algorithm produces a realization for the CME. It is useful when the number of molecules involved in the system is small, i.e. in the order of $10\text{-}10^2$ individuals. For more details readers can look at Gillespie (1976).

In order to define a stochastic simulation we give the corresponding set of reactions and their probabilities of occurence in infinitesimal time interval dt. Following Gillespie (1976), the occurence probability of a reaction is given by the product of the number of the reagents involved in the reaction multiplied by the constant reaction rate.

Chapter 4

Modelling biological systems in PEPA

4.1 Introduction

In this chapter we present the modelling style introduced in Calder et al. (2004). We use the PEPA process algebra Hillston (1995). Readers who do not have familiarity with this process algebra can have a look at the introductory explanation in Chapter 3 and in Appendix A; here, some definitions of functions and sets (e.g. ds, Act, A)used in this thesis can be found, too.

The rest of this chapter is organized as follows. In Section 4.2 we describe the structure of a model where processes represent molecules and each process has an internal state that models the concentration level of the corresponding molecule. In Section 4.3 we introduce the concept of activity graph, a graphical representation of the model, and we describe how to derive ODEs and stochastic simulation from it. In Section 4.4 we illustrate this modelling approach on a model for circadian clocks. We show that processes can represent not only discrete levels of molecule population, but also activiation levels of abstract biological processes. Finally we present some experiments and results that describe the relationships between stochastic and deterministic simulations.

4.2 Model structure and biological interpretation

Molecule concentrations are divided into discrete intervals. Processes represent molecules and each process has an internal state that models the concentration level of the corresponding molecule. Here we consider just high and low concentration levels. For example processes $m[1]$ and $m[0]$ denote, respectively, high and low concentration of molecular species m.

A high level means that the concentration of a reagent is observable and the reagent can take part to rections; a low level, instead, implies that the concentration of a reagent is not observable and the regent can participate to reactions only as a product.

Reactions are modelled by activities. For example, $m[1] \stackrel{def}{=} (\alpha, r).m[0]$ means that reaction α occurs with rate r and it decreases the level of molecule m (from high to low). A molecule m is involved in reactions of three kinds: reactions which decrease the level of m (e.g. $m[1] \stackrel{def}{=} (\alpha, r).m[0]$), reactions which increase the level of m (e.g. $m[0] \stackrel{def}{=} (\alpha, r).m[1]$) and reactions which do not modify the level of m (e.g. $m[1] \stackrel{def}{=} (\alpha, r).m[1]$). Other activities are not allowed. More formally, for any molecule species m, we define two processes $m[1]$ and $m[0]$ such that:

- there exists $a \in Act(m[i])$ such that $m[i] \stackrel{a}{\rightarrow} m[1-i]$ for $i = 0, 1$;
- if there exists $a \in Act(m[1])$ such that $m[1] \stackrel{a}{\rightarrow} c$, then $c = m[1]$ or else $c = m[0]$;
- if there exists $a \in Act(m[0])$ such that $m[0] \stackrel{a}{\rightarrow} c$, then $c = m[1]$.

The first point says that a process $m[1]$ ($m[0]$) always evolves into its complement $m[0]$ ($m[1]$). The last two points formalize the idea that other transitions are not allowed; the exception is $m[1]$ which can become $m[1]$. As a consequence $ds(m[1]) = ds(m[0]) = \{m[0], m[1]\}$. In a biological context these conditions mean that a molecule has to be consumed and produced in at least one reaction of the system.

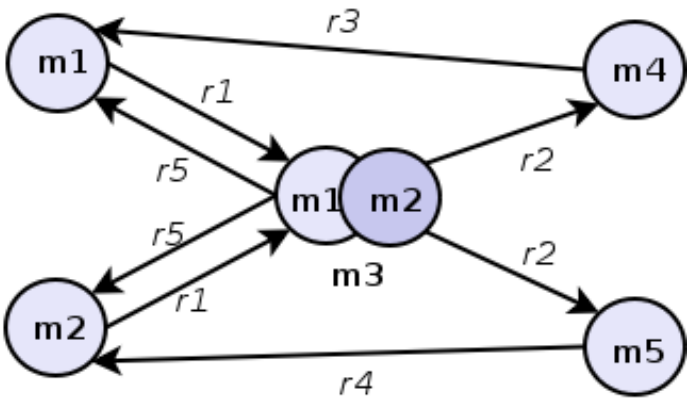
In a biological context a reaction does not have different possible effects on a molecule population. Thus we assume that for any molecule species m instances of activities of the same action

type cannot be used within components $m[0]$ and $m[1]$. In other words sequential components do not have multiple instances of activities of the same action type, see Appendix A.

In the system equation, molecules with an initial concentration are initially high in the system equation, while all others are low. We assume activities of the same action type α are synchronized on α and have the same rate v_α. Therefore, instead of system equations we will often use an initial concentration vector which records initial levels for each molecules.

An example

Figure 4.2 describes a simple biological system. There are five molecule species m_1, m_2, m_3, m_4 and m_5. Molecules m_1 and m_2 bind together to form complex m_3. Molecule m_3 can split into m_1 and m_2 again or else into two new molecules m_4 and m_5 which become m_1 and m_2 respectively.



For $i = 1 \ldots 5$ we define two processes $m_i[0]$ and $m_i[1]$ representing low and high concentration of molecule m_i. For each reaction we define an action type r_1, r_2, r_3, r_4 and r_5; we assume v_{r_j} is the rate for reaction r_j where $j = 1 \ldots 5$. The corresponding PEPA model is given by the following equations.

$$
\begin{aligned}
m_1[0] &\stackrel{def}{=} (r_3, v_{r_3}).m_1[1] + (r_5, v_{r_5}).m_1[1] \\
m_1[1] &\stackrel{def}{=} (r_1, v_{r_1}).m_1[0] \\
m_2[0] &\stackrel{def}{=} (r_4, v_{r_4}).m_2[1] + (r_5, v_{r_5}).m_2[1] \\
m_2[1] &\stackrel{def}{=} (r_1, v_{r_1}).m_2[0] \\
m_3[0] &\stackrel{def}{=} (r_1, v_{r_1}).m_3[1] \\
m_3[1] &\stackrel{def}{=} (r_2, v_{r_2}).m_3[0] + (r_5, v_{r_5}).m_3[0] \\
m_4[0] &\stackrel{def}{=} (r_2, v_{r_2}).m_4[1] \\
m_4[1] &\stackrel{def}{=} (r_3, v_{r_3}).m_4[0] \\
m_5[0] &\stackrel{def}{=} (r_2, v_{r_2}).m_5[1] \\
m_5[1] &\stackrel{def}{=} (r_4, v_{r_4}).m_5[0]
\end{aligned}
$$

In the system equation we assume levels of m_1 and m_2 are initially high.

$$m_1[1] \underset{\{r_1,r_3,r_4,r_5\}}{\bowtie} m_2[1] \underset{\{r_3,r_4,r_5\}}{\bowtie} m_3[0] \underset{\{r_3,r_4\}}{\bowtie} m_4[0] \underset{\{r_4\}}{\bowtie} m_5[0]$$

4.3 Activity graph

Our definition of an activity graph is slightly different from the one in Calder et al.. Roughly speaking an activity graph is a representation of the relations between reagents and reactions. More formally,

Definition 4.3.1. *An activity graph is a directed bipartite graph $G = (V, E)$. Node set V is equal to $R \cup M$ where R is the set of action types (i.e. reactions) and M is the set of process names (i.e.*

molecules). Edge set $E \subseteq (R \times M) \cup (M \times R)$ *is the smallest set such that:*

- *if molecule m is consumed in reaction* α*, then* $(m, \alpha) \in E$*;*
- *if molecule m is produced in reaction* α*, then* $(\alpha, m) \in E$*;*
- *if molecule m is involved in reaction* α *but it is not consumed, then* $(m, \alpha) \in E$ *and* $(\alpha, m) \in E$.

The graph is represented by an activity matrix $A = A^+ - A^-$ of size $|M| \times |R|$. Here $A^+ = \{a^+_{ij}\}$ where $a^+_{ij} = 1$ if $(j, i) \in E$, else $a^+_{ij} = 0$. Instead $A^- = \{a^-_{ij}\}$ where $a^-_{ij} = 1$ if $(i, j) \in E$, else $a^-_{ij} = 0$.

4.3.1 Deriving ODEs

We show how ordinary differential equations can be derived from an activity graph automatically. We assume M is a set of molecules, R a set of reactions and A an activity matrix built as described above. We assume m_i is the i-th element of M and r_i is the i-th element of R. Moreover v_i is the constant rate associated with reaction r_i.

For each process name $m_i \in M$ the rate equation corresponding to the concentration change of molecule m_i in infinitesimal time dt is given by

$$\frac{dm_i(t)}{dt} = \sum_{j=1}^{|R|} v_j a_{ij} \prod_{l=1}^{|M|} m_l(t)^{a^-_{ij}}$$

Here $m_i(t)$ represents a function whereas m_i is a process name. This is an abuse of notation, but it is often clear from the context their meaning. In this formula indeterminate form 0^0 may appear; following Knuth (1992) we assume $0^0 = 1$.

This system of ordinary differential equations is not complete because initial conditions must be specified. Initial conditions depend on the interpretation of the abstract concentrations “high” and “low”.

4.3.2 Deriving stochastic simulations

We derive a stochastic simulation in two steps. First, we build reaction equations from the structure of an activity graph. Second, we compute stochastic rates using information on activity rates.

Given a PEPA model we assume R is the set of actions (i.e. reactions) and M the set of process names (i.e. molecules). We assume m_i is the i-th element of M and r_i is the i-th element of R. Moreover v_i is the constant rate associated with reaction r_i. Let $G = (M \cup R, E)$ be an activity graph. Then for each action $r \in R$ the corresponding reaction equation is given by

$$\sum_{(m,r)\in E} m \rightarrow \sum_{(r,m)\in E} m$$

We need to transform activity rates into "stochastic" rates in order to compare results of deterministic and stochastic models. As an example consider a simple bimolecular reaction r of the form $A + B \rightarrow C$. As we saw in Chapter 3 the reaction can be represented by a deterministic reaction rate,

$$\frac{d|C|}{dt} = k|A||B|$$

Here $|A|$, $|B|$ and $|C|$ denote concentrations of molecules A, B and C respectively, while k is the constant "deterministic" rate for reaction r.

In a stochatic context $\sharp A$, $\sharp B$ and $\sharp C$ represent numbers of molecules A, B and C respectively. Sometimes we write $\sharp A(t)$ for the number of molecules A at time t. Parameter Ω models the volume of the cell, so $\sharp A = \Omega|A|$, $\sharp B = \Omega|B|$ and $C = \Omega|C|$. Therefore as Ω increases, the number of molecules becomes larger

We assume reaction r occurs with average probability c. Thus $c\ \sharp A\ \sharp B\ dt$ is the probability that r occurs once somewhere in the cell in time interval dt. If dt is *small*, two reactions cannot occur in dt. Therefore the average number of new molecules produced in dt is given by

$\sum_{i=0}^{\infty} iP[$ "i reactions in dt" $] = c \sharp A \sharp B\ dt$. In other words,

$$\sharp C(t + dt) - \sharp C(t) = c \sharp A \sharp B\ dt$$

Converting individuals into concentrations we obtain,

$$\Omega(|C(t + dt)| - |C(t)|) = c\ \Omega^2\ |A(t)||B(t)|dt$$

Dividing by dt and Ω and then taking $dt \to 0$, we have

$$\frac{d|C|}{dt} \quad = \quad c\ \Omega\ |A||B|$$

From the equation above we infer that the relation between stochastic rate c and "deterministic" rate k is given by $k = c\Omega$. In general given an n molecular reaction the deterministic rate k is equal to $c\Omega^{n-1}$ where c is the stochastic rate.

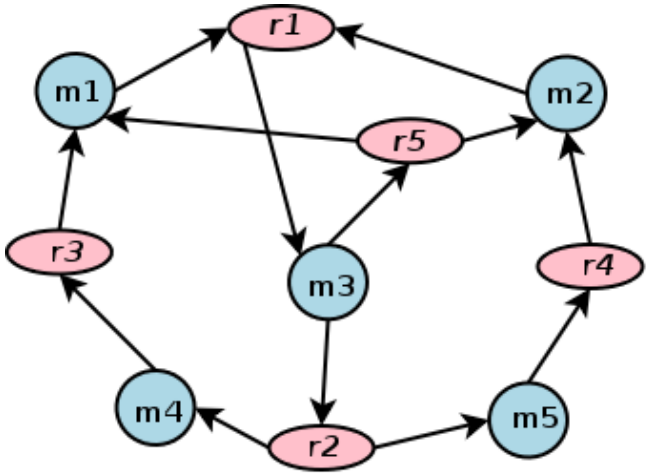
Hence, we are ready to explain how to compute stochastic rates for a PEPA model. Given the activity rate v_i, the corresponding stochastic rate is given by $1/\Omega^{n-1}v_i$, where $n = \sum_{j=1}^{|M|} a_{ji}^{-}$.

Again, the stochastic formulation is not complete. We have to specify initial values for variables. Initial values depend on the interpretation of the abstract levels. We can derive these values from the deterministic representation of the model following the rule $\sharp A = \Omega\ |A|$.

An example (continued)

The activity graph of the model in Figure 4.2 is given in Figure 4.3.2.

We derived ordinary differential equations and a stochastic simulation from the activity graph as described in previous sections. Here there is the deterministic model represented by differential



equations. Initial conditions may be inferred from experimental data, say $m_1(t) = m_2(t) = c > 0$ and $m_3(t) = m_4(t) = m_5(t) = 0$.

$$
\begin{aligned}
\frac{dm_1(t)}{dt} &= -v_{r_1} m_1(t) m_2(t) + v_{r_5} m_3(t) + v_{r_3} m_4(t) \\
\frac{dm_2(t)}{dt} &= -v_{r_1} m_1(t) m_2(t) + v_{r_5} m_3(t) + v_{r_4} m_5(t) \\
\frac{dm_3(t)}{dt} &= v_{r_1} m_1(t) m_2(t) - v_{r_5} m_3(t) - v_{r_2} m_3(t) \\
\frac{dm_4(t)}{dt} &= v_{r_2} m_3(t) - v_{r_3} m_4(t) \\
\frac{dm_5(t)}{dt} &= v_{r_2} m_3(t) - v_{r_4} m_5(t)
\end{aligned}
$$

Here there is a set of reactions for the stochastic simulations with the corresponding probabilities of occurring in infinitesimal time interval dt. In the last column m_i denotes the current number of molecule of species m_i. The initial population is given by $m_1 = m_2 = \Omega c$ and $m_3 = m_4 = m_5 = 0$.

name	reaction	probability
r_1	$m_1 + m_2 \rightarrow m_3$	$v_{r_1}/\Omega m_1 m_2$
r_2	$m_3 \rightarrow m_4 + m_5$	$v_{r_2} m_3$
r_3	$m_4 \rightarrow m_1$	$v_{r_3} m_4$
r_4	$m_5 \rightarrow m_2$	$v_{r_4} m_5$
r_5	$m_3 \rightarrow m_1 + m_2$	$v_{r_5} m_3$

4.4 A case study: circadian clocks

Circadian rhythms are 24 hour cycles shown by physiological processes in most living organisms. For example in animals feeding and sleeping are regulated by an internal clock with a period close to a day. Recent studies have discovered these rhythms depend on genetic mechanisms and rhythmic behaviours rely on the level of gene expression.

Several theoretical models have been proposed for circadian clocks in different organisms (Goldbeter, 2002). Following Gonze et al. (2002a) we consider the minimal model initially suggested for circadian rhythms in *Neurospora*, see Figure 4.1. Although this model is simple, it describes the basic structure for a wide range of biomolecular clocks (Young and Kay, 2001). Therefore it is useful to unravel the general machanisms underlying oscillations.

In Figure 4.1 we show the model for *Neurospora*. The core mechanism of circadian oscillations relies on the negative autoregulation of the clock gene. Gonze et al. (2002b) defined also a similar model which includes phosphorylation of cytosolic proteins and cooperative binding of repressor proteins to gene promoters. At the moment we are going to overlook these details because they are not required for the oscillations (Gonze et al., 2002a).

The time evolution of the concentrations involved in the model is given by the following kinetic

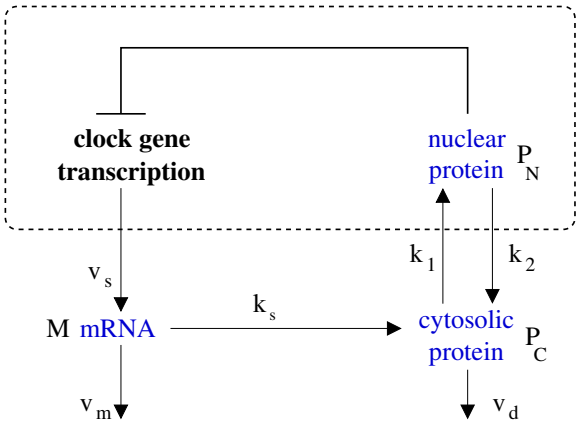


Figure 4.1: **Core model for circadian rhythms.**. M represents mRNA while P_C and P_N are the clock proteins into the cytosol and into the nucleous respectively. P_C is synthesized from the mRNA M, then it is either transported into the nucleus or degradated. P_N exerts a negative feedback on transcription of its gene or else it goes out of the nucleus. Degradations are controlled by enzymes.

equations,

$$\begin{aligned}
\frac{d[M]}{dt} &= v_s \frac{k_I^n}{k_I^n + [P_n]^n} - v_m \frac{[M]}{k_m + [M]} \\
\frac{d[P_C]}{dt} &= k_s M - v_d \frac{[P_C]}{k_d + [P_C]} - k_1[P_C] + k_2[P_N] \\
\frac{d[P_N]}{dt} &= k_1[P_C] - k_2[P_N]
\end{aligned}$$

In these equations, the variables $[M]$, $[P_C]$ and $[P_N]$ denote, respectively, the concentrations of the clock gene mRNA and of the clock gene protein in the cytosol and in the nucleus. Transcription is inhibited by protein P_N. Inhibition is described by term $\frac{k_I^n}{k_I^n + [P_n]^n}$ where k_I is a constant that measures the binding affinity between P_N and the clock gene, n is the Hill coefficient and v_s the maximal rate. Degradations of mRNA and cytosolic protein P_C are enzymatic reactions which follows Michaelis-Mentent rule. Parameters k_m and k_d are the respective Michaelis constants. whereas v_m and v_d are their maximal rates. The other reactions follow the Mass Action Law. In particular $k_s M$, $k_1[P_C]$ and $k_2[P_N]$ correspond to translation, transportation out of and into the nucleus of the clock protein respectively. With proper parameters the ODEs yield oscillations in the molecule concentrations with a period close to 24 hours.

4.4.1 Modelling circadian clock in PEPA

In this section we present a PEPA model of the minimal circadian clock in Figure 4.1. It is challenging to express in PEPA the behaviour described by nonlinear terms appearing in the kinetic equations B.1. These terms, however, do not correspond to single reaction steps. They rather are based on assumptions about the behaviour of enzyme-substrate or gene-repressor complexes. As suggested in Gonze et al. (2002b), we decompose enzyme-substrate and gene-repressor reactions into elementary steps; so we build a new model, which we will refer to as the *developed* model. We develop enzymatic degradation processes into three elementary reactions given by the following formula,

$$S + E_S \underset{u}{\overset{b}{\rightleftarrows}} C_S \xrightarrow{c} \overline{S} + E_S$$

These reactions represent *binding* (b) of substrate S to enzyme E_S to form complex C_S, *disassociation* (u) of C_S and *catalytic decomposition* (c) of C_S to form degradated product $\overline{S}$ and enzyme E_S. In the model degradations of cytosolic protein P_C and mRNA M are enzyme-driven. Repression mechanism is developed in two reaction steps given by the following formula,

$$G + P_N \underset{on}{\overset{off}{\rightleftarrows}} GP$$

Here, repressor protein P_N binds to its gene G to form complex gene-protein GP. Binding and unbinding of protein mean switching gene, respectively, off and on. For simplicity we assume Hill coefficient n is one.

The developed PEPA model is described by the component definitions 4.1 and by the system equation 4.2. As in Calder et al. (2004), the model is based on the variations in concentration of the reagents. Concentrations are represented by discrete values. We consider high (i.e. observable) and low (i.e. unobservable) concentrations of reagents. In contrast with Calder et al. (2004), however, we do not have just concentrations. Here, T and R does not represent concentrations of molecules, but the *effectiveness* of transcription and repression machinery. We have two levels, high and low,

for transcription and repression. For example, high transcription $T[1]$ means the transcription machinery is working at the best of its capability. The concentration M of mRNA depends on both of them.

$$
\begin{aligned}
T[1] &\stackrel{def}{=} (m, v_s).T[1] + (off, \top).T[0] \\
T[0] &\stackrel{def}{=} (on, \top).T[1] \\
R[1] &\stackrel{def}{=} (on, v_{on}).R[0] \\
R[0] &\stackrel{def}{=} (off, \top).R[1] \\
M[0] &\stackrel{def}{=} (m, \top).M[1] + (um, v_{um}).M[1] \\
M[1] &\stackrel{def}{=} (pc_1, k_s).M[1] + (bm, v_{bm}).M[0] \\
E_M[1] &\stackrel{def}{=} (bm, v_{bm}).E_M[0] \\
E_M[0] &\stackrel{def}{=} (um, v_{um}).E_M[1] + (cm, v_{cm}).E_M[1] \\
C_M[1] &\stackrel{def}{=} (um, v_{um}).C_M[0] + (cm, v_{cm}).C_M[0] \\
C_M[0] &\stackrel{def}{=} (bm, v_{bm}).C_M[1] \\
P_C[0] &\stackrel{def}{=} (pc_1, \top).P_C[1] + +(pc_2, k_2).P_C[0] + (upc, v_{upc}).P_C[1] \\
P_C[1] &\stackrel{def}{=} (pn, k_1).P_C[0] + (bpc, v_{bpc}).P_C[0] \\
E_{P_C}[1] &\stackrel{def}{=} (bpc, v_{bpc}).E_{P_C}[0] \\
E_{P_C}[0] &\stackrel{def}{=} (upc, v_{upc}).E_{P_C}[1] + (cpc, v_{cpc}).E_{P_C}[1] \\
C_{P_C}[1] &\stackrel{def}{=} (upc, v_{upc}).C_{P_C}[0] + (cpc, v_{cpc}).C_{P_C}[0] \\
C_{P_C}[0] &\stackrel{def}{=} (bpc, v_{bpc}).C_{P_C}[1] \\
P_N[0] &\stackrel{def}{=} (pn, \top).P_N[1] \\
P_N[1] &\stackrel{def}{=} (pc_2, k_2).P_N[0] + (off, v_{off}).P_N[0]
\end{aligned}
\tag{4.1}
$$

$$
T[1] \underset{J}{\bowtie} (R[0] \underset{K}{\bowtie} (((M[0] \underset{L}{\bowtie} E_M[1]) \underset{M}{\bowtie} C_M[0]) \underset{N}{\bowtie} (((P_C[0] \underset{O}{\bowtie} E_{P_C}[1]) \underset{P}{\bowtie} C_{P_C}[0]) \underset{Q}{\bowtie} P_N[0]))) \tag{4.2}
$$

Where $J = \{m, off, on\}$, $K = \{off\}$, $L = \{um, bm\}$, $M = \{um, bm, cm\}$, $N = \{pc_1\}$, $O = \{upc, bpc\}$,

$P = \{upc, bpc, cpc\}$ and $R = \{pn, pc_2\}$. In the initial state we have high concentrations of enzymes (E_M and E_{P_C}) and the transcription machinery T is working at high level; instead concentrations of the enzyme-substrate complexes (C_M and C_{P_C}) and of mRNA (M), cytosolic protein P_C and nuclear protein P_N are low.

4.4.2 Activity graph

In Figure 4.2 we show the activity graph corresponding to PEPA model 4.1.

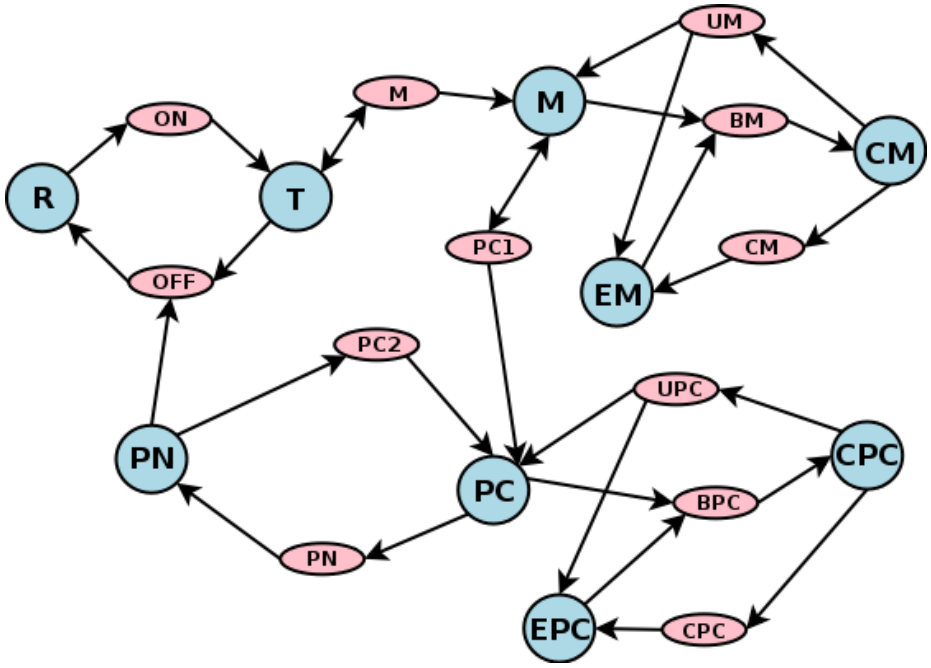


Figure 4.2: **Activity graph for the PEPA model of the circadian clock.**. This is the activity graph for the model in Figure 4.1 described by equations 4.1. Nodes are process names (i.e. molecules) and action type (i.e. reactions) in the algebraic specification. An edge goes out of a node-molecule and enters into a node-reaction if the molecule is consumed in the reaction. Instead an edge is from a node-reaction to a node-molecule if the molecule is produced in the reaction. Here nodes T and R do not represent molecules, but molecular processes.

As described previously we derive ODEs and a stochastic simulation from the activity graph. In Appendix B we listed the ODEs and the stochastic description of the model.

4.4.3 Some experiments and results

Ordinary differential equations were solved numerically using Runge Kutta 5 method. Reaction equations, instead, were used as input for Gillespie's algorithm. In this section we show some experiments and results.

In Figure 4.3 there are results for two deterministic models. The plot shows the time dependent behaviour of mRNA M, cytosolic protein P_C and nuclear protein P_N for the non developed model (left) and for the developed model (right). Results are similar but not equal. In fact the non developed version is an approximation of the developed one based on some steady state assumptions. Moreover there is a lack of data for some rates in the developed model.

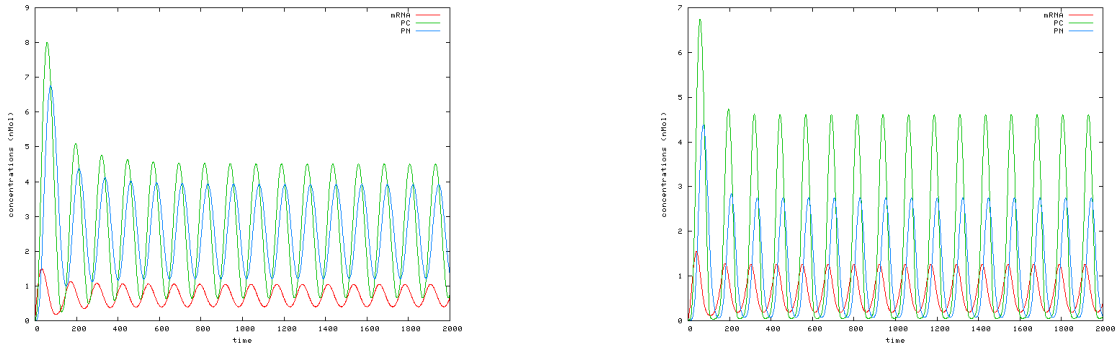


Figure 4.3: **Developed and non developed version of the core clock model**.

In Figure 4.4 we compare stochastic and deterministic simulations. The experiment shows that for sufficiently large values of Ω, a stochastic simulation yields results similar to the deterministic model. The left plot of each row shows the behaviours of mRNA M, cytosolic protein P_C and nuclear protein P_N in the determinisitic model (first row) and in the stochastic model (second row). The right plot, instead, shows the phase plane for mRNA M and nuclear protein P_N; after a while the system converges to a limit cycle.

In Figure 4.5 we show the effect of the number of molecules on noise in the stochastic model. As the value of Ω decreases, oscillations become noisy and the limit cycles disappears.

Since the deterministic model and the stochastic model produce similar result for $\Omega > 1000$, the

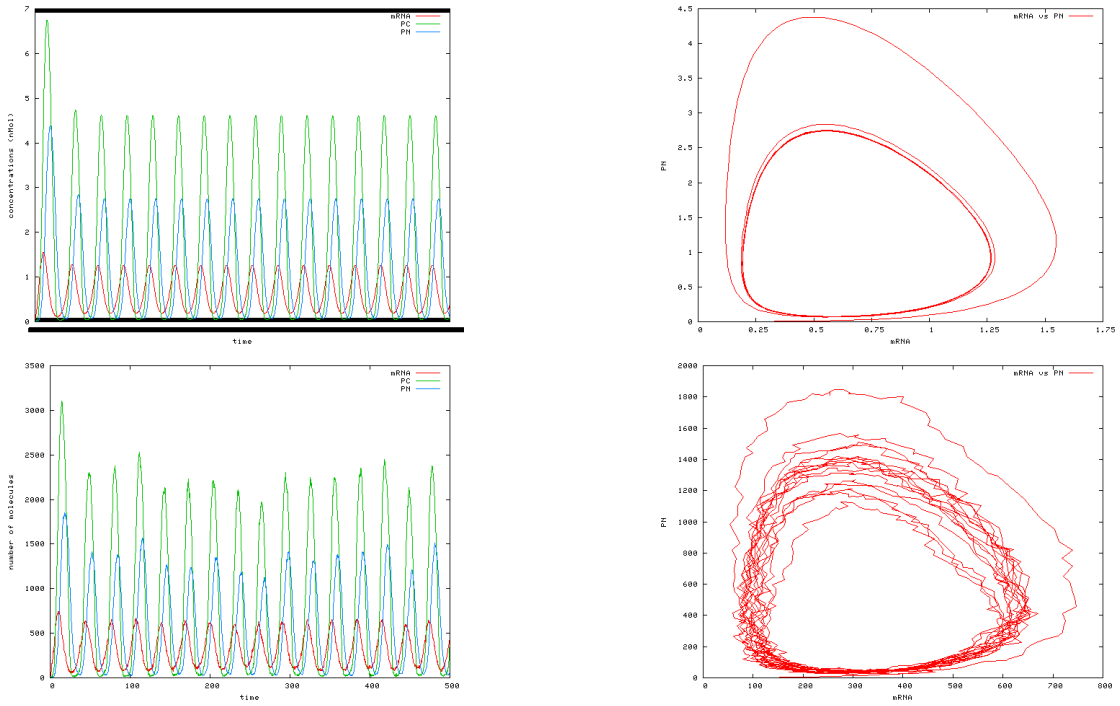


Figure 4.4: **Core model for circadian clock: stochastic and deterministic simulations**. The first row shows the behaviour of the deterministic representation of the core model. Results were produced numerically (Runge Kutta) from the ODEs in Appendix. On the left there is the time evolution of the system for mRNA M, cytosolic protein P_C and nuclear protein P_N; on the right instead there is the phase plane M vs P_N corresponding to the previous graph. The system converges toward a limit cycle. The second row shows the corresponding results for the stochastic model. Simulations were performed using Gillespie's algorithm and $\Omega = 500$.

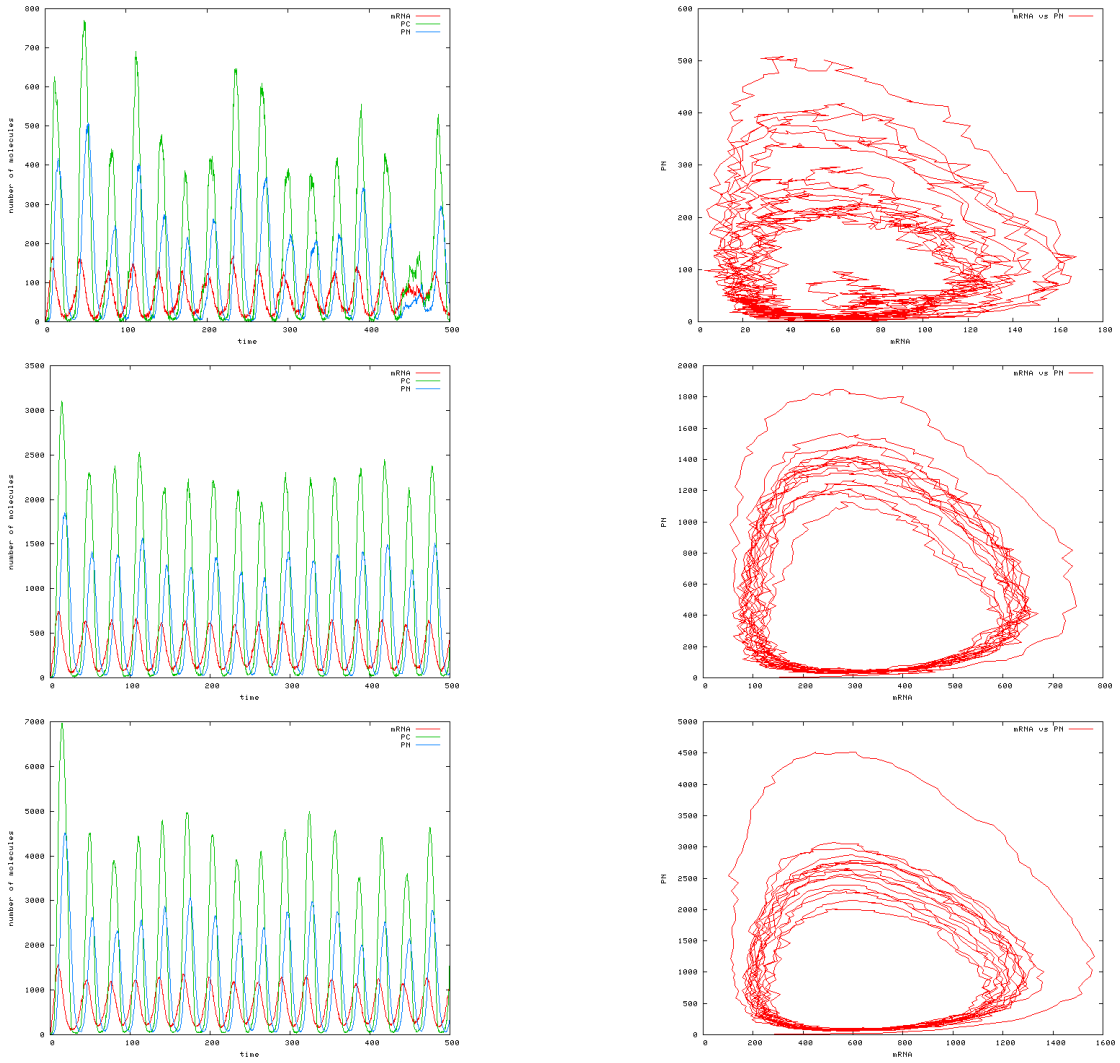


Figure 4.5: **Effect of the number of molecules on noise in the stochastic model**. The figure shows the results of stochastic simulations for increasing values of Ω (1000, 500, 100). The left plot describes the time behaviours of mRNA M, cytosolic protein P_C and nuclear protein P_N. The right one is the corresponding phase plane. The experiment was carried out using Gillespie's algorithm. We observe that for small values of Ω the limit cycles disappears.

deterministic model is not adequate. In fact it describes the behaviour of a system where the number of molecules is of order 10^3 while the number of molecules in regulatory gene networks is usually of order $10 - 10^2$.

Chapter 5

Markov chains with discrete levels and their approximations

5.1 Introduction

The modelling style based on discrete levels of concentration initially was introduced in order to derive ordinary differential equations from the high level specifications of PEPA models (Calder et al.). In this thesis we showed how it is possible to build stochastic simulations in a similar way.

Historically PEPA was designed to represent Markov chains (Hillston, 1995). Therefore in this chapter we analyse the interpretation of models with discrete levels as Markov processes and its relationships with the other interpretations.

There might exist several algorithms that extract different Markov chains from the same model with discrete levels of concentration. We consider just the Markov processes produced from a model with discrete levels interpreted as in Hillston (1995). We will refer to these chains with the term Markov chains with levels or, if there is no ambibuity, Markov chains.

A Markov chain with levels has the state space composed by tuples of nonnegative integers. Transitions are defined between adjacient tuples and rates depend on the current state. However it is not a population model, such as the Markov chain underlying a stochastic simulation, because

we assume there exists a maximal concentration. Moreover this assumption could make models not appropriate for some biological systems. The scope of this chapter is to clear also these aspects.

From some experiments we carried out we observed that a Markov chain based on two levels of concentration could not have enough useful information to understand the behaviour of the stystem. Hence in this chapter we generalize the approach presented in Chapter 4 to $N + 1$ discrete levels of concentration. We show how it is possible to derive an expanded model from a binary model. In order to compare results of different interpretations we need to adjust rates for a stochastic context as we did for Gillespie's algorithm in Chapter 4. Thus model expansion provides also some rules to compute proper rates.

We analyse the behaviour of Markov processes for increasing values of N. In the last part of this chapter we prove that under some conditions the average behaviour of a Markov chain with discrete levels converges to the solution of a system of differential equations.

5.2 Cooperation in biological systems

In this thesis we use a modified version of the cooperation rule defined in Appendix A. In the case of computers' systems the expression $P \underset{L}{\bowtie} Q$ represents a situation where P and Q must work together to undertake an action in L. Therefore the rate of a shared activity depends on the rate of the slower component.

On the contrary in a biological context we follow the Mass Action Law (see Chapter 3): the rate of a reaction is proportional to the product of the concentrations of each reactant molecule. Thus we override the definition of the apparent rate in the following way,

$$r_\alpha(P \underset{L}{\bowtie} Q) = \begin{cases} r_\alpha(P) \times r_\alpha(Q) & \text{if } \alpha \in L \\ r_\alpha(P) + r_\alpha(Q) & \text{if } \alpha \notin L \end{cases}$$

From the definition of cooperation rule given in Appendix A we obtain the following new rule for

cooperation,

$$\frac{E \xrightarrow{(\alpha,r_1)} E' \quad F \xrightarrow{(\alpha,r_2)} F'}{E \underset{L}{\bowtie} F \xrightarrow{(\alpha,r_1 r_2)} E' \underset{L}{\bowtie} F'} \ (\alpha \in L)$$

5.3 Expanded PEPA models

Because of some numerical experiments we carried out we claim that two levels of concentration may be not enough to derive useful information from models. Hence, we generalize the modelling approach based on two concentration levels to $N + 1$ levels.

Given a parameter $N > 0$, concentration interval $[0, M]$ is divided into $N + 1$ discrete concentration levels $l_0, \ldots, l_N$ and we assume the width between consecutive levels is equal to $H = M/N$. In this context N represents the greatest concentration level for each molecule. The discrete concentration corresponding to level l_i is given by $l_i M/N$. We want to study the behaviour of the system for increasing values of N.

We introduce a notation to simplify the description of models. $\sum_{i=0}^{k} C_i$ is an abbreviation for process $C_1 + \ldots + C_k$. The sum of zero terms is a process *nil* and, given a process C, we have $C + nil \equiv C$.

Given N and M we can expand a two level model into an N level model in the following way. We build the activity graph $G = (S \cup R, E)$ corresponding to the two level model as described in Chapter 4. Here S is the set of molecule species m_1, ..., $m_{|S|}$ whereas R is the set of reactions α_1, ..., $\alpha_{|R|}$.

Let m be a molecule in S. We denote with $m[i]$ the process corresponding to the i-th level of molecule m. For each molecule m we define the following process constants,

$$
\begin{aligned}
m[i] &\stackrel{def}{=} \sum_{(\alpha,m)\notin E\wedge(m,\alpha)\in E} (\alpha, iH).m[i-1] + \sum_{(m,\alpha)\notin E\wedge(\alpha,m)\in E} (\alpha, 1).m[i+1] \\
&\quad + \sum_{(m,\alpha)\in E\wedge(\alpha,m)\in E} (\alpha, iH).m[i] \quad \text{for } i = 1\ldots N-1 \\
m[N] &\stackrel{def}{=} \sum_{(m,\alpha)\in E\wedge(\alpha,m)\notin E} (\alpha, NH).m[N-1] + \sum_{(m,\alpha)\in E\wedge(\alpha,m)\in E} (\alpha, NH).m[N] \\
m[0] &\stackrel{def}{=} \sum_{(\alpha,m)\in E\wedge(m,\alpha)\notin E} (\alpha, 1).m[1]
\end{aligned}
$$

Because of the assumptions made in Chapter 4 on the structure of binary models, for any molecule m there exist reactions α_1 and α_2 such that $(m, \alpha_1) \in E$ and $(\alpha_2, m) \in E$. Therefore the model expansion is well defined, in the sense that $Act(m[i]) \neq \emptyset$ for every molecule m and $i = 0, \ldots, N$. Note that the property that every sequence component does not contain instances of the same action type is conserved in the expanded model.

Because of reasons which will be clear soon, we define also a dummy process D whose scope is to adjust rates.

$$
D \stackrel{def}{=} \sum_{\alpha\in R} (\alpha, \frac{v_\alpha}{H}).D
$$

Where v_α is the rate corresponding to reaction α. We remember that we assume activities of the same action type (i.e. reactions) have the same rate.

The system equation contains the components corresponding to the initial level for each molecule and component D. Initial levels depend on initial concentrations which are usually known. As usual we assume that activities with the same action type are synchronized. Therefore we omit action type sets under operator $\bowtie$.

It might not be clear the reason of division by H in the definition of component D. Constant

H is the step width $l_i - l_{i-1}$ for any $i = 1, \ldots, N$. In other words, changing the level of molecule m takes time Δt which is necessary to increase or decrease of H the concentration of m. Therefore the changing rate is given by $\frac{1}{\Delta t}$. Time Δt is computed from the differential equations; for instance consider the following rate equation,

$$\frac{dm(t)}{dt} = v_\alpha m(t)$$

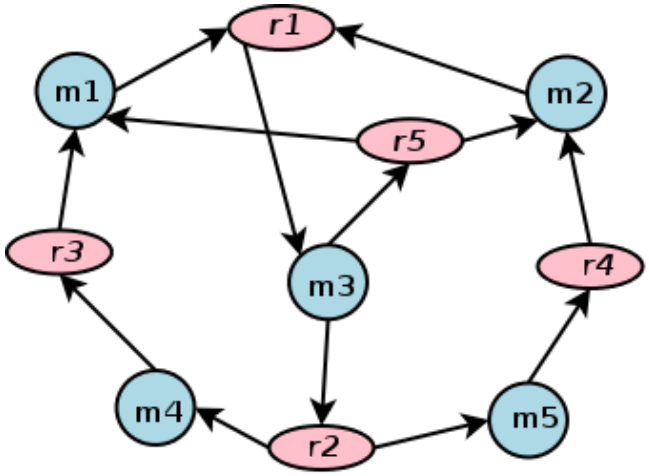
where $m(t)$ is the concentration of molecule m at time t and v_α is the rate of reaction α. If we discretize the equation, we will obtain for small values of Δt,

$$m(t + \Delta t) = m(t) - v_\alpha m(t)\Delta t$$

Because in our model the level step is H, we set $H = v_\alpha m(t)\Delta t$ or else $\Delta t = \frac{H}{v_\alpha m(t)}$. It follows the reaction rate is given by $\frac{v_\alpha m(t)}{H}$.

An example (continued)

Here we derive an expanded PEPA model for the example in Section 4.2. In Figure 5.3 we show the activity graph corresponding to the binary model. We assume all the molecules but m_1 and m_2 have low initial levels. The expanded model is derived from the activity graph as described above.



$$
\begin{aligned}
m_1[0] &\stackrel{def}{=} (r_3, 1).m_1[1] \\
m_1[i] &\stackrel{def}{=} (r_3, 1).m_1[i+1] + (r_1, iH).m_1[i-1] \quad \text{for } i = 1 \ldots N-1 \\
m_1[N] &\stackrel{def}{=} (r_1, NH).m_1[N-1] \\
m_2[0] &\stackrel{def}{=} (r_4, 1).m_2[1] \\
m_2[i] &\stackrel{def}{=} (r_4, 1).m_2[i+1] + (r_1, iH).m_2[i-1] \quad \text{for } i = 1 \ldots N-1 \\
m_2[N] &\stackrel{def}{=} (r_1, NH).m_2[N-1] \\
m_3[0] &\stackrel{def}{=} (r_1, 1).m_3[1] \\
m_3[i] &\stackrel{def}{=} (r_1, 1).m_3[i+1] + (r_2, iH).m_3[i-1] \quad \text{for } i = 1 \ldots N-1 \\
m_3[N] &\stackrel{def}{=} (r_2, NH).m_3[N-1] \\
m_4[0] &\stackrel{def}{=} (r_2, 1).m_4[1] \\
m_4[i] &\stackrel{def}{=} (r_2, 1).m_4[1] + (r_3, iH).m_4[i-1] \quad \text{for } i = 1 \ldots N-1 \\
m_4[N] &\stackrel{def}{=} (r_3, NH).m_4[N-1] \\
m_5[0] &\stackrel{def}{=} (r_2, 1).m_5[1] \\
m_5[i] &\stackrel{def}{=} (r_2, 1).m_5[1] + (r_4, iH).m_5[i-1] \quad \text{for } i = 1 \ldots N-1 \\
m_5[N] &\stackrel{def}{=} (r_4, NH).m_5[N-1] \\
D &\stackrel{def}{=} (r_1, \frac{v_{r_1}}{H}).D + (r_2, \frac{v_{r_2}}{H}).D + (r_3, \frac{v_{r_3}}{H}).D + \\
&\quad +(r_4, \frac{v_{r_4}}{H}).D + (r_5, \frac{v_{r_5}}{H}).D
\end{aligned}
$$

In the binary model the system equation is given by

$$m_1[1] \underset{\{r_1,r_3,r_5\}}{\bowtie} m_2[1] \underset{\{r_1,r_4,r_5\}}{\bowtie} m_3[0] \underset{\{r_2,r_3\}}{\bowtie} m_4[0] \underset{\{r_2,r_4\}}{\bowtie} m_5[0]$$

In the expanded model binary levels are substituted by proper values. In this case 0 remains the same, while 1 becomes N. Dummy process D has to be added. Therefore we obtain,

$$m_1[N] \underset{\{r_1,r_3,r_5\}}{\bowtie} m_2[N] \underset{\{r_1,r_4,r_5\}}{\bowtie} m_3[0] \underset{\{r_2,r_3\}}{\bowtie} m_4[0] \underset{\{r_2,r_4\}}{\bowtie} m_5[0] \underset{\{r_1,r_2,r_3,r_4,r_5\}}{\bowtie} D$$

As an example, a firing of reaction r_1 produces the following transition, as expected,

$$\begin{aligned} m_1[N] \underset{\{r_1,r_3,r_5\}}{\bowtie} m_2[N] \underset{\{r_1,r_4,r_5\}}{\bowtie} m_3[0] \underset{\{r_2,r_3\}}{\bowtie} m_4[0] \underset{\{r_2,r_4\}}{\bowtie} m_5[0] \underset{\{r_1,r_2,r_3,r_4,r_5\}}{\bowtie} D \xrightarrow{(r_1, \frac{v_{r_1}}{H} N^2 H^2)} \\ m_1[N-1] \underset{\{r_1,r_3,r_5\}}{\bowtie} m_2[N-1] \underset{\{r_1,r_4,r_5\}}{\bowtie} m_3[1] \underset{\{r_2,r_3\}}{\bowtie} m_4[0] \underset{\{r_2,r_4\}}{\bowtie} m_5[0] \underset{\{r_1,r_2,r_3,r_4,r_5\}}{\bowtie} D \end{aligned}$$

5.4 Markov chains with discrete levels of concentration

In Hillston (1995) the derivation graph represents all the possible behaviours of a system. Nodes are the derivatives of the initial component and there is an arc between nodes for each possible transition between the corresponding components. Besides each arc is labelled with its own activity. More formally we have the following definition.

Definition 5.4.1 (Hillston (1995)). *Given a PEPA component C and its derivative set $ds(C)$, the derivation graph $D(C)$ is the labelled directed multigraph whose set of nodes is $ds(C)$ and whose multiset of arcs A is defined as follows:*

- *the elements of A belong to the set $ds(C) \times ds(C) \times Act$;*

- $< C_i, C_j, a >$ *occurs in A with the same multiplicity as the number of disinct inference trees for derivation* $C_i \xrightarrow{a} C_J$.

A stochastic process is built assigning a state to each node of the graph and defining transitions for each arc of the graph. If all activity durations are exponentially distributed, it can be proved that this process is a Markov process (Hillston, 1995).

We want to study the structure of a Markov chain corresponding to a PEPA model with $N+1$ discrete levels of concentration. In this context system equations have the form $m_1[l_1] \underset{\_}{\bowtie} \ldots \underset{\_}{\bowtie} m_{|S|}[l_{|S|}]$ where $m_1, \ldots, m_{|S|}$ are the molecules of the system and $l_1, \ldots, l_{|S|}$ the corresponding levels. We omit the set of actions under operator $\underset{\_}{\bowtie}$ and process D for simplicity's sake.

Because a process $m_i[l_i]$ can evolve only into a process $m_i[l_i']$ with $l_i' = l_i$, $l_i + 1$ or $l_i - 1$, the state of the system can be represented by a row vector $w \in \mathbb{N}^{|S|}$ where entry w_i stands for the level of molecule m_i.

We define a function ω in order to switch from the representation of states as components to the notation with tuples of non negative integers. Given a component $C \equiv m_1[l_1] \underset{\_}{\bowtie} \ldots \underset{\_}{\bowtie} m_{|M|}[l_{|M|}]$, the corresponding vector $\omega(C)$ is defined as $(l_1, \ldots, l_{|M|})$. We define also the inverse function ω^{-1} such that given a vector w of size $|M|$ the corresponding component is $\omega^{-1}(w) = m_1[w_1] \underset{\_}{\bowtie} \ldots \underset{\_}{\bowtie} m_{|M|}[w_{|M|}]$.

We say that w' is reachable from w if and only if $\omega(w') \in ds(\omega(w))$. The set of all states reachable from w is denoted by $\rho(w)$.

Following Winskel (1993) $d \vDash C_i \xrightarrow{a} C_j$ denotes the fact that there exists a derivation tree d that infers transition $C_i \xrightarrow{a} C_j$. We want to prove that for every expanded model, if there exists a derivation tree for a transition, then this tree is unique. As a consequence the arcs of the derivation graph have multiplicity one. In other words given a component C_i we want to prove the following proposition

$$Q(C_i) \equiv \forall d_1, d_2 \; \exists C_j, a \; d_1 \vDash C_i \xrightarrow{a} C_j \wedge d_2 \vDash C_i \xrightarrow{a} C_j \Rightarrow d_1 = d_2$$

Lemma 5.4.2. *For every sequence component S* $Q(S)$ *holds.*

Proof. We prove it by induction on the structure of sequential components.

- If $S \equiv (\alpha, r).X$, we can apply only prefix rule, so the thesis follows immediately.
- If $S \equiv S_1 + S_2$, consider two derivation trees such that $d_1 \vDash S \xrightarrow{(\alpha,r)} S'$ and $d_2 \vDash S \xrightarrow{(\alpha,r)} S'$. Because a sequence component does not have multiple instances of activities of the same action type, without loss of generality we assume $\alpha \in A(S_1)$ and $\alpha \notin A(S_2)$. Thus we can apply only a rule to derive $S \xrightarrow{a} S'$,

$$\frac{S_1 \xrightarrow{(\alpha,r)} S'}{S_1 + S_2 \xrightarrow{(\alpha,r)} S'}$$

and because $Q(S_1)$ holds by inductive hypothesis, we conclude that $d_1 = d_2$ and $Q(S)$.

□

Proposition 5.4.3. *For every model component P $Q(P)$ holds.*

Proof. We prove it by induction on the structure of model components. We omit the case of hiding because it does not appear in our models.

- If $P \equiv X$ where X is the name of a sequential component S. Assume there exist two derivation trees $d_1 \vDash P \xrightarrow{(\alpha,r)} P'$ and $d_2 \vDash P \xrightarrow{(\alpha,r)} P'$. We can apply only the rule for constants. Because $Q(S)$ is true for every sequential component S, see previous lemma, we conclude that $d_1 = d_2$ and $Q(P)$ is also verified.
- If $P \equiv P_1 \underset{L}{\bowtie} P_2$, assume there exist two derivation trees $d_1 \vDash P \xrightarrow{(\alpha,r)} P'$ and $d_2 \vDash P \xrightarrow{(\alpha,r)} P'$. If $\alpha \notin L$ we can assume without loss of generality that $\alpha \in A(P_1)$ and $\alpha \notin A(P_2)$. In fact if $\alpha \in A(P_1)$ and $\alpha \in A(P_2)$, because activities of the same action type are synchronized, it must be the case that $\alpha \in L$. Besides if $\alpha \notin A(P_1)$ and $\alpha \notin A(P_2)$, transition $P \xrightarrow{(\alpha,r)} P'$ cannot occurs. Then we can apply only a rule for cooperation

$$\frac{P_1 \xrightarrow{(\alpha,r)} P_1'}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{(\alpha,r)} P_1' \underset{L}{\bowtie} P_2} \ (\alpha \notin L)$$

and because $Q(P_1)$ is true by inductive hypothesis, we conclude that $d_1 = d_2$ and $Q(P)$ is true.

If $\alpha \in L$, there is just one cooperation rule we can apply.

$$\frac{P_1 \xrightarrow{(\alpha,r_1)} P_1' \quad P_2 \xrightarrow{(\alpha,r_2)} P_2'}{P_1 \underset{L}{\bowtie} P_2 \xrightarrow{(\alpha,r_1 r_2)} P_1' \underset{L}{\bowtie} P_2'} \ (\alpha \in L)$$

Because $Q(P_1)$ and $Q(P_2)$ are true by inductive hypothesis, we conclude that $d_1 = d_2$ and also $Q(P)$ is true.

□

We denote with $X_N(t)$ a Markov chain corresponding to a PEPA model C_N with $N+1$ levels of concentration. State $X_N(t)$ represents the number of levels of each molecule species at time t and $HX_N(t)$ is the discrete concentration vector. Let w_N be the initial state vector, namely $w_N = \omega(C_N)$. State space E_N is equal to $\rho(w_N) \subset \mathbb{N}^{|S|}$.

Transitions of the Markov chain correspond to arcs in the derivation graph. By definition if $< C, C', a > \in A$, there exists a derivation $C \xrightarrow{a} C'$ where $a = (\alpha, r)$. In our case $C \equiv m_1[l_1] \underset{\_}{\bowtie} \ldots \underset{\_}{\bowtie} m_{|M|}[l_{|M|}]$ and $C' \equiv m_1[l_1'] \underset{\_}{\bowtie} \ldots \underset{\_}{\bowtie} m_{|M|}[l_{|M|}']$. Thus if a derivation $C \xrightarrow{a} C'$ exists, the following conditions hold:

i. if $(m_i, \alpha) \in E$, then $l_i > 0$, in fact suppose *ad absurdum* that $l_i = 0$, then $\alpha \notin A(m_i[0])$, hence, because activities of the same action type are synchronized, $C \xrightarrow{a} C'$ cannot occur;

ii. if $(\alpha, m_i) \in E$ and $(m_i, \alpha) \notin E$, then $l_i < N$; we can use an argument similar to that one above.

Now we are able to describe the rates of the Markov chain. If there exists an activity $a = (\alpha, r)$ such that $< C, C', a > \in A$ then $q_{\omega(C),\omega(C')} = r = v_\alpha H^{-1} \prod_{(m_i,\alpha)\in E} l_i H$, otherwise $q_{\omega(C),\omega(C')} = 0$.

On the other hand if (i) and (ii) are satisfied, then there exists a derivation $C \xrightarrow{a} C'$ and thus $< C, C', a > \in A$. These facts lead to the following proposition that allows us to describe the structure of the states of a Markov chain given the derivation graph.

Proposition 5.4.4. *$< C, C', a > \in A$ if and only if (i) and (ii) are satisfied.*

When an action/reaction α is undertaken, molecule levels can only increase or decrease of one unit or remain the same. A stoichiometric vector λ_α records this information. This vector has size $|M|$ and entry $\lambda_{\alpha,i}$ is the change in the population of m_i caused by reaction α. More formally, $\lambda_{\alpha,i}$ is defined as 1 if $(m_i, \alpha) \notin E$ and $(\alpha, m_i) \in E$, -1 if $(m_i, \alpha) \in E$ and $(\alpha, m_i) \in E$, 0 otherwise.

It could be useful to look at a Markov chain from the view point of reactions rather than that one of derivation graph. For every reaction α we denote as products the set $prod(\alpha)$ of molecules m such that $(\alpha, m) \in E$ and $(m, \alpha) \notin E$ and as reagents the set $reag(\alpha)$ of molecules m such that $(m, \alpha) \in E$. Because of Proposition 5.4.4 and because, if a reagent has level zero, the product of reagents is also zero, we obtain that for every reaction $\alpha \in R$ and state $x \in E_N$,

$$q_{x,x+\lambda_\alpha} = \begin{cases} \frac{N}{M} v_\alpha \prod_{m_i \in reag(\alpha)} \frac{M}{N} x_i & \text{if } \bigwedge_{m_i \in prod(\alpha)} x_i < N \\ 0 & \text{otherwise} \end{cases}$$

All other transitions are zero. Since levels cannot be greater than N, Markov chains of this kind are finite.

5.5 Some Examples

In this section we study the behaviour of some simple Markov chains $X_N(t)$ for increasing values of N. We derive the corresponding time dependent solutions analytically in order to understand the relatioships between stochastic and deterministic models.

In the first case we consider a decay model. We show how the average behaviour of the Markov chain corresponds to the solution of its deterministic version for every value of N. In the second case we consider a model for exponential growth. In our framework deterministic and stochastic versions for growth model are different because in the stochastic model we assume there exists a maximal concentration. Nevertheless we observe that changing slightly the definition of our approach, for example making parameter H constant and independent from N, the stochastic model converges

towards the deterministic one for $N \to \infty$.

The important aspect of this section is that in these two particular cases the behaviours of stochastic and deterministic models are the same under some conditions. In next sections we analyse this fact for a more general case.

5.5.1 Decay model

We consider a simple model which describes the degradation process of a kind of molecule or substance A. The deterministic behaviour is represented by the following differential equations,

$$\begin{aligned} \frac{da(t)}{dt} &= -\mu a(t) \\ a(0) &= A_0 \end{aligned}$$

It is easy to verify that the system solution is given by,

$$a(t) = A_0 e^{-\mu t}$$

We divide interval $[0, M]$ into $N+1$ discrete levels $l_0, \ldots, l_N$ where $l_{i+1} - l_i = R$ for $i = 0, \ldots, N-1$. We define $A_0 = M$ as obvious and $H = \frac{M}{N}$. The model corresponds to a continuous time Markov chain of kind "pure death" with degradation rates $\mu_i = \frac{\mu}{H} iH = \mu i$. State $A = i$ stands for discrete level $[iH, (i+1)H)$.

We assign reward iH to state A_i, namely to concentration level $[iH, (i+1)H)$. In Figure 5.1 we compare the expected reward with the exact solution of the differential equation. In this simple case we have a good approximation also for few levels and the graphs are indistinguishable.

In Figure 5.2 we report relative and absolute errors for several levels. We observe that the more levels there are, the larger the absolute error is. This is counter intuitive because you may imagine that, if there are more discrete levels, the model should be close to the continuous case. On the contrary the relative error is smaller with more levels.

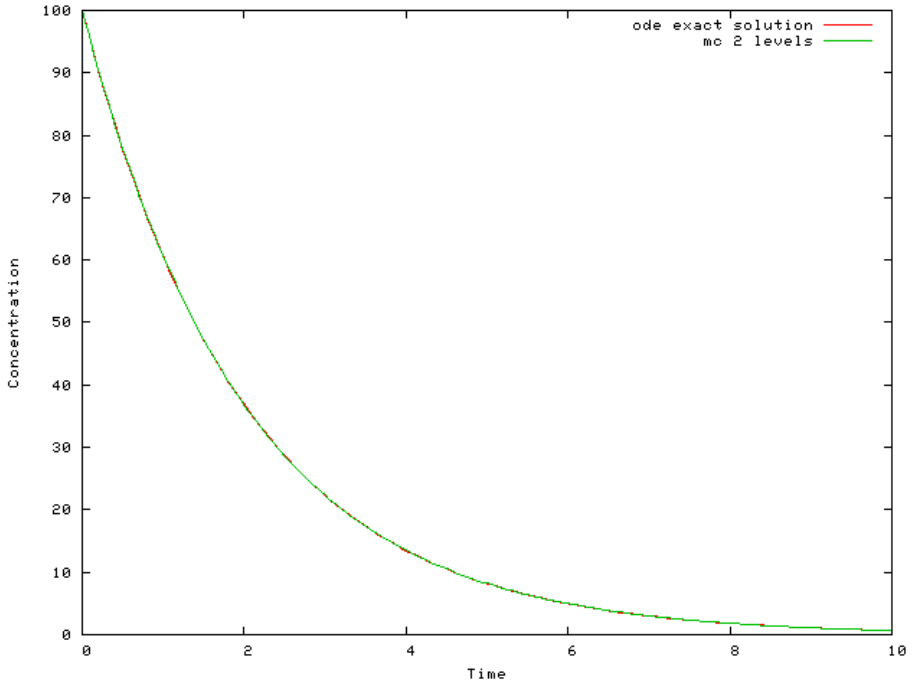


Figure 5.1: **Numerical solutions for deterministic and stochastic decay models.** Experiments show that deterministic and stochastic solutions are very similar.

levels	absolute error	relative error
2	6.44580946487e-06	0.0222091587326
3	6.44580946487e-06	0.0222091587326
4	2.96504156795e-05	0.00246082693734
5	7.87910283347e-05	0.000129234547456

Figure 5.2: **Absolute and relative errors with different number of levels between stochastic and deterministic decay models.** We took 2000 sample points in interval $[0, \ldots, 100]$. In this simple example also few discrete levels are a good approximation for the continuous problem, however, as the number of levels increases, the absolute error grows up.

We are going to explain why this happens. Firstly, we will work out the time dependent solution of the Markov chain. The only transition from state i to state $i - 1$ has rate μi and state 0 is an absorbing state. Hence the transient behaviour of the system is given by the following differential equations,

$$\begin{cases} \dfrac{d\pi_N(t)}{dt} &= -N\mu\pi_N(t) \\ \dfrac{d\pi_i(t)}{dt} &= (i+1)\mu\pi_{i+1}(t) - i\mu\pi_i(t) \quad i = 0, \ldots, N-1 \\ \pi_i(0) &= 0 \quad i = 0, \ldots, N-1 \\ \pi_N(0) &= 1 \end{cases} \tag{5.1}$$

We observe that

$$\frac{d(e^{i\mu t}\pi_i(t))}{dt} = (i+1)\mu\pi_{i+1}(t)e^{i\mu t}$$

Therefore,

$$\pi_i(t) = (i+1)e^{-i\mu t}\mu \int_0^t \pi_{i+1}(\bar{t})e^{i\mu\bar{t}}\, d\bar{t} \tag{5.2}$$

Hence, because $\pi_N(t) = e^{-N\mu t}$ and because of Equation 5.2,we obtain recursively,

$$\pi_i(t) = \binom{N}{i}(e^{-\mu t})^i(1-e^{\mu t})^{N-i} \quad i = 0, \ldots, N \tag{5.3}$$

The absolute error at time t is given by

$$\epsilon_\alpha(t) = \left| a(t) - \sum_{i=1}^{N} i\frac{A_0}{N}\pi_i(t) \right| \tag{5.4}$$

where $a(t)$ is the exact solution at time t for the deterministic model and $\sum_{i=0}^{N} i\frac{A_0}{N}\pi_i(t)$ is the avarage concentration at time t for the Markov chain.

$$
\begin{aligned}
\epsilon_\alpha(t) &= \left| a(t) - \sum_{i=0}^{N} i\frac{A_0}{N}\pi_i(t) \right| \\
&= \left| A_0 e^{-\mu t} - \sum_{i=1}^{N} i\frac{A_0}{N}\binom{N}{i}(e^{-\mu t})^i(1-e^{\mu t})^{N-i} \right| \\
&= \left| A_0 e^{-\mu t} - \frac{A_0}{N}\sum_{i=1}^{N} i\binom{N}{i}(e^{-\mu t})^i(1-e^{\mu t})^{N-i} \right| \\
&= \left| A_0 e^{-\mu t} - \frac{A_0}{N}\sum_{i=1}^{N} i\frac{N(N-1)!}{i(i-1)!(N-i)!}(e^{-\mu t})^i(1-e^{\mu t})^{N-i} \right| \\
&= \left| A_0 e^{-\mu t} - A_0\sum_{i=1}^{N} \binom{N-1}{i-1}(e^{-\mu t})^i(1-e^{\mu t})^{N-i} \right| \\
&= \left| A_0 e^{-\mu t} - A_0 e^{-\mu t}\sum_{i=1}^{N} \binom{N-1}{i-1}(e^{-\mu t})^{i-1}(1-e^{\mu t})^{(N-1)-(i-1)} \right| \\
&= \left| A_0 e^{-\mu t} - A_0 e^{-\mu t}\sum_{j=0}^{N-1} \binom{N-1}{j}(e^{-\mu t})^{j}(1-e^{\mu t})^{(N-1)-j} \right| \\
&= \left| A_0 e^{-\mu t} - A_0 e^{-\mu t}(e^{-\mu t}+1-e^{\mu t})^{N-1} \right| \\
&= \left| A_0 e^{-\mu t} - A_0 e^{-\mu t} \right| = 0
\end{aligned}
$$

Thus, for this particular case, the discrete level approach is **equivalent** to the deterministic model for every number of levels[1]. The absolute errors in Figure 5.2 are non zero because of error propagation in the numerical methods used to compute the transitory probabilities. In general the larger the size of the Markov chain, the greater the propagation error is.

5.5.2 Growing model

We consider a model which represents the growth of a population of molecules of kind *A*. The differential equation is given by

[1] Another way to prove it: $\sum_{i=0}^{N} i\pi_i(t)$ is the mean of a binomial distribution with parameter $e^{-\mu t}$, namely $Ne^{-\mu t}$

$$\begin{aligned} \frac{da(t)}{dt} &= \lambda a(t) \\ a(0) &= A_0 > 0 \end{aligned}$$

The only solution is

$$a(t) = A_0 e^{\lambda t}$$

We choose $H = A_0$ because it is easier to find out a closed form time dependent solution for the underlying Markov chain. We divide concentration interval $[0, NA_0]$ into $N + 1$ discrete levels $l_0, \ldots, l_N$ where $l_{i+1} - l_i = A_0$ for $i = 1, \ldots N - 1$. Level l_i corresponds to interval $[iA_0, (i + 1)A_0)$ and has reward iA_0. The model represents a "pure birth" Markov chain with rate $\lambda_i = \frac{\lambda}{A_0} iA_0 = \lambda i$.

In Figure 5.3 we compare the exact ODE solution with the expected reward of the Markov chain for increasing number of levels. The deterministic solution does not have a maximal concentration value as assumed in the Markovian model; hence the Markovian and the deterministic approaches yield different results. However the greater the number of levels is, the better the approximation seems to be.

We are interested in solving the Kolmogorov equations in order to understand the behaviour observed experimentally. The transitory behaviour of the Markov chain is described by the following differential equations,

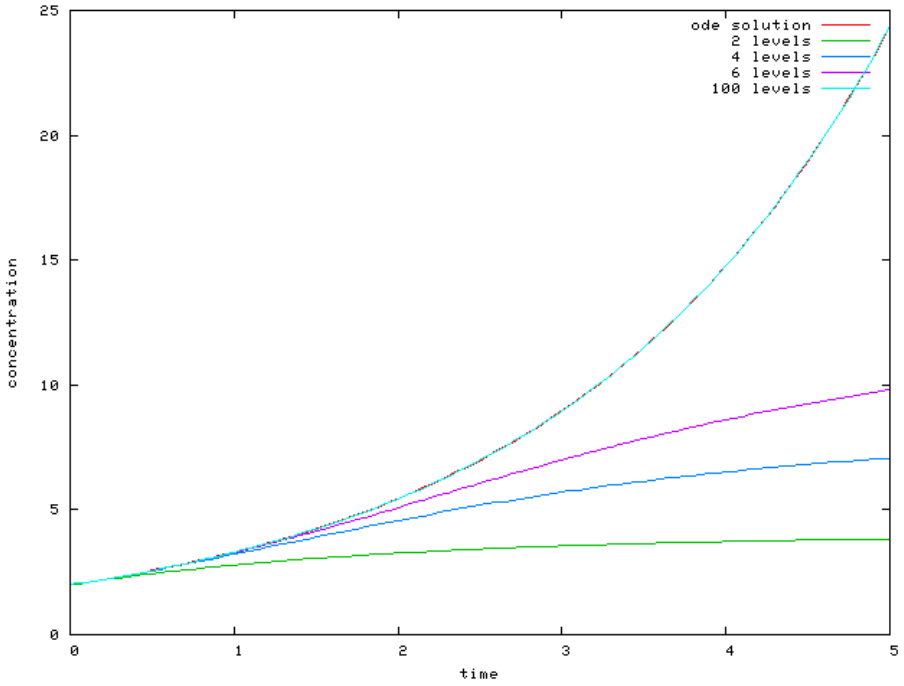


Figure 5.3: Deterministic and Markovian analysis for the simple growing model.

$$\begin{cases} \dfrac{d\pi_1(t)}{dt} &= -\lambda\pi_1(t) \\ \dfrac{d\pi_i(t)}{dt} &= -i\lambda\pi_i(t) + (i-1)\lambda\pi_{i-1}(t) \quad i = 2,\ldots,N-1 \\ \dfrac{d\pi_N(t)}{dt} &= (N-1)\lambda\pi_{N-1}(t) \\ \pi_i(0) &= 0 \quad i = 2,\ldots,N \\ \pi_1(0) &= 1 \end{cases} \tag{5.5}$$

We claim that $\pi_i(t) = e^{-\lambda t}(1-e^{-\lambda t})^{i-1}$ for any $i = 1,\ldots,N-1$ is a solution for the system (5.5). We prove it by recursion on the number of state i. The base case is trivial; in fact from the first and last equations we obtain $\pi_1(t) = e^{-\lambda t}$. We assume $\pi_i(t) = e^{-\lambda t}(1-e^{-\lambda t})^{i-1}$ is true for $i > 1$ and we want to prove the same property for $i+1$. Firstly, we observe that, for any $i = 1,\ldots,N-1$, π_i can

be written as

$$\pi_i(t) = \int_0^t (i-1)\lambda\pi_{i-1}(s)e^{-i\lambda(t-s)}\,ds$$

Therefore

$$\begin{aligned}
\pi_{i+1}(t) &= \int_0^t i\lambda\pi_i(s)e^{-(i+1)\lambda(t-s)}\,ds \\
&= \int_0^t i\lambda e^{-\lambda s}(1-e^{\lambda s})^{i-1}e^{-(i+1)\lambda(t-s)}\,ds \\
&= e^{-(i+1)\lambda t}\lambda\int_0^t ie^{i\lambda s}(1-e^{\lambda s})^{i-1}\,ds \\
&= e^{-(i+1)\lambda t}\lambda\int_0^t ie^{\lambda s}(e^{\lambda s}-1)^{i-1}\,ds \\
&= e^{-(i+1)\lambda t}\int_1^{e^{\lambda t}} i(x-1)^{i-1}\,dx \\
&= e^{-(i+1)\lambda t}\int_0^{e^{\lambda t}-1} ix^{i-1}\,dx \\
&= e^{-(i+1)\lambda t}(1-e^{-\lambda t})^i = e^{-\lambda t}(1-e^{-\lambda t})^i
\end{aligned}$$

Finally, because $\frac{d\pi_N(t)}{dt} = (N-1)\lambda\pi_{N-1}(t)$ and $\pi_{N-1}(t) = e^{-\lambda t}(1-e^{-\lambda t})^{N-2}$, we obtain $\pi_N(t) = (1-e^{-\lambda t})^{N-1}$.

The avarage concentration value in the Markov chain is defined as

$$\begin{aligned}
E[A(t)] &= A_0e^{-\lambda t} + \sum_{i=2}^{N-1}(iA_0e^{-\lambda t}(1-e^{-\lambda t})^{i-1}) + NA_0(1-e^{-\lambda t})^{N-1} \\
&= A_0e^{-\lambda t} + A_0(2(1-e^{-\lambda t}) + \frac{(1-e^{-\lambda t})^2 - (1-e^{-\lambda t})^{N-1}}{e^{-\lambda t}} \\
&\quad -(N-1)(1-e^{-\lambda t})^{N-1}) + NA_0(1-e^{-\lambda t})^{N-1} \\
&= A_0e^{\lambda t}(1-(1-e^{-\lambda t})^N) = A_0e^{\lambda t} - A_0e^{\lambda t}(1-e^{-\lambda t})^N
\end{aligned}$$

The absolute error is given by

$$\epsilon_\alpha(t,N) \quad = \quad |a(t) - E[A(t)]| = A_0 e^{\lambda t}(1-e^{-\lambda t})^N$$

Hence $\epsilon_\alpha(t,N) = 0$ only if $t = 0$. As observed from numerical simulations, for any fixed time t, $\lim_{N\to\infty} \epsilon_\alpha(t,N) = 0$. Thus $\lim_{N\to\infty} E[A(t)] = a(t)$ or, in losing words, if the Markov chain is infinite, the deterministic simulation is equivalent to the avarage behaviour of the Markov chain. For any fixed number of level N we study the behaviour of $E[A(t)]$ in time. The function $E[A(t)]$ is always positive non-decreasing and $E[A(0)] = A_0$. Moreover for $t \to \infty$ $E[A(t)]$ tends to the maximal concentration value $A_0 N$, we prove it using L'Hopital's Rule,

$$\begin{aligned}
\lim_{t\to\infty} E[A(t)] &= \lim_{t\to\infty} A_0 \frac{1-(1-e^{-\lambda t})^N}{e^{-\lambda t}} \\
&= A_0 \lim_{t\to\infty} N(1-e^{-\lambda t})^{N-1} \\
&= A_0 N
\end{aligned}$$

5.6 Limit distribution of Markov chains with discrete levels of concentration

5.6.1 Kurtz's theorem

In this section we introduce the main results of Kurtz's theorem; for a more technical and formal presentation readers can look through Kurtz (1970, 1971).

A *density dependent family* of Markov chains X_v is a sequence $\{X_v\}$ of Markov processes such that v is positive, the state space of X_v is $E_v \subset \mathbb{Z}^m$ and the transition rates are given by

$$q_{x,x+l} = v f\left(\frac{1}{v}x, l\right) \quad l \neq 0$$

where $f(x,l)$ with $x \in \mathbb{R}^m$ and $l \in \mathbb{Z}^m$ are continuous functions.

Roughly speaking such a family is defined by a parameter v which represents volume, population size or whatever else. The states of the Markov chains are normalized with respect to v and the state spaces of this sort of dentities are considered instead. The transition rates depends on the densities, hence the name density dependent Markov chains.

Theorem 5.6.1 (Kurtz (1970)). *Define a function F in the following way*

$$F(x) = \sum_l l f(x,l)$$

Let $E \subset \mathbb{R}^m$ be an open set and M_E a constant such that

i. $|F(x) - F(y)| < M_E|x-y|$ *for any* $x, y \in E$,

ii. $sup_{x\in E} \sum_l |l| f(x,l) < \infty$ *and*

iii. $\lim_{d\to\infty} sup_{x\in E} \sum_{|l|>d} |l| f(x,l) = 0$.

Assume $X(s)$ is a solution of the ordinary differential equations

$$\begin{aligned} \frac{\partial X(s)}{\partial s} &= F(X(s)) \\ X(0) &= x_0 \end{aligned}$$

where $X(s) \in E$ for $0 \le s \le t$ and $\lim_{v\to\infty} v^{-1}X_v(0) = x_0$, then for every $\delta > 0$

$$\lim_{v\to\infty} P\left\{ sup_{s\le t} \left| \frac{1}{v} X_v(s) - X(s) \right| > \delta \right\} = 0$$

As parameter v grows arbitrarily large, the sequence of stochastic processes $v^{-1}X_v(t)$ converges to a deterministic process $X(t)$ which is solution of the ordinary differential equations defined above. While systems with finite v are discrete processes, the limiting system is continuous.

Kurtz's theorem has been used in a lot of chemical and biological applications to clear the

relationship between the stochastic and deterministic models where the state spaces represent populations of individuals or molecules and v is the volume or area of the region in which the populations live.

We want to study the behaviour of Markov chains with discrete levels of concentration for increasing values of N. We consider a sequence of Markov chains defined as in Section 5.4,

$$X_k(t), X_{k+1}(t), \ldots, X_N(t), \ldots$$

We want to study the convergence of this sequence for $N \to \infty$ via Kurtz's theorem. Our case is not similar to the other biological applications which use Kurtz's results. We have Markov chains $X_N(t)$ whose structure depends on parameter N, namely the number of levels. However the stochastic process $N^{-1}X_N(t)$ does not represent a concentration or a population density, but a normalized or scaled level. For instance event $N^{-1}X_{iN}(t) = 1$ means the level of molecule m_i is the highest possibile and the actual level value depends on the scale factor N. Transitions depend on "densities", but in general $X_N(t)$ is not density dependent because transition rates cannot be rewritten in terms of continuous functions. For example, if a product of a reaction has level N, the reaction cannot be undertaken although all reagents have levels greater than zero. Despite of that there exist some particular cases of Markov chains with discrete levels that are also density dependent. In next sections we will explore them.

5.6.2 Reaction networks

We want to define a property of a PEPA model which allows us to infer that the sequence of Markov chains extracted from the model for increasing values of N is density dependent. Instead of a PEPA model we consider the corresponding activity graph. Besides we extend the notion of activity graph with some information on the levels of each molecule species after some reactions are carried out. We call this new object reaction network. More formally we give the following definition.

Definition 5.6.2. *An activity graph is a directed bipartite graph $G = (V, E)$. Node set V is equal to $R \cup S$ where R is the set of action types (i.e. reactions) and S is the set of process names (i.e. molecules). Edge set $E \subseteq (R \times S) \cup (S \times R)$ is the smallest set such that:*

- *if molecule m is consumed in reaction a, then $(m, a) \in E$;*
- *if molecule m is produced in reaction a, then $(a, m) \in E$;*
- *if molecule m is involved in reaction a but it is not consumed, then $(m, a) \in E$ and $(a, m) \in E$.*

The graph is represented by an activity matrix $A = A^+ - A^-$ of size $|S| \times |R|$. Here $A^+ = \{a^+_{ij}\}$ where $a^+_{ij} = 1$ if $(j, i) \in E$, else $a^+_{ij} = 0$. Instead $A^- = \{a^-_{ij}\}$ where $a^-_{ij} = 1$ if $(i, j) \in E$, else $a^-_{ij} = 0$.

Every node m_i has a weight $w_i \in \mathbb{N}$ which models the level number of molecule m_i. The vector $w_N \in \mathbb{N}^{|S|}$ is the row vector of initial levels. Given a weight assignment w, a sequence of reactions $r = r_{i_1}, \ldots r_{i_m}$ is a possible succession of reactions. Not all sequences r are possible; in fact when the level of some reagents is zero, the corresponding reactions are not allowed. A sequence of reaction can be empty, in this case we denote r with λ. Given a weight assignment w and a possible reaction r_j the next state w' of the system is given by

$$w' \quad = \quad w + A\epsilon_j$$

where ϵ_j is the j-th column of the identity matrix of size $|R| \times |R|$.

Given an assignment w and a possible sequence of reactions r, $\bar{r} \in \mathbb{N}^{|R|}$ is a column vector which counts the frequencies of each reaction in r; i.e. $\bar{r}_i$ is the occurences of reaction R_i in r. Hence, $w^r = w + A\bar{r}$ is the vector of levels after all the reactions in r are carried out.

A reaction network is closed if arrivals and departures into and out of the system are not allowed. A network with departures can be mimicked with a closed network where a special kind of molecule represents lost molecules; we will use this observation to model degradation processes. Instead, in order to represent arrivals, we can define a special kind for outside molecules with infinite weight.

In the rest of this work we will consider just closed network with finite weights. We observe that reaction networks are a particular case of Petri Net.

We introduce the vector notation we use in this work. The i-th entry of a vector w is denoted by w_i. Given two vector w and w' $w < w'$ if and only if $w_i < w'_i$ for every i; relations $\leq$, $\geq$, $>$ and $=$ are defined in a similar way. Given a vector w and a scalar k $w < k$ means that every entry w_i of w are less than k; the same for the other relations.

5.6.3 $k/0$ networks

Definition 5.6.3. *Let R be a set of reactions, S a set of molecules and w a starting weight assignment. A reaction network $G = (S \cup R, E)$ is $k/0$ if and only if for every possible sequence of reactions r and for every reaction $\alpha \in R$ the following condition holds,*

$$\bigvee_{m_i \in prod(\alpha)} w_i^r \geq k \Rightarrow \bigvee_{m_i \in reag(\alpha)} w_i^r = 0 \tag{5.6}$$

In a $k/0$ network if some products of a reaction have a level equal to or greater than k at a given point of the evolution of the system, then there exists at least one reagent of the reaction with level zero. As a consequence, the reaction cannot occur. Therefore k represents a bound for the possible level values.

If a molecule has a starting concentration level higher than the maximal level, we may assume the concentration has been rised artificially. Nevertheless, we will consider $w_i \leq k$.

5.6.4 Bounded networks

In this section we explore the idea of $k/0$ networks as bounded networks.

Definition 5.6.4. *Assume $G = (V, E)$ with $V = S \cup R$ is a reaction network and $w \in \mathbb{N}^{|S|}$ an assignment. A node $m_i \in S$ is k-bounded if and only if for every possible sequence r of reactions, $w_i^r \leq k$. If every node $m_i \in S$ is k-bounded, then network G is k-bounded and vice versa. A network G is bounded if and only if there exists k such that G is k-bounded.*

Lemma 5.6.5. *Given an assignment $w \leq k$, a reaction network G is k-bounded if and only if G is $k/0$.*

Proof.

$\Rightarrow$ If G is k-bounded, then every node $m_i \in S$ is k-bounded, specifically for any possible r, $w_i^r \leq k$. Consider a reaction α where $\bigvee_{m_i \in prod(\alpha)} w_k^r \geq k$. We assume *ad absurdum* α is enabled, i.e. $\bigwedge_{m_i \in reag(\alpha)} w_i^r \neq 0$. Without loss of generality, let $m_i \in prod(\alpha)$ such that $w_i^r \geq k$. The reaction sequence $r' = r\alpha$ is still valid because of our assumption and we get $w_i^{r'} > k$ which contradicts the hypothesis of boundedness. Thus, we conclude $\bigvee_{m_i \in reag(\alpha)} w_i^r = 0$. Hence, G is $k/0$.

$\Leftarrow$ Let k be a number such that G is $k/0$. Suppose *ad absurdum* there exists r such that $w^r > k$ and assume $w_1^r > k$ without loss of generality. Since $w \leq k$ and levels can increase of a unit, there exists a prefix $\bar{r}$ of r such that $w_1^{\bar{r}} = k$. Because $w_1^{\bar{r}} = k$, for every reaction α such that $m_1 \in prod(\alpha)$ there exists $m_j \in reag(\alpha)$ such that $w_1^{\bar{r}} = 0$. Hence reaction α is not enabled and the level of m_1 cannot be increased. This contradicts our hypothesis and so $w^r \leq k$ for any r. Therefore G is k-bounded.

□

5.6.5 Application of Kurtz's theorem

The following lemma illustrates the relation between the states of a Markov chain $X_N(t)$ and those of a reaction network $G = (S \cup R, E)$ corresponding to the same activity graph.

Lemma 5.6.6. *For every $x \in \rho(w_N)$, there exists a valid sequence of reactions r such that $x = w_N^r$.*

Proof. Note that $x \in \rho(w_N)$ if and only if $\omega(x) \in ds(\omega(w_N))$ if and only if there exists a sequence of transitions $\omega(w_N) \xrightarrow{(\alpha_1, r_1)} \dots \xrightarrow{(\alpha_{k-1}, r_{k_1})} \omega(x)$. We want to prove by induction on the length of the transition chain that there exists a valid r such that $x = w_N^r$.

- If $x = w_N$, then we take $r = \lambda$ and the conclusion follows immediately.

- We assume there exists a sequence of transitions $\omega(w_N) \xrightarrow{(\alpha_1,r_1)} \dots \xrightarrow{(\alpha_{k-1},r_{k_1})} \omega(x)$ such that r is valid and $x = w_N^r = w_N + A\bar{r}$. Let (α_k, r_k) be an activity such that $\omega(x) \xrightarrow{(\alpha_k,r_k)} \omega(x')$. We need to prove that $r\alpha_k$ is still a valid sequence and that $x' = w_N^{r\alpha_k}$. Assume *ad absurdum* that α_k is not possible, then there exists $m_i \in reag(\alpha_k)$ such that $x_i = 0$. Thus $\alpha_k \notin A(m_i[0])$ and, because activities of the same action type are synchronized, $\omega(x) \xrightarrow{(\alpha_k,r_k)} \omega(x')$ cannot occur. Assume j is the index corresponding to reaction α_k in A; ϵ_j is the j-th column of identity matrix. It is trivial to realize that

$$\overline{r\alpha_k} = \bar{r} + \epsilon_j$$

Thus, given that a_j denotes the j-th column of A, we have

$$w_N^{r\alpha_k} = w_N + A\overline{r\alpha_k} = w_N + A\bar{r} + a_j^T = x + a_j^T$$

By definition of A it follows,

$$w_{Ni}^{r\alpha_k} = \begin{cases} x_i + 1 & \text{if } (\alpha_k, m_i) \in E \wedge (m_i, \alpha_k) \notin E \\ x_i - 1 & \text{if } (\alpha_k, m_i) \notin E \wedge (m_i, \alpha_k) \in E \\ x_i & \text{else} \end{cases}$$

We conclude that $x' = w_N^{r\alpha_k}$.

□

We note that the contrary is not always true. In fact a network can be bounded whereas the state space of a Markov chain with levels is always finite.

The following lemma states the relation between a Markov chain with levels and the corresponding reaction network. We show the equivalence between the boundedness of the network and the property that the Markov chain has density dependent rates.

Lemma 5.6.7. *Given $w_n \leq N$, $X_N(t)$ is a Markov chain with discrete concentration levels with initial state vector w_N; G is a reaction network. Then, G is $N/0$ for w_N if and only if for every reaction α in R and any state $x \in E_N$, rate $q_{x,x+\lambda_a}$ is equal to $q_{x,x+\lambda_\alpha} = \frac{N}{M} v_\alpha \prod_{m_i \in reag(\alpha)} \frac{M}{N} x_i$.*

Proof.

$\Rightarrow$ Assume G is $N/0$ for w_N. By definition, given a state x of $X_N(t)$ and any reaction α,

$$q_{x,x+\lambda_\alpha} = \begin{cases} \frac{N}{M} v_\alpha \prod_{m_i \in reag(\alpha)} \frac{M}{N} x_i & \text{if } \bigwedge_{m_i \in prod(\alpha)} x_i < N \\ 0 & \text{otherwise} \end{cases}$$

If $\bigwedge_{m_i \in prod(\alpha)} x_i < N$, the conclusion follows immediately. Otherwise, since $x \in \rho(w_N)$ is reachable from w_N, $x = w_N^r$ for some reaction sequence r (see Lemma 5.6.6). Therefore, because G is $N/0$ for w_N, if $\bigvee_{m_i \in prod(\alpha)} x_i \geq N$, then $\bigvee_{m_i \in reag(\alpha)} x_i = 0$ and thus $q_{x,x+\lambda_\alpha} = 0 = \frac{N}{M} v_\alpha \prod_{m_i \in reag(\alpha)} \frac{M}{N} x_i$.

$\Leftarrow$ For every $x \in \rho(w_N)$ the following proposition Q is verified

$$\forall \alpha \in R \bigvee_{m_i \in prod(\alpha)} x_i \geq N \Rightarrow \bigvee_{m_i \in reag(\alpha)} x_i = 0$$

In fact because $q_{x,x+\lambda_\alpha} = \frac{N}{M} v_\alpha \prod_{m_i \in reag(\alpha)} \frac{M}{N} x_i$ by hypothesis, if $\bigvee_{m_i \in prod(\alpha)} x_i \geq N$, $q_{x,x+\lambda_\alpha}$ must be equal to zero by definition. Hence, $\bigvee_{m_i \in reag(\alpha)} x_i = 0$. We want to prove that, given $w_N \leq N$, for every valid sequence of reactions r, $w_N^r \in \rho(w_N)$. Thus $Q(w_N^r)$ is true and G is $N/0$. We prove this fact by induction on the length of the sequence r.

- If $r = \lambda$, because $w_N \in \rho(w_N)$ by definition, the conclusion follows immediately.
- Assume that $w_N^r \in \rho(w_N)$ for r, namely there exists a sequence of derivations $\omega(w_N) \xrightarrow{(\alpha_1, r_1)} \ldots \xrightarrow{(\alpha_{k-1}, r_{k_1})} \omega(w_N^r)$. Let α_k be a possible reaction after r, i.e. $\bigwedge_{m_i \in reag(\alpha_k)} w_{Ni}^r > 0$. Then $w_N^{r\alpha_k} \in \rho(w_N)$ if there exists a transition $\omega(w_N^r) \xrightarrow{(\alpha_k, r_k)} \omega(w_N^{r\alpha_k})$. α_k is enabled if

$\bigwedge_{m_i \in reag(alpha_k)} w^r_{Ni} > 0$ and $\bigwedge_{m_i \in prod(\alpha_k)} w^r_{Ni} < N$. The first condition is true because by hypothesis reaction α_k is possible. The second condition is true by induction. In fact if *ad absurdum* $\bigwedge_{m_i \in prod(\alpha_k)} w^r_{Ni} \geq N$, because $Q(w^r_N)$ is true, $\bigvee_{m_i \in reag(\alpha_k)} w^r_{Ni} = 0$ that contradicts the hypothesis that α_k is possible. If $x = w^r_N$ and $\omega(x) \xrightarrow{(\alpha_k, r_k)} \omega(x')$, then $x' = w^{r\alpha_k}_N$. The proof is equal to the last part of the proof of Lemma 5.6.6.

□

We are now able to define some conditions on the structure of the activity graph in order to understand when the underlying sequence of Markov chains is density dependent. The following corollary highlights the fact that the class of Markov chains defined by Lemma 5.6.7 is a class of density dependent Markov chains. It follows from the definition directly.

Corollary 5.6.8. *A reaction network G is N-bounded for $w_N \leq N$ if and only if for any reaction α and state x Markov chain $X_N(t)$ has rate $q_{x,x+\lambda_\alpha} = N f(N^{-1}x, \lambda_\alpha)$ where $f(x, \lambda_\alpha) = \frac{\nu_\alpha}{M} \prod_{m_i \in reag(\alpha)} M x_i$.*

Proof. It follows from Lemma 5.6.7 and Lemma 5.6.5. □

Corollary 5.6.9. *There exists $k \geq 0$ such that for every $N \geq k$ the reaction network G is N-bounded for $w_N \leq N$ if and only if the sequence $\{X_N(t)\}_{N \geq k}$ is density dependent.*

Proof. It follows from Corollary 5.6.8 and from the definition of density dependent Markov chains.

□

We observe that in general G can be structurally bounded, namely bounded for every possible assigment, and at the same time the condition of Corollary 5.6.9 can be not satisfied. In fact a network can be bounded for every initial assignment and in particular for w_N, but not N-bounded. As an example, consider the network in Figure 5.4 where m_1 and m_2 initial concentrations are high. The network is structurally bounded. Molecules m_1 and m_2 have initial level N whereas molecule m_3 has level zero. It is trival to see that, after a sequence of N reactions r_1 and N reactions r_2s, the level of m_3 becomes $2N$. Thus the network is not N-bounded for every values of N.

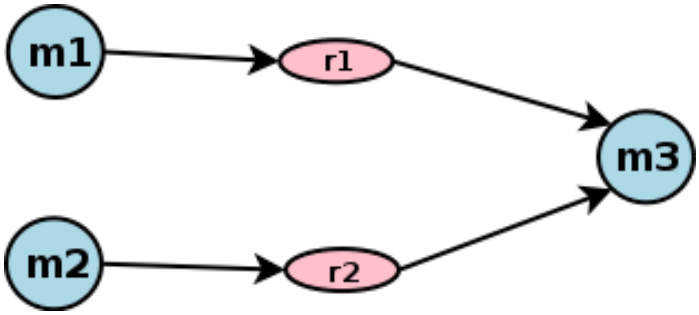


Figure 5.4: **Structural boundedness and density dependency.** This figure shows that a structurally bounded network cannot satisfy conditions for desity dependency of the corresponding Markov chains. Consider an initial assignment where m_1 and m_2 are N whereas m_3 is zero. The network is bounded, but not N-bounded.

The following proposition gives us a necessary condition for boundedness. The idea bihind the proposition is the following. consider a vector u such that $Au \geq 0$ and $Au \neq 0$. Thus, if we have an initial assignment w with values large enough, we can build a valid sequence of reaction r such that $u = \bar{r}$. Therefore r can only increase the level of some molecules, Hence, if we reuse the same sequence many times, the network grows unboudedly.

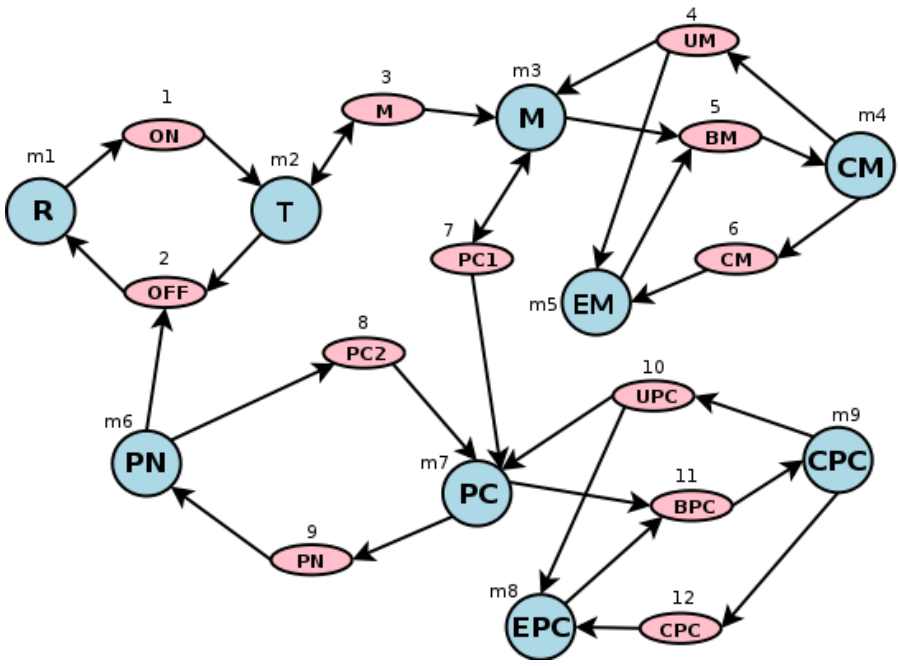
Proposition 5.6.10. *If there exists $u \geq 0$ such that $Au \geq 0$ and $Au \neq 0$, then G is unbounded*

Proof. Consider $u \geq 0$ such that $Au \geq 0$ and $Au \neq 0$. Let $w > \sum_i u_i$ be an initial assignment and r a sequence of reactions that contains u_1 occurrences of reaction r_1, u_2 occurrences of reaction r_2 and so on. Then r is a valid sequence of reactions. In fact suppose *ad absurdum* that r is not valid, then $r = r'\alpha r''$ where r' is a valid sequence of reactions and α is such that $m_i \in reag(\alpha)$ and $w_i^{r'} = 0$ for some i. Because $w_i > \sum_i u_i$, in r' there euist at least $\sum_i u_i + 1$ reactions which decrease the level of m_i. However it is not possible, because in r we have $\sum_i u_i$ reactions.

Therefore $w^r = w + A\bar{r}$. Because $A\bar{r} \geq 0$ and $A\bar{r} \neq 0$, rr is still a valid sequence of reactions and $w^{rr} = w^r + A\bar{r}$. Hence, we can apply r infinitely many times and G is unbounded. □

An example: circadian clocks

We consider the model for circadian clocks presented in Chapter 4. In Figure 5.6.5 there is the corresponding activity graph where reactions and molecules are numbered. From the matrix A



which represents the activity graph a system of linear inqualities $Au \geq 0$ is extracted where some

constraints are different from zero.

$$\begin{aligned}
-u_1 + u_2 &\geq 0 \\
u_1 - u_2 &\geq 0 \\
u_3 + u_4 - u_5 &\geq 0 \\
-u_4 + u_5 - u_6 &\geq 0 \\
u_4 - u_5 + u_6 &\geq 0 \\
-u_2 - u_8 + u_9 &\geq 0 \\
u_7 + u_8 - u_9 + u_{10} - u_{11} &\geq 0 \\
u_{10} - u_{11} + u_{12} &\geq 0 \\
-u_{10} + u_{11} - u_{12} &\geq 0
\end{aligned}$$

There exist many solutions $u > 0$ such that $Au \geq 0$ and $Au \neq 0$. Thus the circadian clock model is not bounded for some initial assignments (Proposition 5.6.10). For example, $u_i = 0$ for $i \neq 3$ and $u_3 > 0$ is a solution. This solution corresponds to a sequence of reactions r_3, i.e. transcription. A reaction r_3 is valid if the level of m_2 is greater than zero. In our case every initial assignment w has w_2 entry greater than zero. Hence, the model can grow unboundly for every w of interest.

Because the model for circadian clocks is not bounded, the underlying sequence of Markov chains is not density dependent and Kurtz's theorem cannot be applied. This does not mean that the stochastic process does not converge to a (deterministic) limit distribution or that there do not exist any relations with the corresponding determinitic model. However, at the moment, we are not able to say anything using Kurtz's theorem.

5.6.6 Some issues on initial assignments

The vector of initial levels w_N depends on the vector of initial concentrations c that is usually known. Level l_i corresponding to concentration c_i is given by the formula N/Mc_i. However N/Mc_i has to

be approximated to some integer value in order to represent a level. Therefore it is not possible to define a general rule to determine which approximation is more appropriate for a given model.

So far we have assumed to know vector w_N. Nevertheless in order to apply results of Corollary 5.6.9 we need to define w_N for every $N \geq k$. Models often describe a scenario where some substances are consumed by reactions. In this case at initial time some molecules species have the greatest concentration, while the others have concentration equal to zero. In other words, given initial concetrations c, for every $k > 0$ the vector w_k is defined as follows,

$$w_{ki} = \begin{cases} k & \text{if } c_i > 0 \\ 0 & \text{if } c_i = 0 \end{cases}$$

One can verify trivially that the following relation between entries of w_k and w_{k+1} is true,

$$w_{k+1i} = \begin{cases} w_{ki} + 1 & \text{if } w_{ki} > 0 \\ w_{ki} & \text{if } w_{ki} = 0 \end{cases}$$

Consider a sequence of initial assignements $\{w_k\}$ defined as above. We hope that the particular structure of these assignments allow us to find a way to semplify the condition of Proposition 5.6.9. In other words we would like to prove the following result.

Proposition 5.6.11. *Let $\{w_k\}$ be a sequence of vectors defined as above and G a reaction network; if G is k-bounded for w_k, then G is also $k+1$-bounded for w_{k+1}.*

The proposition would allow us to prove a corollary similar to Corollary 5.6.9 that is true in this particular case. Thus we have a sufficient condition for density dependency, e.g. when the network is 1-bounded.

Corollary 5.6.12. *Let $\{w_i\}$ be a sequence of vectors defined as above and G a reaction network. If there exists k such that G is k-bounded for w_k, then the sequence $\{X_N(t)\}_{N \geq k}$ is a family of density dependent Markov chains.*

Proof. It follows from Proposition 5.6.11 and from Corollary 5.6.9. □

Unluckly Proposition 5.6.11 is not true. As a counter-example, consider the network in Figure 5.5 where m_1 and m_4 have high initial concentrations. It is easy to verify that the net is 1-bounded for w_1, but it is not 2-bounded for w_2. We belive that, if the net presents some structural properties, then Proposition 5.6.11 is verified. At the moment we are wroking on this problem.

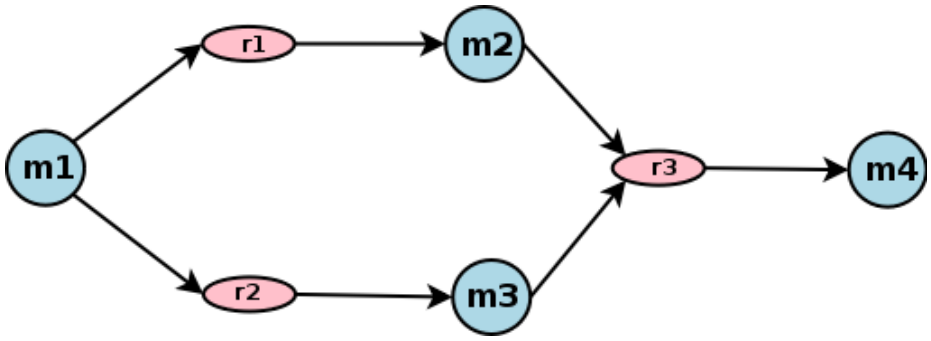


Figure 5.5: **Again on boundedness.** This figure shows that Proposition 5.6.11 is not true. Consider an initial assignment where m_1 and m_4 are N, while m_2 and m_3 are zero. Then the net is 1-bounded for w_1, but it is not 2-bounded for w_2.

5.6.7 Limit distribution and deterministic model

Consider a sequence of Markov chains $\{X_N(t)\}$ defined as above and assume it density dependent. Since function $F(x) = \sum_{\alpha \in R} \lambda_a f(x, \alpha)$ is continuously differentiable, F is locally Lipshitz. Then conditions to apply Kurz's theorem are satisfied for any bounded open set.

Therefore $N^{-1}X_N(t)$ *converges* for $N \to \infty$ to a limit distribution $X(t)$ which is solution of the following differential equations,

$$X(0) = x_0 \tag{5.7}$$

$$\frac{dX(t)}{dt} = F(x) \tag{5.8}$$

Where $F(x) = \sum_{\alpha \in R} \lambda_\alpha f(x, \lambda_\alpha)$ and for every reaction $\alpha \in R$,

$$f(x,\lambda_\alpha) = \frac{v_\alpha}{M} \prod_{m_i \in reag(\alpha)} x_i M$$

We want to find which relation exists between $X(t)$ and the corresponding deterministic model $\overline{X}(t)$, define as follows,

$$\overline{X}(0) = Mx_0 \tag{5.9}$$

$$\frac{d\overline{X}(t)}{dt} = \overline{F}(x) \tag{5.10}$$

Where $\overline{F}(x) = \sum_{\alpha \in R} \lambda_\alpha \overline{f}(x,\lambda_\alpha)$ and for every reaction $\alpha \in R$,

$$\overline{f}(x,\lambda_\alpha) = v_\alpha \prod_{m_i \in reag(\alpha)} x_i$$

Proposition 5.6.13. *If $X(t)$ is a solution of 5.7, then $MX(t)$ is a solution of 5.9.*

Proof. Since $X(0) = x_0$, $\overline{X}(0) = MX(0)$. Consider $t > 0$, we substitute $MX(t)$ instead of $\overline{X}(t)$ in equation $\frac{d\overline{X}(t)}{dt} = \overline{F}(x)$ and we prove the equality holds.

$$\begin{aligned} \frac{d\overline{X}(t)}{dt} &= M\frac{dX(t)}{dt} = \\ &= MF(X(t)) = M\sum_{\alpha \in R} \lambda_\alpha f(X(t),\lambda_\alpha) \end{aligned}$$

Because $Mf(x,\lambda_\alpha) = v_\alpha \prod_{m_i \in reag(\alpha)} x_i M = f(Mx,\lambda_\alpha)$, we conclude $\frac{dMX(t)}{dt} = \overline{F}(MX(t))$. Therefore $\overline{X}(t) = MX(t)$ is solution of 5.9. □

Proposition 5.6.14. *If $\overline{X}(t)$ is a solution of 5.9, then $M^{-1}\overline{X}(t)$ is a solution of 5.7.*

Proof. For $t = 0$ we have $X(0) = 1/M\overline{X}(0) = x_0$. For $t > 0$ we proceed as before and we substitute $1/M\overline{X}(t)$ instead of $X(t)$.

$$\begin{aligned}\frac{dX(t)}{dt} &= \frac{1}{M}\frac{d\overline{X}(t)}{dt} = \\ &= \frac{1}{M}\overline{F}(\overline{X}(t)) = \frac{1}{M}\sum_{a\alpha\in R}\lambda_\alpha \overline{f}(\overline{X}(t),\lambda_\alpha)\end{aligned}$$

Because $\frac{1}{M}\overline{f}(x,\lambda_\alpha) = \frac{v_\alpha}{M}\prod_{m_i\in reag(\alpha)} x_i = f(M^{-1}x,\lambda_\alpha)$, we get $\frac{dM^{-1}\overline{X}(t)}{dt} = F(M^{-1}\overline{X}(t))$. Hence $X(t) = M^{-1}\overline{X}(t)$ is a solution of 5.7.

□

Through Kurtz's theorem for sufficiently large N we have $1/NX_N(t) \simeq X(t)$. Considering concentrations instead of levels we get $M/NX_N(t) \simeq MX(t)$. Because $X(t)$ is a deterministic distribution, $E[X(t)] = X(t)$. Therefore, the average of discrete concentrations $M/NE[X_N(t)]$ converges to $MX(t)$ which is solution of the corresponding deterministic model (Proposition 5.6.13).

5.7 A case study: ERK signalling pathway

The ERK signalling pathway is a biological process involved in cellular division and differentiation. Understanding its dynamics is of interest to cancer research because cell populations grow uncontrollably when the pathway does not work correctly.

Here, we consider a model presented in Cho et al. (2003) that describes how RKIP regulates the behaviour of the ERK pathway. In Figure 5.6 there is a graphical representation of the model.

Here, nodes represent molecules while edges model unbinding and binding reactions between molecules. For example m_1 and m_2 bind together to form a complex m_3 and m_3 splits into molecules m_1 and m_2. Reaction names are given in the rectangles; r_i/r_j denotes binding reaction r_i and unbinding reaction r_j. Initially all concentrations are zero with the exception of the concentrations of molecules m_1, m_2, m_7, m_9 and m_{10}. Each node is labelled with the corresponding protein name.

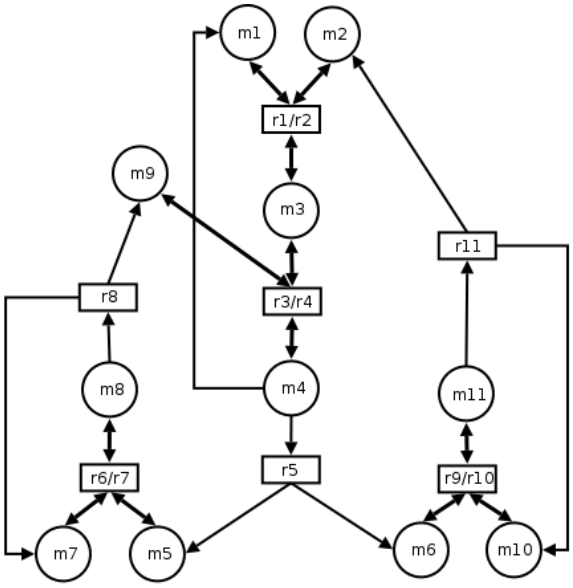


Figure 5.6: ERK signalling pathway regulated by RKIP.

5.7.1 PEPA model and activity graph

We build a PEPA model with two levels of concentration following conventions defined in Chapter 4. The code is similar to the one in Calder et al. (2004) except for the addition of MEK protein and its associated complex. The code can be found in Appendix B. In Figure 5.7 we show the corresponding activity graph.

From the activity graph we derive ODEs (Chapter 4) and a Markov chain with N levels of concentration. For more details readers can have a look at Appendix B.

5.7.2 Convergence to a deterministic distribution

We use results of the previous section in order to study the convergence of the underlying Markov chains for increasing values of N. We want to prove that for a large number of levels the Markovian model behaves as the deterministic interpretation of the PEPA model. This result was observed experimentally (without a formal proof) also in Calder et al. (2005).

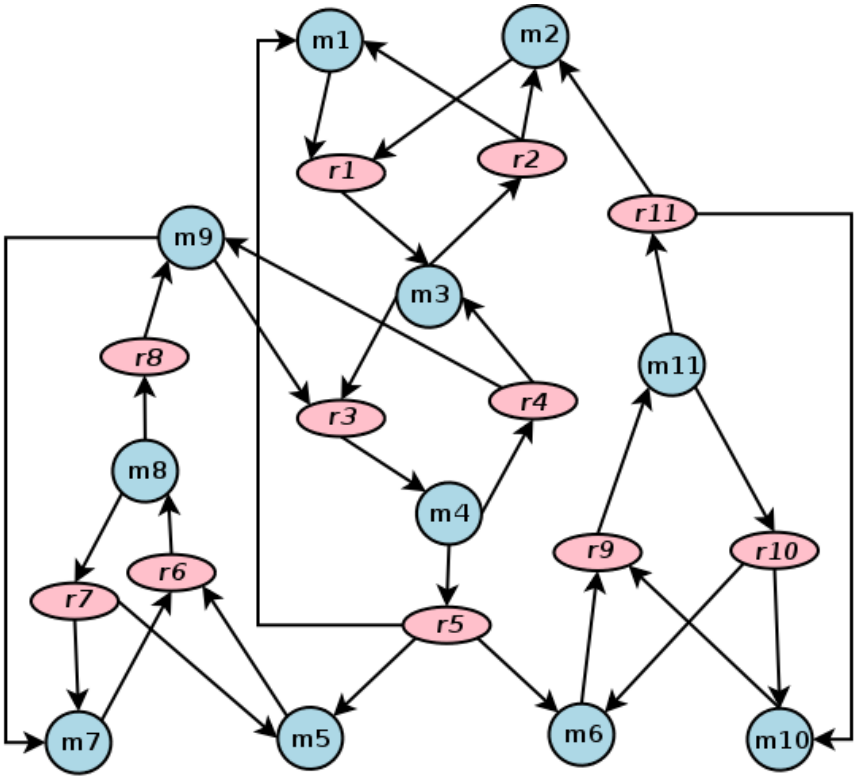


Figure 5.7: **Activity graph for ERK signalling pathway.**

By assumption molecules m_1, m_2, m_7, m_9 and m_{10} have the highest initial level, whereas all others have level zero. Therefore vector w_k has only zero or k entries. If for every $k \geq 1$ the network is k-bounded for w_k, then the underlying Markov chains are density dependent by Corollary 5.6.9. In order to prove k-boundeness we show that for each molecule m_i a state $w(m_i)$ where $w(m_i)_i >= k+1$ and $w(m_i)_j >= 0$ for $j \neq i$ is not reachable from w_k. By definition of a reaction network if there does not exist a vector $u \geq 0$ such that $Au = w - w_k$, then w is not reachable from w_k. Thus we need to solve some systems of integer linear inequalities with symbolic coefficient k.

The matrix corresponding to our model is defined in the following way,

$$A = \begin{pmatrix} -1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 \end{pmatrix}$$

For each molecule m_i we define the corresponding vector $w(m_i) - w_k$ where $w(m_i)_i \geq k + 1$ and $w(m_i)_j \geq 0$ $j \neq i$. We list these vectors in the following table.

molecule	$w(m_i) - w_k$
m_1	$(1, -k, 0, 0, 0, 0, -k, 0, -k, -k, 0)$
m_2	$(-k, 1, 0, 0, 0, 0, -k, 0, -k, -k, 0)$
m_3	$(-k, -k, k+1, 0, 0, 0, -k, 0, -k, -k, 0)$
m_4	$(-k, -k, 0, k+1, 0, 0, -k, 0, -k, -k, 0)$
m_5	$(-k, -k, 0, 0, k+1, 0, -k, 0, -k, -k, 0)$
m_6	$(-k, -k, 0, 0, 0, k+1, -k, 0, -k, -k, 0)$
m_7	$(-k, -k, 0, 0, 0, 0, 1, 0, -k, -k, 0)$
m_8	$(-k, -k, 0, 0, 0, 0, -k, k+1, -k, -k, 0)$
m_9	$(-k, -k, 0, 0, 0, 0, -k, 0, -1, -k, 0)$
m_{10}	$(-k, -k, 0, 0, 0, 0, -k, 0, -k, 1, 0)$
m_{11}	$(-k, -k, 0, 0, 0, 0, -k, 0, -k, -k, k+1)$

We show that system $Ax \geq w(m_1) - w_k$ has no solution; the other cases are similar and therefore are omitted. Consider the subsystem (first, third and forth rows):

$$\begin{aligned} -u_1 + u_2 + u_5 &\geq 1 \\ u_1 - u_2 - u_3 + u_4 &\geq 0 \\ u_3 - u_4 - u_5 &\geq 0 \end{aligned}$$

We infer the following constraints $-u_3 + u_4 + u_5 > 0$ and $u_3 - u_4 - u_5 \geq 0$ that cannot be satisfied at the same time. We used lpsolve (Notebaert, 2005), a Mixed Integer Linear Programming (MILP) solver, in order to prove that for every molecule m_i there does not exist any solution $u \geq 0$ such that $Au = w(m_i) - w_k$, hence for every $k \geq 0$ network G is k-bounded for w_k.

We illustrate the theoretical result with a numerical experiment. In Figure 5.8 we plot the time dependent behaviour of MEKPP for different values of N. We note that also for small values of N the stochastic and deterministic solutions become indistinguishable.

The condition used to prove the boundedness of the ERK model is stronger that the one we need because it is sufficient but not necessary. This condition requires to solve some systems of linear integer inequalities. Integer programming problems are in the worst case undecidable (Schrijver, 1998); there are however some subclasses of problems that are solvable in polynomial time. For example if the matrix is *totally unimodular* and the right-hand sides of the constraints are integers (Papadimitriou and Steiglitz, 1989). Nevertheless these facts do not allow us to infer anything about decidability of our formulation of boundedness problem.

5.8 Discussion

In this chapter we studied the problem of convergence of Markovian models with discrete levels of concentration. We showed that under some structural conditions and for a sufficiently large number of levels the average behaviour of a Markov process is equal to the solution of the corresponding

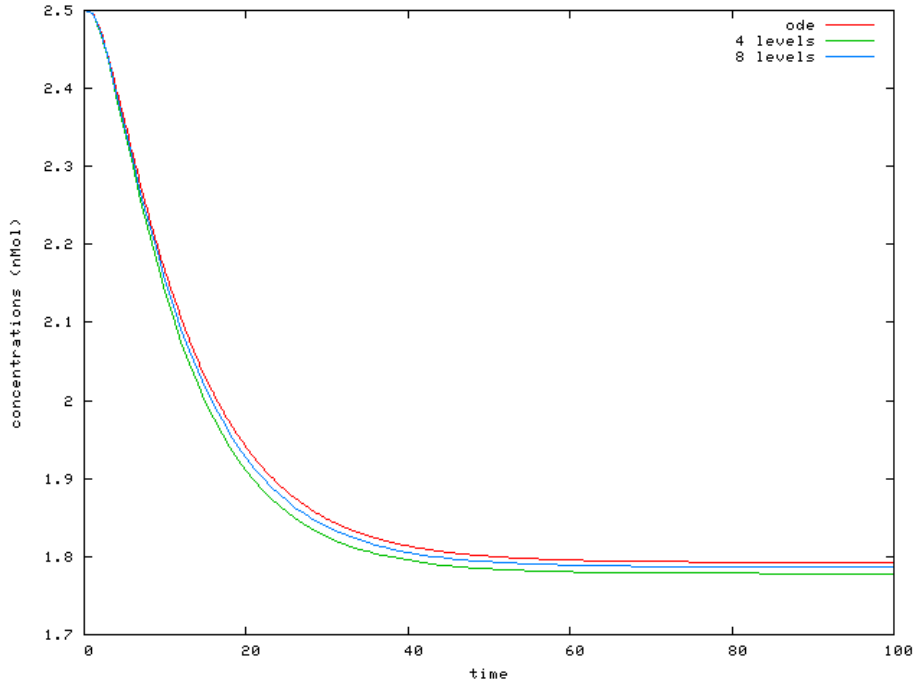


Figure 5.8: **Comparing stochastic and deterministic solutions for the ERK pathway model.** This figure shows the deterministic and stochastic behaviour of MEKPP protein. Also for small values of N the stochastic and deterministic solutions are indistinguishable.

deterministic model.

In some sense we could say that the deterministic model is derived from the stochastic one. We want to tell if the contrary is true: given a deterministic model can we build a stochastic one that is "equivalent" to it? At the moment we are not interested in a formal definition of model equivalence, we say that two model are equivalent if they behave in the same way under some conditions. The answer seems to be negative in general and we propose a simple example where a deterministic model and its stochastic couterpart behave differently.

The Predator-Prey model describes a simple biological system in which two species, predators and preys, interact. This model consists of a pair of nonlinear differential equations,

$$\frac{dx}{dt} = -ax + bxy \quad \frac{dy}{dt} = -bxy + cy \tag{5.11}$$

Here, x and y represent the number of predators and prey respectively while a, b and c are rate constants which model the interaction between the two species. The prey are assumed to reproduce exponentially, that is cy. Otherwise prey are killed by predators $-bxy$; the rate of predation is assumed to be proportional to the probability of meeting between predators and prey. On the other hand, predators die exponentially, namely $-ax$, or else they will grow if they find something to eat, i.e. bxy. We use the same rate constant b for predator growth and prey death in order to adapt the model to our Markovian framework. In the classical model these two constants are different.

Equations 5.11 present periodic solutions. Unluckly there does not exist an analytical solution, however stability and bifurcation analysis allow us to get some useful information on the nonlinear behaviour of the system.

We are interested in discovering for which values of x and y the level of population does not change or, in other words, the system is in equilibrium.

$$\begin{aligned} -ax + bxy &= 0 \\ -bxy + cy &= 0 \end{aligned}$$

The above system of equations yields two solutions, $(0, 0)$ and $(c/b, a/b)$, which are the equilibrium points. Thus equilibrium depends on rate constants.

Then we study the stability of the equilibria using linear stability analysis (see Chapter 3). The Jacobian matrix for the predator-prey model is given by,

$$J(x, y) = \begin{pmatrix} -a + by & bx \\ -by & -bx + c \end{pmatrix}$$

We compute the eigenvalues of the Jacobian matrix evaluated in both the equilibrium points. The matrix $J(0,0)$ has eigenvalues $\lambda_1 = -a$ and $\lambda_2 = c$, hence the equilibrium $(0,0)$ is a saddle point and is unstable. Instead the matrix $J(c/b, a/b)$ has eigenvalues $\lambda_{1,2} = \pm i\sqrt{ac}$, thus the equilibrium $(c/b, a/b)$ is a centre and so the levels of predator and prey polulations oscillate around it.

We built a Markov chain as described above for this model where M is the maximal concentration, $N+1$ the number of discrete levels of concentration and $H = M/N$ the step between consecutive levels. We observe that in this case the Markov chain has two absorbing states corresponding to states $(0,0)$ and $(0,N)$; in general (i,j) is a state where the level of predators is i and the level of prey j. The second absorbing state $(0,N)$ exists because we consider finite Markov chains; if $N \to \infty$, this absorbing state becomes transient. Thus we are interested mainly in the first one $(0,0)$. One may think that state $(0,0)$ corresponds to equilibrium point $(0,0)$ in the deterministic model. In some sense it is true, but the deterministic and Markovian states have a completely different nature. Firstly, every realization of the Markov chain will drop into $(0,0)$ eventually; that is not the case for the deterministic model. Secondly, in the determinstc model $(0,0)$ is an unstable equilibrium point, it means that after a small perturbation the system goes away from equilibrium and yields a limit cycle. Of course, we can not compare equilibrium points and absorbing states, but the behaviour of the equilibrium point seems the contrary of that one of an absorbing state. In conclusion, these two model seem to be qualitatively different and they describe different systems. Note that it is true also if you consider number of molecules instead of concentration levels and we have the same problem also in stochastic simulation.

I consider the lowest level with a different meaning; the concentration is never zero, but it is just very small. I do not know if this assumption makes sense because in this way we lose some information on extinction, however the Markov chain becomes ergodic. I observed that for some values of N and M the equilibrium behaviour of the deterministic model is recovered by the Markov chain; for example if the initial state in the Markov chain corresponds to equilibrium point $(c/b, a/b)$ in the deterministic model, the timed avarage behaviour of the Markov chain is similar to the deterministic one. It is not clear why and how it is true only for particular values of N and

M. Instead the Markov chain with absorbing states reaches the absorbing state later for the same values.

Chapter 6

Conclusions

In this thesis we showed a novel and alternative style of modelling biological systems in the context of process algebra PEPA (Calder et al., 2004). Each molecule is represented by a process and each process has an index that represents the current level of the corresponding molecule. Actions model reactions as usual and rates are computed following some rules. An activity graph is a graphical representation of a PEPA model that represents increasing and descreasing of molecular concentrations in reactions. From an activity graph several mathematical interpretations can be derived. We considered three possible interpretations for a PEPA model: Markov chains with discrete levels of concentration, ordinary differential equations and stochastic simulation based on Gillespie's algorithm (Gillespie, 1976).

We proposed a model for circadian clocks based on the core model for *Neurospora* (Goldbeter, 2002). We showed how a model with discrete concentration levels can also represent different activation levels for abstract biological processes. We illustrated through some numerical experiments how stochastic simulations converge to a deterministic model as the number of involved molecules is sufficiently large. Our approach yields results similar to the literature (Gonze et al., 2002a,b).

We analysed the relationships between Markov chains with discrete levels and differential equations. For some simple examples that can be solved analytically the average Markovian behaviour converges to the solution of the deterministic systems as the number of levels increases. We gener-

alized this result for a class of Markov chains which present some structural properties.

We used Kurtz's results on convergence of density dependent Markov chains (Kurtz, 1970). A density dependent family of Markov chains X_v is a sequence $\{X_v\}$ of Markov processes defined by a parameter v. The states of the Markov chains are normalized with respect to v and transition rates depend on these densities.

The Markovian representation of a PEPA model with discrete levels is a parametrized Markov chain $X_N(t)$, where N is the number of levels. However, we showed that the sequence $\{X_N(t)\}$ is not always density dependent. We formulated a sufficient and necessary structural condition on the state space of $X_N(t)$ in such a way that the sequence $\{X_N(t)\}$ is a sequence of dentity dependent Markov chains and we can apply Kurtz's results on convergence. We proposed an effective way to verify if a model satifies this condition. Roughly speaking we defined a generalization of the notion of activity graph to represent the state spaces of all the Markov chains. Then if the activity graph presents some structural properties (e.g. boundedness), the corresponding sequence is density dependent.

We proved that we cannot apply Kurtz's theorem to the circadian clock model because some molecule species can grow unboundedly. In gerneral we have this problem in every model with positive feedback loops and in general with metabolic pathways. Nevertheless the class of models whose convergence can be proved via Kurtz is quite wide. For example signalling pathway models such as ERK pathway (Cho et al., 2003) belong to this class. We proved that the representation of the ERK signalling pathway with discrete levels of concentrations yields results similar to the deterministic model when the number of levels is *sufficiently* large. These results were observed experimentally also in Calder et al. (2005).

This work offers several possible future extentions. The next step could be to study the convergence rate of Markov models with discrete levels towards the corresponding deterministic models. Besides it would be interesting to compare the convergence of stochastic models with discrete concentration levels and the one of stochastic models with individual molecule species. Another possibility would be to study the behaviour of discrete stochastic models around their limit deterministic distribution varying the number of levels.

In order to make the framework described in this thesis more realistic it is necessary to introduce different maximal concentrations for different molecule species and to allow also arbitrary initial concentrations.

It is important to understand which information (if any) is lost in Markovian models with few levels and if it is possible to infer some properties of the corresponding deterministic models from them. In this way we could perform model checking on the discrete models, e.g.using CSL logic, and obtain results valid also for the differential equations.

We need to extend the results on convergence to a wider class of models. It could be necessary to develop theoretical results similar to Kurtz's theorem.

Finally in this thesis we took in account the average behaviour of a stochastic process as a measure correlated with the solution of the corresponding deterministic model. We observed that under some conditions the mean is equal to the solution of the deterministic system. In general, however, it is not true. As an example consider a deterministic model which presents sustained osciallations, e.g. the circadian clock model. If the corresponding Markov chain is ergodic as in our case, there exists a steady state distribution, hence the mean or any other reward function can describe damped oscillation at most. Dealing with these problems is still an open question.

Chapter 7

Conclusioni

In questa tesi abbiamo mostrato uno stile nuovo ed alternativo per modellare i sistemi biologici nel contesto dell'algebra di processo PEPA (Calder et al., 2004). Ogni molecola è rappresentata da un processo ed ogni processo ha un indice che rappresenta il livelo corrente della molecola corrispondente. Le azioni modellano le reazioni come al solito e le velocità sono calcolate seguendo qualche regola. Un grafo delle attività à una rappresentazione grafica di un modello PEPA che rappresenta gli incrementi e i decrementi delle concentrazioni molecolari nelle reazioni. Dal grafon delle attività sono derivate alcune interpretazioni matematiche. Noi consideriamo tre possibili interpretazioni di un modello PEPA: catene di Markov con livelli discreti di concentrazione, equazioni differenziali ordinarie e simulazioni stocastiche basate sull'algoritmo di Gillespie (Gillespie, 1976).

Abbiamo proposto un modello per i cicli circadiani basato sul modello fondamentale per *Neurospora* (Goldbeter, 2002). Abbiamo mostrato come un modello con livelli discreti di concentrazioni può anche rappresentare livelli di attivazione per dei processi biologici astratti. Abbiamo illustrato attraverso qualche esperimento numerico come le simulazioni stocastiche convergano ad un modello deterministico quando il numero di molecole coinvolte ùfficientemente grande. Il nostro approccio produce risultati simili alla letteratura (Gonze et al., 2002a,b).

Abbiamo analizzato le relazioni tra le catene di Markov con livelli discreti e le equazioni differenziali. Per alcuni semplici esempi che possono essere risolti analiticamente il comportamento

markoviano medio converge alla soluzione dei sistemi deterministici quando il numero di livelli cresce. Abbiamo generalizzato questo risultato ad una classe di catene di Markov che presentano alcune proprietà strutturali.

Abbiamo utilizzato i risultati di Kurtz sulla convergenza delle catene di Markov dipendenti dalla densità (Kurtz, 1970). Una famiglia di catene di Markov X_v dipendente dalla densità è una sequenza $\{X_v\}$ di processi di Markov definiti da un parametro v. Gli stati delle catene di Markov sono normalizzati rispetto a v e le velocità di transizione dipendono da queste densità.

La rappresentazione markoviana di un modello PEPA con livelli discreti di concentrazione è una catena di Markov parametrizzata $X_N(t)$, dove N è il numero di livelli. Però abbiamo mostrato che la sequenza $\{X_N(t)\}$ non è sempre dipendente dalla densità. Abbiamo formulato una condizione strutturale sufficiente e necessaria sullo spazio degli stati di $X_N(t)$ per la quale la sequenza $\{X_N(t)\}$ è una sequenza di catene di Markov dipendenti dalla densità e possiamo applicare i risultati di Kurtz sulla convergenza. Abbiamo proposto un metodo effettivo per verificare se un modello soddisfa questa condizione. In parole povere abbiamo definito una generalizzazione della nozione di grafo delle attività per rappresentare gli spazi degli stati di tutte le catene di Markov. Poi, se il grafo delle attività presenta alcune proprietà strutturali (cioè è bounded), la sequenza corrispondente è dipendente dalla densità.

Abbiamo provato che non possiamo applicare il teorema di Kurtz al modello del ciclo circadiano perchè alcune molecole possono crescere senza limiti. In generale abbiamo questo problema in ogni modello con feedback positivo ed con pathway metabolici. Nonostante questo la classe dei modelli la cui convergenza può essere provata via Kurtz è abbastanza vasta. Ad esempioi modelli per pathway di segnali biologici come quello per ERK (Cho et al., 2003) appartengono a questa classe. Abbiamo provato che la rappresentazione del pathway dell'ERK con livelli discreti di concentrazione produce risultati simili al modello deterministico quando il numero di livelli è sufficientemente grande. Questi risultati sono stati osservati numericamente anche in Calder et al. (2005).

Questo lavoro si presta a diverse possibili estensioni future. Il prossimo passo potrebbe essere studiare la velocità di convergenza dei modelli di Markov con livelli discreti verso il corrispondente

modello deterministico. Inoltre sarebbe interessante confrontare la convergenza dei modelli stocastici con livelli discreti di concentrazione e quella dei modelli stocastici con le singole molecole. Un'altra possibilità potrebbe essere studiare il comportamento dei modelli stocastici discreti intorno alla loro distribuzione limite deterministica variando il numero di livello.

Per rendere l'approccio descritto in questa tesi più realistico è necessario introdurre differenti concentrazioni massime per tipi di molecole differenti e permettere anche arbitrarie concentrazioni iniziali.

È importante capire quale informazione viene persa (se accade) nei modelli markoviani con pochi livelli e se è possibile inferire alcune proprietà dei corrispondenti modelli deterministici da questi. In questo modo potremmo eseguire del model checking sui modelli discreti, ad esempio usando logiche tipo CSL, ed ottenere risultati validi anche per le equazioni differenziali.

È necessario estendere i risultati della convergenza ad una classe più ampia di modelli. Potrebbe essere necessario sviluppare risultati teorici simili al teorema di Kurtz.

Infine in questa tesi abbiamo preso in considerazione il comportamento medio dei processi stocastici come misura relazionata con la soluzione del corrispondente modello deterministico. Abbiamo osservato che sotto certe condizioni la media è uguale alla soluzione del sistema deterministico. In generale, però, non è vero. Per esempio considera un modello deterministico che prensenta delle oscillazioni periodiche, ad esempio il modello per il ciclo circadiano. Se la corrispondente catena di Markov è ergodica come nel nostro caso, esiste una distribuzione steady state, quindi la media o qualsiasi altra funzione di rewarding può descrivere al più delle oscillazioni smorzate. Gestire questi problemi è ancora una questione aperta.

Appendix A

PEPA: the language

In this chapter a brief introduction to PEPA language can be found. The explanation is quite informal and readers interested in details can have a look at the work of Jane Hillston Hillston (1995).

A.1 Syntax

In PEPA systems are represented as *components* which take part to *activities*. The syntax of PEPA is defined by the following grammar rules:

$$S ::= (\alpha, r).X \mid S + S$$

$$P ::= P \underset{L}{\bowtie} P \mid P/L \mid X$$

Here S is a *sequential component* and P is a *model component*. X is a constant name which refers to a sequential component. We assume that every constant X are associated to exactly one sequential component. We observe that operator $\bowtie$ can be only at top level; this is a necessary condition for the ergodicity of the underlying Markovian model (Hillston, 1995). PEPA has five combinators: prefix, choice, constant, hiding and cooperation.

Prefix is the basic component to build up complex systems, the process $(\alpha, r).P$ carries out action α at rate r and then it behaves as P. The prefix (α, r) is termed activity while α is the action

type and r the rate of the activity.

Choice models competition between two processes: the component $P + Q$ represents a system which may evolve either into P or Q.

Constant allows us to assign names to components, for example $X \stackrel{def}{=} (\alpha, r).P$ means that variable X behaves as process $(\alpha, r).P$.

Hiding is a mechanism to abstract away some aspects of a component's behaviour. For instance, the process $P \setminus \{\alpha\}$ hides the action α and prevent other processes to join in.

Cooperation allows two processes to be synchronized over a set of actions. In expression $P \underset{L}{\bowtie} Q$ processes P and Q must cooperate on actions contained in the set L, but other enabled actions are carried out independently and concurrently. When L is empty, we write $P||Q$ instead of $P \underset{L}{\bowtie} Q$.

When a component enables an activity whose action type is in the cooperation set, it will be stuck until the other component enable an activity of that type. The rates of shared activities depend on the rate of both cooperands' rates. In other words, the appearent rate of a shared activity is the rate of the slower component.

Sometimes a component may be passive with respect to an action in a cooperation set. Consider for example a client waiting for a service. In these cases the rate of the activity is unspecified (symbol $\top$) and it depends on the rate of the activity of the other cooperand. All passive actions must be synchronized in the final model.

The action types which a component P can next undertake form the set of the current action types of P, denoted $A(P)$. The activities which a component P can next perform are the current activities of P, denoted $Act(P)$.

A.2 Semantics

The structured operational semantics of PEPA are shown in Figure A.2.

Definition A.2.1 (Hillston (1995)). *The apparent rate of action type α in a component P, denoted $r_\alpha(P)$ is the sum of the rates of all activities of type α in $Act(P)$. More formally,*

Prefix

$$\frac{}{(\alpha, r).E \xrightarrow{(\alpha,r)} E}$$

Choice

$$\frac{E \xrightarrow{(\alpha,r)} E'}{E + F \xrightarrow{(\alpha,r)} E'}$$

$$\frac{F \xrightarrow{(\alpha,r)} F'}{E + F \xrightarrow{(\alpha,r)} F'}$$

Hiding

$$\frac{E \xrightarrow{(\alpha,r)} E'}{E/L \xrightarrow{(\alpha,r)} E'/L} \ (\alpha \notin L)$$

$$\frac{E \xrightarrow{(\alpha,r)} E'}{E/L \xrightarrow{(\tau,r)} E'/L} \ (\alpha \in L)$$

Cooperation

$$\frac{E \xrightarrow{(\alpha,r)} E'}{E \underset{L}{\bowtie} F \xrightarrow{(\alpha,r)} E' \underset{L}{\bowtie} F} \ (\alpha \notin L)$$

$$\frac{F \xrightarrow{(\alpha,r)} F'}{E \underset{L}{\bowtie} F \xrightarrow{(\alpha,r)} E \underset{L}{\bowtie} F'} \ (\alpha \notin L)$$

$$\frac{E \xrightarrow{(\alpha,r_1)} E' \quad F \xrightarrow{(\alpha,r_2)} F'}{E \underset{L}{\bowtie} F \xrightarrow{(\alpha,R)} E' \underset{L}{\bowtie} F'} \ (\alpha \in L)$$

where $R = \dfrac{r_1}{r_\alpha(E)} \dfrac{r_2}{r_\alpha(F)} r_\alpha(E \underset{L}{\bowtie} F)$

Constant

$$\frac{E \xrightarrow{(\alpha,r)} E'}{A \xrightarrow{(\alpha,r)} E'} \ (A \stackrel{def}{=} E)$$

Figure A.1: PEPA Structured Operational Semantics

- $r_\alpha((\beta, r).P) = \begin{cases} r & \text{if } \alpha = \beta \\ 0 & \text{if } \alpha = 0 \end{cases}$
- $r_\alpha(P + Q) = r_\alpha(P) + r_\alpha(Q)$
- $r_\alpha(P/L) = \begin{cases} r_\alpha(P) & \text{if } \alpha \notin L \\ 0 & \text{if } \alpha \in L \end{cases}$
- $r_\alpha(P \underset{L}{\bowtie} Q) = \begin{cases} min(r_\alpha(P), r_\alpha(Q)) & \text{if } \alpha \in L \\ r_\alpha(P) + r_\alpha(Q) & \text{if } \alpha \notin L \end{cases}$

For a PEPA component the set of derivatives is the set of all the behaviours into which the component can evolve.

Definition A.2.2 (Hillston (1995)). *The derivative set of a PEPA component C is denoted $ds(C)$ and defined recursively as the smallest set of components such that:*

- $C \in ds(C)$;
- *if $C_i \in ds(C)$ and there exists $a \in Act(C_i)$ such that $C_i \xrightarrow{a} C_j$, then $C_j \in ds(C)$.*

Appendix B

Model specifications

In this chapter we report the algebraic specifications in PEPA language for each model presented in this thesis. For every specifications we list also the underlying mathematical interpretations, such as ordinary differential equations, reactions for Gillespie's algorithm and Markov chains with discrete levels of concentrations.

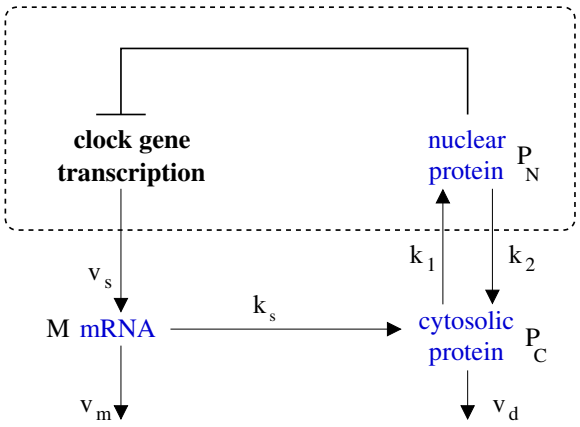
Ordinary differential equations are described with the standard mathematical representation. Experiments were carried out using Runge-Kutta 5 method implemented in GNU/Octave (Eaton, 2005), an high-level Matlab-like language for solving linear and nonlinear problems numerically.

Stochastic simulations are described by a set of chemical reactions using the built-in language of Dizzy (Ramsey, 2006). Dizzy is a chemical kinetics simulation software package written in the Java programming language. It allows to define models as systems of chemical reactions. It performs several kinds of stochastic and deterministic simulations (e.g. Gillespie)

Markov chains are described using PRISM language (Parker et al., 2006) following conventions described in (Calder et al., 2005). PRISM is a probabilistic model checker written in Java for modelling and analysing probabilistic systems. It supports continuous time Markov chain models and implements CSL model checking (Aziz et al., 1996), a logic to express properties of steady state and transient behaviour of Markov processes. Transient and steady state analysis of Markov chains was performed using CSL formulas

B.1 Circadian clock model

Several theoretical models have been proposed for circadian clocks in different organisms (Goldbeter, 2002). Following Gonze et al. (2002a) we consider the minimal model initially suggested for circadian rhythms in *Neurospora*. M represents mRNA while P_C and P_N are the clock proteins into the cytosol and into the nucleous respectively. P_C is synthesized from the mRNA M, then it is either transported into the nucleus or degradated. P_N exerts a negative feedback on transcription of its gene or else it goes out of the nucleus. Degradations are controlled by enzymes. The time



evolution of the concentrations involved in the model is given by the following kinetic equations,

$$\begin{aligned}
\frac{d[M]}{dt} &= v_s \frac{k_I^n}{k_I^n + [P_n]^n} - v_m \frac{[M]}{k_m + [M]} \\
\frac{d[P_C]}{dt} &= k_s M - v_d \frac{[P_C]}{k_d + [P_C]} - k_1[P_C] + k_2[P_N] \\
\frac{d[P_N]}{dt} &= k_1[P_C] - k_2[P_N]
\end{aligned}$$

PEPA model

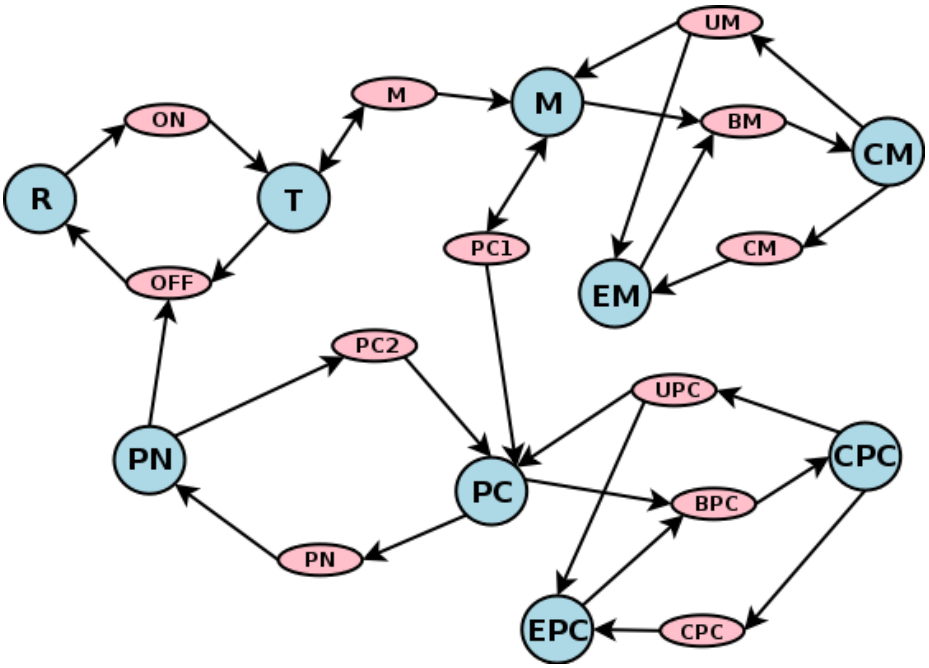
$$
\begin{aligned}
T^h &\stackrel{def}{=} (m, v_s).T^h + (off, \top).T^l \\
T^l &\stackrel{def}{=} (on, \top).T^h \\
R^h &\stackrel{def}{=} (on, v_{on}).R^l \\
R^l &\stackrel{def}{=} (off, \top).R^h \\
M^l &\stackrel{def}{=} (m, \top).M^h + (um, v_{um}).M^h \\
M^h &\stackrel{def}{=} (pc_1, k_s).M^h + (bm, v_{bm}).M^l \\
E^h_M &\stackrel{def}{=} (bm, v_{bm}).E^l_M \\
E^l_M &\stackrel{def}{=} (um, v_{um}).E^h_M + (cm, v_{cm}).E^h_M \\
C^h_M &\stackrel{def}{=} (um, v_{um}).C^l_M + (cm, v_{cm}).C^l_M \\
C^l_M &\stackrel{def}{=} (bm, v_{bm}).C^h_M \\
P^l_C &\stackrel{def}{=} (pc_1, \top).P^h_C + +(pc_2, k_2).P^l_C + (upc, v_{upc}).P^h_C \\
P^h_C &\stackrel{def}{=} (pn, k_1).P^l_C + (bpc, v_{bpc}).P^l_C \\
E^h_{P_C} &\stackrel{def}{=} (bpc, v_{bpc}).E^l_{P_C} \\
E^l_{P_C} &\stackrel{def}{=} (upc, v_{upc}).E^h_{P_C} + (cpc, v_{cpc}).E^h_{P_C} \\
C^h_{P_C} &\stackrel{def}{=} (upc, v_{upc}).C^l_{P_C} + (cpc, v_{cpc}).C^l_{P_C} \\
C^l_{P_C} &\stackrel{def}{=} (bpc, v_{bpc}).C^h_{P_C} \\
P^l_N &\stackrel{def}{=} (pn, \top).P^h_N \\
P^h_N &\stackrel{def}{=} (pc_2, k_2).P^l_N + (off, v_{off}).P^l_N
\end{aligned}
$$

$$
T^h \underset{J}{\bowtie} (R^l \underset{K}{\bowtie} (((M^l \underset{L}{\bowtie} E^h_M) \underset{M}{\bowtie} C^l_M) \underset{N}{\bowtie} (((P^l_C \underset{O}{\bowtie} E^h_{P_C}) \underset{P}{\bowtie} C^l_{P_C}) \underset{Q}{\bowtie} P^l_N)))
$$

Where $J = \{m, off, on\}$, $K = \{off\}$, $L = \{um, bm\}$, $M = \{um, bm, cm\}$, $N = \{pc_1\}$, $O = \{upc, bpc\}$, $P = \{upc, bpc, cpc\}$ and $R = \{pn, pc_2\}$. In the initial state we have high concentrations of enzymes (E_M and E_{P_C}) and the transcription machinery T is working at high level; instead concentrations of

the enzyme-substrate complexes (C_M and C_{P_C}) and of mRNA (M), citosolic protein P_C and nuclear protein P_N are low.

Activity graph



ODE model

$$
\begin{aligned}
\frac{d[T]}{dt} &= -v_{off}[T][P_N] + v_{on}[R] \\
\frac{d[R]}{dt} &= v_{off}[T][P_N] - v_{on}[R] \\
\frac{d[M]}{dt} &= v_s[T] - v_{bm}[M][E_M] + v_{um} * [C_M] \\
\frac{d[E_M]}{dt} &= -v_{bm}[M][E_M] + v_{um}[C_M] + v_{cm}[C_M] \\
\frac{d[C_M]}{dt} &= v_{bm}[M][E_M] - v_{um}[C_M] - v_{cm}[C_M] \\
\frac{d[P_C]}{dt} &= k_s[M] - k_1[P_C] + k_2[P_N] - v_{bpc}[P_C][E_{P_C}] + v_{upc}[C_{P_C}] \\
\frac{d[E_{P_C}]}{dt} &= -v_{bpc}[P_C][E_{P_C}] + v_{upc}[C_{P_C}] + v_{cpc}[C_{P_C}] \\
\frac{d[C_{P_C}]}{dt} &= v_{bpc}[P_C][E_{P_C}] - v_{upc}[C_{P_C}] - v_{cpc}[C_{P_C}] \\
\frac{d[P_N]}{dt} &= k_1[P_C] - k_2[P_N] - v_{off}[T][P_N]
\end{aligned}
$$

Dizzy model

```
#model "clock";


// volume parameter
OMEGA = 1000;


// deterministic rates
vs = 0.5  ;
off = 0.4   ;
on = 0.2  ;
vbm = 16.5;
vum = 3.0;
vcm = 0.3;
vbpc = 165.0;
vupc = 15.0;
vcpc = 1.5;
k1 = 0.2;
k2 = 0.2;
ks = 2.0;


// stochastic rates
s_vs = vs  ;
s_off = off / OMEGA ;
s_on = on  ;
s_vbm = vbm / OMEGA;
s_vum = vum;
s_vcm = vcm;
```

```
s_vbpc = vbpc / OMEGA;

s_vupc = vupc;

s_vcpc = vcpc;

s_k1 = k1;

s_k2 = k2;

s_ks = ks;


// initial population

EM = 1*OMEGA;

M = 0;

CM = 0;

EPC = 1*OMEGA;

PC   = 0;

CPC = 0;

PN = 0;

T = 1*OMEGA;

R = 0;


enzyme_mRNA_combine,         EM + M -> CM     ,  s_vbm;

enzyme_substrate_separate,   CM    -> EM + M ,   s_vum;

degradate_mRNA,                  CM    -> EM     ,   s_vcm;


enzyme_PC_combine,EPC + PC -> CPC, s_vbpc;

enzyme_PC_separate,CPC -> EPC + PC, s_vupc;

degradate_PC,CPC -> EPC, s_vcpc;


translation,M -> M + PC, s_ks;
```

```
move_into_nucleous,PC -> PN, s_k1;
move_out_of_nucleous,PN -> PC, s_k2;

transcription,T -> T + M, s_vs;
switch_off,T + PN -> R, s_off;
switch_on,R -> T, s_on;
```

B.2 Decay model

We consider a simple model which describes the degradation process of a kind of molecule or substance *A*. The deterministic behaviour is represented by the following differential equations,

$$\begin{aligned} \frac{da(t)}{dt} &= -\mu a(t) \\ a(0) &= A_0 \end{aligned}$$

It is easy to verify that the system solution is given by,

$$a(t) = A_0 e^{-\mu t}$$

PRISM model

```
stochastic

const int N = 1000;
const int MAX = 100;
const double R = MAX/N;
const double mu = 0.5;
```

```
module AProcess
A: [0..N] init N;
[degradate] (A>0)-> A*R : (A'=A-1);
[null]     (A=0) -> 1: (A'=A);
endmodule

module ConstantProcess
dummy: bool init true;
[degradate] (dummy=true) -> mu/R : (dummy'=true);
endmodule

system
AProcess || ConstantProcess
endsystem

rewards
true: A*R:
endrewards
```

CSL properties

```
// T represents time
const double T;
// time dependent behaviour
// return reward A*R at instant time (I) T.
R=? [ I=T ]
```

B.3 Growth model

We consider a model which represents growing of a population of molecules of kind *A*. The differential equation is given by

$$\begin{aligned} \frac{da(t)}{dt} &= \lambda a(t) \\ a(0) &= A_0 > 0 \end{aligned}$$

The only solution is

$$a(t) = A_0 e^{\lambda t}$$

PRISM model

```
stochastic

const int N = 100;
const double lambda = 0.5;
const double A0 = 2.0;

module AProcess
A: [1..N] init 1;
[degradate] (A<N)-> (A*A0) : (A'=A+1);
[null]     (A=N)-> 1: (A'=A);
endmodule

module ConstantProcess
```

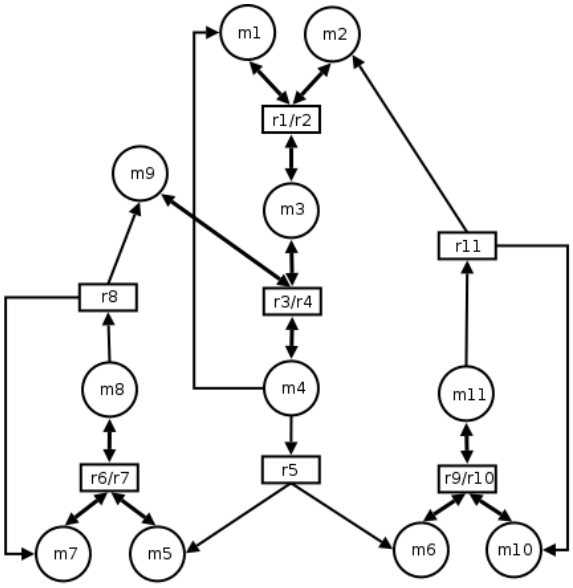
```
dummy: bool init true;
[degradate] (dummy=true) -> lambda/A0 : (dummy'=true);
endmodule

system
AProcess || ConstantProcess
endsystem

rewards
true:  A*R;
endrewards
```

CSL properties

```
// T represents time
const double T;
// time dependent behaviour
// return reward A*R at instant time (I) T.
R=?[I=T]
```

B.4 ERK pathway model

The ERK signalling pathway is a biological process involved in cellular division and differentiation. Here, we consider a model presented in Cho et al. (2003) that describes how RKIP regulates the behaviour of the ERK pathway. In Figure 5.6 there is a graphical representation of the model.



PEPA model

$$m_1[0] \stackrel{def}{=} (r_2, v_{r_2}).m_1[1] + (r_5, v_{r_5}).m_1[1]$$

$$m_1[1] \stackrel{def}{=} (r_1, v_{r_1}).m_1[0]$$

$$m_2[0] \stackrel{def}{=} (r_2, v_{r_2}).m_2[1] + (r_{11}, v_{r_{11}}).m_2[1]$$

$$m_2[1] \stackrel{def}{=} (r_1, v_{r_1}).m_2[0]$$

$$m_3[0] \stackrel{def}{=} (r_1, v_{r_1}).m_3[1] + (r_4, v_{r_4}).m_3[1]$$

$$m_3[1] \stackrel{def}{=} (r_2, v_{r_2}).m_3[0] + (r_3, v_{r_3}).m_3[0]$$

$$m_4[0] \stackrel{def}{=} (r_3, v_{r_3}).m_4[1]$$

$$m_4[1] \stackrel{def}{=} (r_4, v_{r_4}).m_4[0] + (r_5, v_{r_5}).m_4[0]$$

$$m_5[0] \stackrel{def}{=} (r_5, v_{r_5}).m_5[1] + (r_5, v_{r_5}).m_5[1]$$

$$m_5[1] \stackrel{def}{=} (r_7, v_{r_7}).m_5[0]$$

$$m_6[0] \stackrel{def}{=} (r_5, v_{r_5}).m_6[1] + (r_{10}, v_{r_{10}}).m_6[1]$$

$$m_6[1] \stackrel{def}{=} (r_9 v_{r_9}).m_6[0]$$

$$m_7[0] \stackrel{def}{=} (r_7, v_{r_7}).m_7[1]$$

$$m_7[1] \stackrel{def}{=} (r_6, v_{r_6}).m_7[0]$$

$$m_8[0] \stackrel{def}{=} (r_6, v_{r_6}).m_8[1]$$

$$m_8[1] \stackrel{def}{=} (r_7, v_{r_7}).m_8[0] + (r_8, v_{r_8}).m_8[0]$$

$$m_9[0] \stackrel{def}{=} (r_4, v_{r_4}).m_9[1] + (r_8, v_{r_8}).m_9[1]$$

$$m_9[1] \stackrel{def}{=} (r_3, v_{r_3}).m_9[0]$$

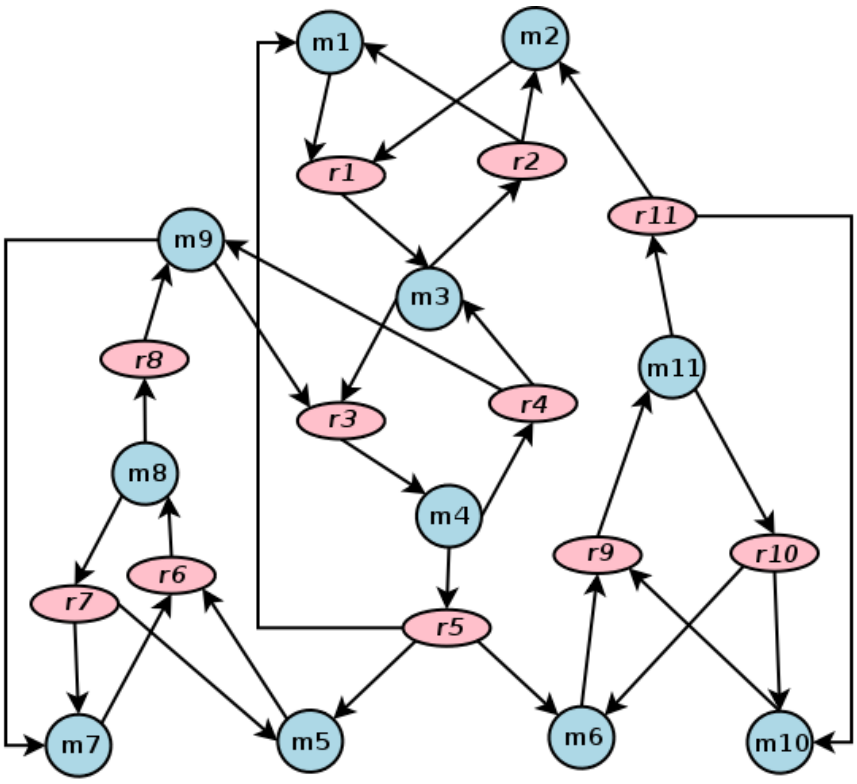
$$m_{10}[0] \stackrel{def}{=} (r_{10}, v_{r_{10}}).m_{10}[1]$$

$$m_{10}[1] \stackrel{def}{=} (r_9, v_{r_9}).m_{10}[0]$$

$$m_{11}[0] \stackrel{def}{=} (r_9, v_{r_9}).m_{11}[1]$$

$$m_{11}[1] \stackrel{def}{=} (r_{10}, v_{r_{10}}).m_{11}[0] + (r_{11}, v_{r_{11}}).m_{11}[0]$$

Activity graph



ODE model

The following set of nonlinear ODEs is extracted from the PEPA model and it is the same as that one in Cho et al. (2003).

$$
\begin{aligned}
\frac{dm_1(t)}{dt} &= -k_1 m_1(t) m_2(t) + k_2 m_3(t) + k_5 m_4(t) \\
\frac{dm_2(t)}{dt} &= -k_1 m_1(t) m_2(t) + k_2 m_3(t) + k_{11} m_{11}(t) \\
\frac{dm_3(t)}{dt} &= k_1 m_1(t) m_2(t) - k_2 m_3(t) - k_3 m_3(t) m_9(t) + k_4 m_4(t) \\
\frac{dm_4(t)}{dt} &= k_3 m_3(t) m_9(t) - k_4 m_4(t) - k_5 m_4(t) \\
\frac{dm_5(t)}{dt} &= k_5 m_4(t) - k_6 m_5(t) m_7(t) + k_7 m_8(t) \\
\frac{dm_6(t)}{dt} &= k_5 m_4(t) - k_9 m_6(t) m_{10}(t) + k_{10} m_{11}(t) \\
\frac{dm_7(t)}{dt} &= -k_6 m_5(t) m_7(t) + k_7 m_8(t) + k_8 m_8(t) \\
\frac{dm_8(t)}{dt} &= k_6 m_5(t) m_7(t) - k_7 m_8(t) - k_8 m_8(t) \\
\frac{dm_9(t)}{dt} &= -k_3 m_3(t) m_9(t) + k_4 m_4(t) + k_8 m_8(t) \\
\frac{dm_{10}(t)}{dt} &= -k_9 m_6(t) m_{10}(t) + k_{10} m_{11}(t) + k_{11} m_{11}(t) \\
\frac{dm_{11}(t)}{dt} &= k_9 m_6(t) m_{10}(t) - k_{10} m_{11}(t) - k_{11} m_{11}(t)
\end{aligned}
$$

PRISM model

Source code as in Calder et al. (2005).

```
stochastic


const int N = 3;
const double Ro = 2.5/N;
```

```
rate k1 = 0.53;

module RAF1
RAF1: [0..N] init N;
[r1] (RAF1 > 0) -> RAF1*Ro: (RAF1' = RAF1 - 1);
[r2] (RAF1 < N) -> 1: (RAF1' = RAF1 + 1);
[r5] (RAF1 < N) -> 1: (RAF1' = RAF1 + 1);
endmodule

module RKIP
RKIP: [0..N] init N;
[r1] (RKIP > 0) -> RKIP*Ro: (RKIP' = RKIP - 1);
[r2] (RKIP < N) -> 1: (RKIP' = RKIP + 1);
[r8] (RKIP < N) -> 1: (RKIP' = RKIP + 1);
endmodule

module RAF1RKIP
RAF1RKIP: [0..N] init 0;
[r1] (RAF1RKIP < N) -> 1: (RAF1RKIP' = RAF1RKIP + 1);
[r2] (RAF1RKIP > 0) -> RAF1RKIP*Ro: (RAF1RKIP' = RAF1RKIP - 1);
[r3] (RAF1RKIP > 0) -> RAF1RKIP*Ro: (RAF1RKIP' = RAF1RKIP - 1);
[r4] (RAF1RKIP < N) -> 1: (RAF1RKIP' = RAF1RKIP + 1);
endmodule

module ERKPP
ERKPP: [0..N] init N;
[r3] (ERKPP > 0) -> ERKPP*Ro: (ERKPP' = ERKPP - 1);
```

```
[r4] (ERKPP < N) -> 1: (ERKPP' = ERKPP + 1);
[r11] (ERKPP < N) -> 1: (ERKPP' = ERKPP + 1);
endmodule

module RAF1RKIPERKPP
RAF1RKIPERKPP: [0..N] init 0;
[r3] (RAF1RKIPERKPP < N) -> 1: (RAF1RKIPERKPP' = RAF1RKIPERKPP + 1);
[r4] (RAF1RKIPERKPP > 0) -> RAF1RKIPERKPP*Ro: (RAF1RKIPERKPP' = RAF1RKIPERKPP - 1);
[r5] (RAF1RKIPERKPP > 0) -> RAF1RKIPERKPP*Ro: (RAF1RKIPERKPP' = RAF1RKIPERKPP - 1);
endmodule

module ERK
ERK: [0..N] init 0;
[r5] (ERK < N) -> 1: (ERK' = ERK + 1);
[r9] (ERK > 0) -> ERK*Ro: (ERK' = ERK - 1);
[r10] (ERK < N) -> 1: (ERK' = ERK + 1);
endmodule

module RKIPP
RKIPP: [0..N] init 0;
[r5] (RKIPP < N) -> 1: (RKIPP' = RKIPP + 1);
[r6] (RKIPP > 0) -> RKIPP*Ro: (RKIPP' = RKIPP - 1);
[r7] (RKIPP < N) -> 1: (RKIPP' = RKIPP + 1);
endmodule

module RP
RP: [0..N] init N;
```

```
[r6] (RP > 0) -> RP*Ro: (RP' = RP - 1);
[r7] (RP < N) -> 1: (RP' = RP + 1);
[r8] (RP < N) -> 1: (RP' = RP + 1);
endmodule

module MEKPP
MEKPP: [0..N] init N;
[r9] (MEKPP > 0) -> MEKPP*Ro: (MEKPP' = MEKPP - 1);
[r10] (MEKPP < N) -> 1: (MEKPP' = MEKPP + 1);
[r11] (MEKPP < N) -> 1: (MEKPP' = MEKPP + 1);
endmodule

module MEKPPERK
MEKPPERK: [0..N] init 0;
[r9] (MEKPPERK < N) -> 1: (MEKPPERK' = MEKPPERK + 1);
[r10] (MEKPPERK > 0) -> MEKPPERK*Ro: (MEKPPERK' = MEKPPERK - 1);
[r11] (MEKPPERK > 0) -> MEKPPERK*Ro: (MEKPPERK' = MEKPPERK - 1);
endmodule

module RKIPPRP
RKIPPRP: [0..N] init 0;
[r6] (RKIPPRP < N) -> 1: (RKIPPRP' = RKIPPRP + 1);
[r7] (RKIPPRP > 0) -> RKIPPRP*Ro: (RKIPPRP' = RKIPPRP - 1);
[r8] (RKIPPRP > 0) -> RKIPPRP*Ro: (RKIPPRP' = RKIPPRP - 1);
endmodule

module Constants
```

```
fake: bool init true;
[r1] (fake) -> k1/Ro: (fake' = true);
[r2] (fake) -> 0.0072/Ro: (fake' = true);
[r3] (fake) -> 0.625/Ro: (fake' = true);
[r4] (fake) -> 0.00245/Ro: (fake' = true);
[r5] (fake) -> 0.0315/Ro: (fake' = true);
[r6] (fake) -> 0.92/Ro: (fake' = true);
[r7] (fake) -> 0.00122/Ro: (fake' = true);
[r8] (fake) -> 0.87/Ro: (fake' = true);
[r9] (fake) -> 0.8/Ro: (fake' = true);
[r10] (fake) -> 0.0075/Ro: (fake' = true);
[r11] (fake) -> 0.071/Ro: (fake' = true);
endmodule

rewards
true: MEKPP*Ro;
endrewards
```

CSL logic

We can verify if a Markov chain is density dependent using CSL logic. In other words, we need to prove that, when some products of a reaction have level N, then reagents of the reaction have level zero. As usual N is the maximal level.

The following properties in CSL represent the probabilities that the reagents of a reaction have level greater than zero when some products have level equal to N in every point of the evolution of the system . Since all the probabilities are zero for the ERK model, the Markov chain is density dependent for N. Readers can try to verify this property for increasing values of N.

```
P=?[true U ((RAF1 > 0 & RKIP>0) | RAF1RKIPERKPP>0 ) & RAF1RKIP=N]

P=?[true U RAF1=N & (RAF1RKIP>0 | RAF1RKIPERKPP>0)]

P=?[true U RKIP=N & (RAF1RKIP>0 | RKIPPRP>0)]

P=?[true U RAF1RKIPERKPP=N & (RAF1RKIP>0 & ERKPP>0)]

P=?[true U ERKPP=N & (MEKPPERK>0 | RAF1RKIPERKPP>0)]

P=?[ true U MEKPPERK=N & (MEKPP>0 & ERK>0)]

P=?[true U MEKPP=N & MEKPPERK>0]

P=?[true U ERK=N & RAF1RKIPERKPP>0]

P=?[true U RKIPP=N & (RAF1RKIPERKPP>0 | RKIPPRP>0)]

P=?[ true U RP=N & RKIPPRP>0]

P=?[true U RKIPPRP=N & (RKIPP>0 & RP>0)]
```

Appendix C

Glossary of biological terms

Here, readers who do not have familiarity with molecular biology can find a definition of the biological terms used in this thesis. Definition are taken and adapted from Alberts et al. (2004).

DNA molecule formed by a long chain of nucleotides. It containts hereditary information.

messenger RNA (mRNA) molecule formed by a long chain of nucleotides. It is a complementar "copy" (without non coding regions) of one or more genes and contains information to produce proteins.

enzyme protein that catalyzes a specific chemical reaction.

protein a molecule formed by a sequence of amino acids. Proteins have several functions, e.g. signalling, transporting and structure.

gene a region of DNA that controls a hereditary characteristic, usually it corresponds to a single protein or RNA.

transcription copying a strand of DNA into a complementary RNA sequence. Transcription is undertaken by the enzyme RNA polymerase. In eucaryotes transcription occurs inside the nucleus.

translation creating of a sequence of amino acids using a sequence of mRNA. It occurs on a ribosome outside the nucleus.

nucleus membrane-bounded organelle in a eucaryotic cell. It contains DNA.

Bibliography

Alberts, , Johnson, Lewis, Raff, Roberts, and Walter. *The Cell*. Garland Publishing, 2004.

A. Aziz, K. Sanwal, V. Singhal, , and R. Brayton. Verifying continuous time markov chains. In *CAV'96*, pages 269–276, 1996.

M. Calder, V. Vyshemirsky, D. Gilbert, and R. Orton. Analysis of signalling pathways using the PRISM model checker. In *Proc. Computational Methods in Systems Biology (CMSB'05)*, 2005.

Muffy Calder, Stephen Gilmore, and Jane Hillston. Automatically deriving ODEs from process algebra models of signalling pathways.

Muffy Calder, Stephen Gilmore, and Jane Hillston. Modelling the influence of RKIP on the ERK signalling pathway using the stochasitc process algebra PEPA. In *Bio-Concur*. 2004.

Kwang-Hyun Cho, Sung-Young Shin, Hyun-Woo Kim, Olaf Wolkenhauer, Brian McFerran, and Walter Kolch. Mathematical modelling of the influence of RKIP on the ERK signalling pathway. *Computational Methods in Systems Biology*, 2003.

Michael Cox and David L. Nelson. *Principles of Biochemistry*. Palgrave Macmillan, 4th edition, 2005.

John W. Eaton. *GNU Octave Manual*. -, 2005.

D.T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Phisics*, 22(403-434), 1976.

Stephen Gilmore. *the PEPA Workbench: user's manual*. University of Edinburgh, September 2001.

Albert Goldbeter. Computational approaches to cellular rhythms. *Nature*, 420:238–245, November 2002.

D. Gonze, J. Halloy, and P. Gaspard. Biochemical clocks and molecular noise: theoretical study of robustness factors. *Journal of chemical physics*, 116(24):10997–11010, June 2002a.

D. Gonze, J. Halloy, and A. Goldbeter. Deterministic versus stochastic models for circadian rhythms. *Journal of biological physics*, 28:637–653, 2002b.

Jane Hillston. *A compositional approach to performance modelling*. Cambridge University Press, 1995.

Donald Knuth. Two notes on notation. In *AMM 99*, number 5, pages 403–422. May 1992.

T. G. Kurtz. Solutions of ordinary differential equations as limits of pure jump markov processes. *Journal of Applied Probability*, 7(1):49–58, April 1970.

T. G. Kurtz. Limit theorems for sequences of jump markov processes approximating ordinary differential process. *Journal of Applied Probability*, 8(2):344–356, June 1971.

Yuri A. Kuznetsov. *Elements of Applied Bifurcation Theory*. Springer, 2004.

Robin Milner. *A calculus of communicating systems*. Springer-Verlag, 1980.

Robin Milner. *Communicating and mobile systems: the π-calculus*. Springer-Verlag, 1999.

Peter Notebaert. *lpsolve 5.5 reference guide*. -, 2005.

Christos H. Papadimitriou and Kenneth Steiglitz. *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications, 1989.

Dave Parker, Gethin Norman, and Marta Kwiatkowska. *the PRISM manual: user's guide*. University of Birmingham, 3.0 edition, March 2006.

C. Priami, A. Regev, W. Silverman, and E. Shapiro. Application of stochastic name passing calculus to representation and simulation of molecular processes. *Information Processing Letters*, 80:25–31, 2001.

Stephen Ramsey. *Dizzy User Manual*. CompBio Group, Institute for Systems Biology, 1.11.3 edition, April 2006.

A. Regev, W. Silverman, and E. Shapiro. Representation and simulation of biochemical processes using the pi-calculus process algebra. In *Proceedings of the Pacific Symposium of Biocomputing*, pages 459–470, 2001.

Alexander Schrijver. *Theory of Linear and Integer Programming*. John Wiley and Sons, 1998.

Glynn Winskel. *The formal semantics of programming languages: an introduction*. MIT Press, 1993.

Michael W. Young and Steve A. Kay. Time zones: a comparative genetics of circadian clocks. *Nature*, 2:702–715, September 2001.