A GENERIC APPROACH TO BEHAVIOUR-DRIVEN BIOCHEMICAL MODEL CONSTRUCTION

A thesis submitted for the degree of
Doctor of Philosophy

By
Zujian Wu

School of Information Systems, Computing and Mathematics
Brunel University
October 2012

Table of Contents

Table of Contents ii

Abstract x

Acknowledgements xii

1 Introduction 1

1.1 Motivation 1
1.2 Contributions 3
1.3 Publications 5
1.4 Overview of Chapters 5

2 Background of Modelling Biochemical Systems 7

2.1 Introduction 7
2.2 Brief History 8
2.3 Systems Biology and Synthetic Biology 13
2.4 General Modelling Approaches 14
2.4.1 Top-down approach 15
2.4.2 Bottom-up approach 16
2.5 Parameters of Biochemical Models 17
2.6 Representation of Biochemical Systems in Silico 19
2.6.1 Petri Nets 19
2.6.2 SBML 20
2.6.3 P Systems 21
2.7 Modelling Simulators 22
2.7.1 BioNessie 23
2.7.2 Snoopy 24
2.7.3 COPASI 25
2.7.4 CellDesigner 27

2.8 Metaheuristics and Modelling of Biochemical Systems 28
2.8.1 Optimization methods and metaheuristics 28
2.8.2 Simulated annealing 32
2.8.2.1 Principle of simulated annealing 32
2.8.2.2 SA based structure modelling 35
2.8.2.3 SA based kinetic rates optimization 37
2.8.3 Evolution strategy 39
2.8.3.1 Principle of evolution strategy 39
2.8.3.2 ES based structure modelling 41
2.8.3.3 ES based kinetic rates optimization 43
2.9 Summary 45

3 Representation and Composition of Biochemical Systems 47
3.1 Introduction 47
3.2 Enzymatic Reaction 48
3.3 Mass Action Kinetics 50
3.4 Declarations of Component and Model 53
3.4.1 Patterns 53
3.4.2 Syntax of a component 56
3.4.3 Semantics of a component 58
3.4.4 Syntax of a model 59
3.4.5 Semantics of a model 60
3.5 Libraries of Components and Models 61
3.5.1 Components Library 62
3.5.2 Models Library 63
3.6 Modification of Petri Nets 64
3.7 Composition Operators and Rules 66
3.7.1 Addition operator 68
3.7.2 Addition rules 69
3.7.3 Subtraction operator 73
3.7.4 Subtraction rules 74
3.7.5 Crossover operator 77
3.7.6 Crossover rules 79
3.8 Maintenance of Composed Models 81
3.8.1 Maintaining the species 81
3.8.2 Maintaining the components 82
3.8.3 Maintaining the structures 82
3.9 Examples of Composing Biochemical Systems 85

3.9.1 Composition of an enzymatic reaction based on MA1 85
3.9.2 Composition of a 3-cascade pathway without feedback 87
3.9.3 Composition of a 3-cascade pathway with feedback 92
3.10 Summary 94

4 Hybrid Modelling of Biochemical Systems 96
4.1 Introduction 96
4.2 Related Works of Hybrid Modelling 98
4.3 General Framework of Hybrid Modelling 100
4.3.1 General framework 100
4.3.2 Hybrid approach 101
4.4 A 1D Hybrid Modelling Approach 103
4.4.1 Topology generation 104
4.4.1.1 Development of topology 104
4.4.1.2 Experimental results 106
4.4.2 Kinetic rates optimization 111
4.4.2.1 Optimization of kinetic rates 111
4.4.2.2 Experimental results 113
4.4.3 Discussion 117
4.5 A 2D Hybrid Modelling Approach 119
4.5.1 A general flowchart of the hybrid modelling 122
4.5.2 Topology construction based on ES outer layer 124
4.5.3 Kinetic rates optimization based on SA inner layer 126
4.6 Summary 128

5 Variants of Hybrid Modelling Approach 130
5.1 Introduction 130
5.2 A GridGain based Parallelized 2D Hybrid Modelling Approach 132
5.2.1 Motivation 134
5.2.2 Parallelized modelling process 135
5.2.3 Parallel performance 138
5.2.4 Discussion 139
5.3 Evaluation of Composed Models 141
5.3.1 Behaviour comparison 141
5.3.1.1 Average method 142
5.3.1.2 Maximum method 143
5.3.2 Reward and penalty 143
5.3.3 Objective function 144
5.4 Exploration of Topologies Space 145

5.4.1 Construction and evaluation of composed topologies 145
5.4.2 Quantitative evaluation of topologies 146
5.4.2.1 Compression 147
5.4.2.2 Coverage 147
5.5 Variants of Hybrid Piecewise Modelling 148
5.5.1 Methods of driving models composition 150
5.5.1.1 Fixed method 150
5.5.1.2 Dynamic method 151
5.5.2 Methods of selecting survival models 152
5.5.2.1 SES method 152
5.5.2.2 PES method 152
5.5.3 Methods of implementing mutation operators 154
5.5.3.1 Fixed method 154
5.5.3.2 Random method 155
5.5.4 Methods of performing crossover operator 155
5.5.4.1 Best method 155
5.5.4.2 Random method 156
5.5.5 Methods of evaluating models 157
5.5.5.1 ED method 158
5.5.5.2 ED+RP method 159
5.6 Summary 159

6 Cases Study 161
6.1 Introduction 161
6.2 RKIP Pathway 162
6.2.1 Generation of similar behaviour 164
6.2.2 Convergence of composed model fitness 172
6.2.3 Quantitative analysis of composed topologies 174
6.2.3.1 Compression 174
6.2.3.2 Coverage 175
6.2.4 Generation of alternative topologies 176
6.3 Levchenko Pathway 177
6.3.1 Generation of similar behaviour 178
6.3.2 Convergence of composed model fitness 182
6.3.3 Quantitative analysis of composed topologies 184
6.3.3.1 Compression 184
6.3.3.2 Coverage 185
6.3.4 Generation of alternative Topologies 186

6.4 Simulations and statistical analysis on modelling variants 187
6.4.1 Simulation settings 187
6.4.2 Statistical analysis 188
6.4.3 Comparison of modelling variants 191
6.4.3.1 Fixed vs Dynamic - Data driven 191
6.4.3.2 SES vs PES - Survival selection 194
6.4.3.3 Fixed vs Random - Mutation operator 196
6.4.3.4 Best vs Random - Crossover operator 197
6.4.3.5 ED vs ED+RP - Objective function 199
6.4.4 A summary of findings 202
6.5 Summary 203

7 Conclusions and Future Work 204
7.1 Conclusions 204
7.2 Future Work 208

Bibliography 212

A Statistical Analysis of Two-sample Tests in R 234

B A Hybrid Piecewise Modelling Environment 236

List of Figures

2.1 Classification of optimization methods 29

2.2 A genealogy of applications of metaheuristics 30

3.1 An enzymatic reaction based on MA1 51

3.2 An enzymatic reaction based on MA2 52

3.3 An enzymatic reaction based on MA3 53

3.4 Binding pattern 55

3.5 Unbinding pattern 56

3.6 Entity relationship of models and components in database 62

3.7 Addition of a component to a model without modifications 71

3.8 Addition of a component to a model with modifications 72

3.9 Replacement of labels in a component 73

3.10 Creation of a new component 74

3.11 Removal of a component from a model 76

3.12 Removal of a component and linkages of isolated components 77

3.13 Mechanism of *Cut and Splice*. 78

3.14 An original model for subtraction 83

3.15 One structure of the model after subtraction 84

3.16 Another structure of the model after subtraction 85

4.1 A general framework of modelling biochemical systems 101

4.2 A general illustration of hybrid modelling 102

4.3 The *ERK* signalling pathway 107

4.4 Behaviour of all species in *ERK* signaling pathway regulated by *RKIP* . . . 115
4.5 Behaviour of a *RKIP* model from optimization of one kinetic rate 116
4.6 Behaviour of a *RKIP* model from optimization of all kinetic rates 117
4.7 A flowchart of hybridizing ES and SA to model biochemical systems 123

5.1 Implementation of computational grid in GridGain 133
5.2 Sequential and parallelized hybrid modelling processes. 136
5.3 Implementation of GridGain to individual models 137
5.4 Performance of GridGain implementation 139

6.1 A graphical representation of the *RKIP* pathway 163
6.2 Comparison of species *Raf1* behaviour 165
6.3 Comparison of species *RKIP* behaviour 165
6.4 Comparison of species *RKIP*|*Raf1* behaviour 166
6.5 Comparison of species *ERK* behaviour 167
6.6 Comparison of species *MEKPP* behaviour 168
6.7 Comparison of species *ERK*|*MEKPP* behaviour 168
6.8 Comparison of species *RP* behaviour 169
6.9 Comparison of species *RKIPP* behaviour 169
6.10 Comparison of species *RKIPP*|*RP* behaviour 170
6.11 Comparison of species *ERKPP*|*RKIP*|*Raf1* behaviour 171
6.12 Comparison of species *ERKPP* behaviour 171
6.13 Average and five best fitness values of synthetic *RKIP* models 173
6.14 Compression analysis of the synthetic topologies 174
6.15 Coverage analysis of the synthetic topologies 175
6.16 Three signalling cascades of Levchenko2000 178
6.17 Comparison of species *Raf* behaviour 179
6.18 Comparison of species *RafP* behaviour 179
6.19 Comparison of species *RasGTP* behaviour 180
6.20 Comparison of complex *Raf*|*RasGTP* behaviour 180

6.21 Comparison of *Phase3* behaviour 181
6.22 Comparison of species *MEK* behaviour 181
6.23 Comparison of species *MEKP* behaviour 182
6.24 Average and four best fitness values of synthetic Levchenko2000 models . . 183
6.25 Compression analysis of the synthetic topologies 184
6.26 Coverage analysis of the synthetic topologies 185
6.27 Data Driven: Fixed vs Dynamic, Comparison of average fitness of models. . 192
6.28 Data Driven: Fixed vs Dynamic, Coverage and Compression 193
6.29 Survival Selection: SES vs PES, Comparison of average fitness of models . 194
6.30 Survival Selection: SES vs PES, Coverage and Compression 195
6.31 Mutation: Fixed vs Random, Comparison of average fitness of models. . . . 197
6.32 Mutation: Fixed vs Random, Coverage and Compression 198
6.33 Recombination: Best vs Random, Comparison of average fitness of models. 199
6.34 Recombination: Best vs Random, Coverage and Compression 200
6.35 Objective Function: ED vs ED+RP, Comparison of average models fitness. 201
6.36 Fitness Function: ED vs ED+RP, Coverage and Compression 201

B.1 A hybrid piecewise modelling environment for biochemical systems. 237
B.2 A library for preserving instantiated components for composition. 238
B.3 A library for preserving composed models. 238
B.4 An example of generated ODEs illustrating a composed model. 239
B.5 Results of simulating a composed model. 239

Abstract

Modelling of biochemical systems has received considerable attention over the last decade from bioengineering, biochemistry, computer science, and mathematics. This thesis investigates the applications of computational techniques to computational systems biology, for the construction of biochemical models in terms of topology and kinetic rates.

Due to the complexity of biochemical systems, it is natural to construct models representing the biochemical systems incrementally in a piecewise manner. Syntax and semantics of two patterns are defined for the instantiation of components which are extendable, reusable and fundamental building blocks for models composition. We propose and implement a set of genetic operators and composition rules to tackle issues of piecewise composing models from scratch. Quantitative Petri nets are evolved by the genetic operators, and evolutionary process of modelling are guided by the composition rules.

Metaheuristic algorithms are widely applied in BioModel Engineering to support intelligent and heuristic analysis of biochemical systems in terms of structure and kinetic rates. We illustrate parameters of biochemical models based on Biochemical Systems Theory, and then the topology and kinetic rates of the models are manipulated by employing evolution strategy and simulated annealing respectively. A new hybrid modelling framework is proposed and implemented for the models construction. Two heuristic algorithms are performed on two embedded layers in the hybrid framework: an outer layer for topology mutation and an inner layer for rates optimization. Moreover, variants of the hybrid piecewise modelling framework are investigated. Regarding flexibility of these variants, various combinations of evolutionary operators, evaluation criteria and design principles can be taken into account. We examine performance of five sets of the variants on specific aspects of modelling. The comparison of variants is not to explicitly show that one variant clearly outperforms the others, but it provides an indication of considering important features for various aspects of the modelling. Because of the very heavy computational demands, the process of modelling is paralleled by employing a grid environment, GridGain. Application of the GridGain and heuristic algorithms to analyze biological processes can support modelling of biochemical systems in a computational manner, which can also benefit mathematical modelling in computer science and bioengineering.

We apply our proposed modelling framework to model biochemical systems in a hybrid piecewise manner. Modelling variants of the framework are comparatively studied on specific aims of modelling. Simulation results show that our modelling framework can compose synthetic models exhibiting similar species behaviour, generate models with alternative topologies and obtain general knowledge about key modelling features.

Acknowledgements

First and foremost, I would like to thank my supervisor Prof. David Gilbert, for his invaluable support and guidance. His in-depth knowledge, clarity of thought, wealth of ideas, enthusiasm and energy have made working with him an exceptional experience for me. David taught and trained me a lot in systems biology, computer science, biomodel engineering, scientific writing, presentation skills, and many other important aspects of my research career. Moreover, David was a good friend who shared things like happiness, knowledge and culture. He was very happy to observe and enjoy different culture. In addition, I did enjoy lots of wonderful conversations with David about tea, art and life, from which I learn how to balance work and life. I fully enjoy my stay in UK and I very appreciate to have had David as my advisor and mentor.

My grateful thanks also go to Prof. Xiaohui Liu for additional supervision. Thanks for lots of discussions and feedbacks during my research, and great promptly help. Big thanks to Prof. Shengxiang Yang for his important teaching, guidance and help. Without great help from Shengxiang and David, we can not well prepare GECCO2011 and PPSN2012 papers and obtain fruitful results. I must thank Prof. Monika Heiner for her detailed teaching and answers to my questions, helpful comments and suggestions on my research, and patient correction of my writing.

I would like to thank people who have hidden with me in the attic! Many thanks to Dr. Crina Grosan who keeps suggesting my research topics, and promptly provides useful information. Crina is an excellent advisor, a nice cooperator, and a good friend who can share with interesting things. Special thanks to Dr. Huma Lodhi, Dr. Xuan Liu and Mr. Jun Wang. Huma, Xuan and Jun help me a lot not only on my research but also on my life in UK. Many thanks to Qian Gao and Maciej Trybiło who both shared their knowledge and experience with me, no matter how hot or cold the attic was! Moreover, I would like to thank Mrs Teresa Czachowska, Ms Ela Heaney, Ms Susan Standing and other people in the school. Thank them for kind help and creating a friendly and comfortable research environment which is essential for me at Brunel.

Finally, I can never forget the love and support of my family. After my graduation from the high school, no matter how far away from home it would be when I made decisions, mum, dad and my younger brother always support me enough for absolutely everything. They do not travel a lot as I have done because of study in different cities and countries, but they understand it is important for me and just let me have a try, always enjoying my non-stop speeches at home when I come back for short holidays. Many thanks to my mother-in-law and father-in-law who support me as their son by everything they can do. Special thanks to my wife, Huiqin. Your love, encouragement and advice fill my heart and support my life. I am so very lucky and glad to have met you 14 years ago. Thanks to my newborn son, Deming. Your special actions and cute facial expression let me relax after a long day of writing. Without the love of my family, I can not finish my thesis happily.

Zujian Wu
London, UK
October 30, 2012

Chapter 1

Introduction

This chapter introduces the motivation of the research, presents the contributions of the investigation and summarizes the contents of the chapters in this thesis.

1.1 Motivation

Engineering models of biological systems has been investigated recently by employing computational methodologies in BioModel Engineering, for systematically designing, constructing and analyzing characteristics of target biological systems. BioModel Engineering [Brei 10] is inspired by concepts from software engineering and computer science, and it is an interdisciplinary science at the interface of biology, engineering, mathematics and computing science. Intracellular molecular processes have been examined and modelled for explaining observations of the biological systems or predicting behaviour exhibited by the systems.

Systems biology and synthetic biology are two major studied disciplines of BioModel Engineering. In the former, research focuses on the analysis of molecular interactions in biological systems at systematic level, for discovering the 'principles of kinetic laws' which govern biological systems exhibiting behavior. In the latter, study focuses on the design of new biological systems from scratch to obtain specific functionalities.

The aims of synthetic biology are to synthesize biological complex and synthetic systems displaying novel functionalities that do not exist in nature. Synthetic biology invents new biological entities which interact with each other in artificial biological systems consisting of designed properties, by utilizing knowledge of experimental biology. Therefore, it is essential to obtain primary knowledge of biochemical working mechanisms. Systems biology tries to discover biological patterns by systematically analyzing molecular interactions within intracellular environment, especially on metabolic, signalling and gene regulatory networks.

Moreover, because modelling of biological systems in systems biology can be approached by top-down and bottom-up approaches, topologies of models can be built up in alternative structures compared to the experimental ones. In other words, modelling of biological systems in systems biology is able to validate experimental conclusions and discover new biochemical patterns which are important for application of synthetic biology

The motivation of this work is to apply techniques from computer science to develop a methodology enabling the behaviour driven construction of biochemical models in terms of topology and kinetic rates, by intelligently and heuristically reusing components from a user predefined library. The work in this thesis aims to bring the interests of communities of software engineering and mathematics to a multidisciplinary area, ‘intelligently and heuristically modelling biochemical systems in systems biology’, which would gain more and more attentions from academia and industry in near future.

1.2 Contributions

The main contributions of our research can be summarized as follows:

1. We have defined two basic patterns for instantiating extendable and reusable biological components in syntax and semantics, see Chapter 3. In our research, biochemical models under construction are also based on the components instantiated from these patterns. The instantiated components can help improve construction of a foundational bio-bricks library in synthetic biology and systems biology.

2. We have proposed and implemented genetic operators and composition rules for piecewise composing models of biochemical systems, see Chapter 3. Moreover, since components and models manipulated by the operators and rules are presented in Petri nets, our study addresses the evolution of quantitative Petri nets and could thus be applied to stochastic and hybrid Petri nets as well as continuous Petri nets, which can benefit mathematical modelling in engineering, computer science and bioengineering;

3. We have implemented modelling of biochemical systems in a simulated annealing based one dimension hybrid modelling environment in terms of topology and kinetic rates separately, see Chapter 4. A global search mechanism was applied to the processes of piecewise constructing the topology and fine tuning the kinetic rates, driven by target models behaviour. The study of piecewise construction is an implementation of fitting parameters of biochemical models;

4. We have adopted a hybrid approach for the model construction in terms of topology and kinetic rates, and studied variants of the hybrid modelling approach;

   (a) We have proposed a two dimensions hybrid piecewise modelling framework, in which two heuristic algorithms are applied to manipulate topology and kinetic rates of a biochemical model on two switchable layers respectively;

   (b) We have investigated different modelling variants of the hybrid approach, and summarized performance of these variants with the aim of understanding advantage and disadvantage of compared variants focusing on specific modelling aspects, see Chapter 4;

   (c) We have applied the hybrid piecewise modelling framework to model signalling pathways of biochemical systems, see Chapter 6. Simulations results and analysis show it is feasible to apply our modelling framework to assemble alternative models exhibiting similar species behaviour to desired ones in target signalling pathways, and it is possible to perform genetic operators evolving models candidates. In addition, a tradeoff can be approached for switching topology construction and kinetic rates optimization while composing biochemical models.

5. We have parallelized the hybrid piecewise modelling process for improvement of composing models, where topologies and kinetic rates of models under construction can be manipulated in parallel, see Chapter 5;

6. We have developed two extendable components and models libraries in a MySQL database, see Chapter 3. The database is integrated with the hybrid piecewise modelling approach on a platform which is developed by Java programming language with an user-friendly interface, see Appendix B.

1.3 Publications

Parts of this thesis have been summarized and published in peer-reviewed conferences during the course of this thesis.

- Z. Wu, Q. Gao, and D. Gilbert. Target driven biochemical network reconstruction based on petri nets and simulated annealing. In: Proceedings of the 8th International Conference on Computational Methods in Systems Biology, pp. 33-42, ACM, New York, NY, USA, 2010.

- Z. Wu, S. Yang, and D. Gilbert. A hybrid approach to piecewise modelling of biochemical systems. In: C. Coello Coello, V. Cutello, K. Deb, S. Forrest, G. Nicosia, and M. Pavone, Eds., Parallel Problem Solving from Nature - PPSN XII, pp. 519-528, Springer Berlin Heidelberg, 2012.

1.4 Overview of Chapters

This thesis is organized as follows:

Chapter 2 introduces the background of modelling biochemical systems in this study and describes the main aspects of biochemical models with corresponding presentations in silico. We examine modelling issues related to the topology and kinetic rates, and present popular simulators.

Chapter 3 firstly defines binding and unbinding patterns in formal syntax and semantics for instantiation of biological components and composition of models. Two libraries based on a MySQL database technique are designed and implemented to preserve instantiated components and constructed models during the process of models composition. Then, three genetic composition operators and a set of composition rules are proposed and illustrated with demonstration examples. After fine tuning models by the composition operators and rules, manipulated models are studied to ensure generated models in Petri nets are with non-conflicting entities names, connective structures and unique components.

Chapter 4 proposes a modelling framework with different hybrid methodologies. The hybrid modelling framwork has focused on construction of models by manipulating topology or optimizing kinetic rates in an independently or hybrid manner.

Chapter 5 develops introduce a grid technique to the two dimensions hybrid piecewise modelling framework to parallel the modelling process. Modelling variants of the proposed hybrid modelling approach are illustrated. Evaluation of composed models is investigated by including pure Euclidean distance function and a reward and penalty function in an objective function. Exploration of topologies of models generated by our hybrid modelling approaches is examined with quantitative and qualitative methods in this chapter.

Chapter 6 presents the application of our two dimensions hybrid piecewise modelling approach and modelling variants to model biochemical pathways. Simulation results and statistical analysis show that it is feasible to piecewise construct alternative models exhibiting similar species behaviour to the ones of target biochemical systems.

Chapter 7 summarizes the research, draws conclusions from our research and discusses further research ideas raised from this thesis.

Chapter 2

Background of Modelling Biochemical Systems

2.1 Introduction

This chapter introduces the system concept of modelling biochemical systems in Section 2.2. Section 2.3 gives an illustration of the aims and functions in systems biology and synthetic biology which are two major research areas of BioModel Engineering. Two different but complementary modelling strategies, top-down and bottom-up approaches, are illustrated in Section 2.4 with related works of modelling of biochemical systems. In Section 2.5, parameter variables of biochemical systems under investigation are shown by employing the Biochemical Systems Theory which represents the biochemical processes in a mathematical way.

Biochemical systems are represented and investigated widely in the community of computational biology. In Section 2.6, we introduce three well defined and implemented computer based biochemical model formats, Petri Nets, SBML and P Systems, which are in a graphical presentation or a XML based format. We present four popular modelling simulators in communities of systems and synthetic biology for models construction, analysis, optimization and simulations in Section 2.7. All these simulators can work with biochemical models constructed in aforementioned biochemical model formats by import and export functionalities.

In Section 2.8, we present implementation of metaheuristics in modelling of biochemical systems, with a brief introduction of classification and characteristics of different algorithms in the metaheuristics. Since we mainly apply two algorithms, simulated annealing and evolution strategy, to our proposed hybrid modelling framework, the basic principles of these two algorithms are illustrated. Then we review related works of applying the simulated annealing and evolution strategy to develop models structures and to optimize kinetic rates.

Section 2.9 gives a brief summary of the contents of this chapter.

2.2 Brief History

Modelling biochemical systems has been investigated widely in computational biology, especially in systems biology. Constructing models of biochemical systems can be dated back to three academic periods from theory preparation to formation of system concept and development of modelling in systems biology. Details are illustrated as follows.

- **Before 1940s, preparation of theory foundation**

  Since 1854, Claude Bernard used a phrase '*Milieu intérieur*' (*the environment within*) in his works to refer to the extra-cellular fluid environment which is the physiological capacity that provides protective stability for the tissues and organs of multicellular living organisms. Furthermore, Bernard summarized it as following [Bern 74]:

  > The fixity of the milieu supposes a perfection of the organism such that the

> external variations are at each instant compensated for and equilibrated.... All of the vital mechanisms, however varied they may be, have always one goal, to maintain the uniformity of the conditions of life in the internal environment .... The stability of the internal environment is the condition for the free and independent life.

Walter Bradford Cannon developed the idea of *Milieu intérieur* into *Homeostasis* (*mechanistic*) [Cann 32] in his book *The wisdom of the Body* in 1932, and later Cannon described the homeostasis systems as follows [Cann 35]:

> A homeostatic system is an open system that maintains its structure and functions by means of a multiplicity of dynamic equilibriums rigorously controlled by interdependent regulatory mechanisms.

Since the concept of *Milieu intérieur* has been suggested by Bernard, it is possible to obtain the foundation of understanding the internal physiology of cellular and extracellular basic systems. Moreover, dynamics of homeostasis in the communication systems is benefit from the concept of *Milieu intérieur* with its developments.

- **From 1950s to 1980s, formalization of systems concept**

  Systems biology is a new interdisciplinary area in last decade for most biologists, mathematicians, computer scientists and engineers, but the concept of system was used to describe the application of systems and control theory to biology around 1960s. In 1960, the first computer model of the heart pacemaker was presented by Denis Noble [Nobl 60]. Norbert Wiener defined *Cybernetics* [Wien 65] and the mathematical formulation description of physiological systems in 1965. Then, the concepts of cybernetics and negative feedback were introduced into the nervous system and nonliving machines. Later, Ludwig von Bertalanffy tried to construct a general systems theory [Bert 68] in 1968. But the theory was too general and not devised rigorously as a scientific discipline. Moreover, the concepts of robustness and feedback control were already discussed and investigated widely and extensively at that time [Kita 02c].

Complex molecular systems, for instance metabolic control analysis and biochemical systems theory, were studied by employing several approaches from the 1960s to 1970s [Kacs 73, Sava 76]. Quantitative modelling biological processes was achieved by progressing biochemical research throughout the 1980s [DeLi 88, Mora 98]. In 1989, Christopher Langton and other scientists developed theories for living systems by claiming concept of artificial life [Lang 89], but the theories focused on the area of engineering not the biological sciences.

In this period, genetic analysis of biochemical systems in molecular biology developed quickly, with basis of examining functions of compounds at cellular level by utilizing deductive approaches. But interactions and biochemical relationships among components, such as genes and proteins, were not the subjects of scientific research.

- **After 1990s, development of modelling biochemical systems in systems biology**

  Traditional study of genomics has focused on details of static aspects of the genomic information, for instance DNA sequence or structures. After the completion of the whole genome sequencing and implementation of high-throughput measurement technologies [Kita 02c], the community began to study modelling at systematic level. Functional genomics was developed under the framework of molecular biology. Information about functions and interactions among the genes, proteins and other compounds can be obtained from the vast wealth of data produced by genomic projects, for instance the Human Genome Project. Recently, the main subjects under examination among these data are gene transcription, translation and protein-protein interactions.

There are two distinct branches in the study of systems biology: knowledge discovery and simulation-based analysis. The former one abstracts the hidden patterns from huge quantities of experimental data and the latter one tests hypotheses with models in silico experiments [Kita 02b]. Regarding difference between research of static aspects of the genomic information and study of dynamics of functional genomics, more realistic models can be constructed and analyzed by employing high performance computing techniques *in silico* to obtain knowledge from large quantity and high quality data.

Therefore, systems biology has attracted much attention in the scientific community since 1990s, accompanying completion of various genomic projects (such as genome sequencing projects). High-throughput experimental methods also provide great opportunities to investigate these interactions among compounds inside the cells, supporting the rapid development of systems biology. Thus, process inside cells is studied by employing systems biology discipline in post-genomics era, which has been investigated on networks, states, and dynamics [Kita 02a].

General research in systems biology can be particularized into following areas: research of molecular/biochemical/cellular biology, computational studies and software tools, analysis of dynamics of the system, technologies for high-precision and comprehensive measurements. Furthermore, research with system-level understanding in systems biology could be classified into four parts [Kita 02c]:

1. ***System Structure*** mainly involves the network and physical structure of the system. For the network of gene regulation, metabolism and signal transduction, structure study should be on elements, interactions among elements, and parameters related in the system. There were methods of simulation on the network modelling in early research stage, but these methods were of the problems of lacking precise data and knowledge for precision simulation. Above problem was addressed later by the appearance of high-throughput measurements. But problems of structure study still exist, such as information loss and large noisy data for system structure modelling.

2. ***System Behaviour*** could be understood in the analysis of the system from steady state to dynamic state. The number of parameters investigated would affect the known level on the system behaviour.

3. ***System Control*** is employed in system biology after understanding system structure and behaviour. Drugs usage and treatment methods may benefit from system control, for example controlling the drug absorption or physical intervention.

4. ***System Design*** would be the application stage of system biology. It is possible to construct models of biological systems for achieving special aims, such as curing diseases, by investigation of key issues of diseases in the models.

While attempting to reveal working mechanisms in cellular and extracellular environment in biology, it is important to have the system concept. From genomics to post genomics eras, investigated biological research is moved from genomic level to systematic level. Overall investigation of biology can be achieved by modelling biochemical systems in systems biology. The study is supported by the state of the art experimental techniques in wet-lab and analytical simulation tools in dry-lab.

2.3 Systems Biology and Synthetic Biology

Systems biology [Kita 02a, Klip 05, Ferr 09, Vall 10, Joyn 11, Mach 11] and synthetic biology [Benn 05, Andr 06, Hein 06, Mukh 09, Khal 10, Step 12, Voig 12] are two primary application areas of BioModel Engineering. The former one aims to construct and analyze biological models for illustrating observed characteristics of the systems and predicting behaviour of the experimental systems. The latter one attempts to design and create artificial biological systems from scratch for obtaining novel and specific functionalities in these synthetic systems.

In systems biology, computational methodologies and high-throughput experimental data are employed to model biochemical processes, including metabolic pathways, signalling pathways and gene regulatory networks. Applications of systems biology include validation of assumptions of experimental investigations *in vivo* or *in vitro*, analysis of multicellular or intracellular interactions, explanation of biochemical phenomena observed in wet-lab, and prediction of biochemical systems behaviour with regard to biological knowledge. Moreover, discovering of biochemical patterns is crucial in systems biology. Regarding experimental restrictions in wet-lab, principles of governing molecular interactions which support life are very difficult to observe and obtain. Examinations and conclusions of biochemical reactions patterns from application of systems biology can enable researchers to explore functions of biochemical entities within multi/intra-cellular environment, and can support further research in synthetic biology to design artificial biological systems and to approach desired functionalities for specific requirements.

In synthetic biology, biochemical complex of artificial biological systems are synthetic from scratch to generate novel desired functionalities that do not exist in nature. Thus life forms can be engineered with specific aims to sort out concrete problems in our real world, for instance pollution issues in environment protection, energy production and therapy of human disease. Principles of biochemical reactions in biological systems are obtained from experimental investigation (e.g. wet-lab) or computational simulations (e.g. dry-lab). Therefore, different hierarchy of biological systems (such as individual molecules, whole cells, tissues and organisms) can be engineered with guides of the obtained life principles to design ‘artificial life’ in a rational and systematic manner.

Systems biology and synthetic biology focus on different application areas of engineering biological systems, with attempts to validate, obtain and utilize biological knowledge. Although different motivations of studying biological systems exist in these two interdisciplinary subjects, exploration of life patterns in systems biology and utilization of biological principles in synthetic biology, it is essential for both subjects to understand details of biological systems at a systematic level for revealing biochemical principles forming our living world.

2.4 General Modelling Approaches

Information on all individual parts and interactions in biochemical systems is required for systems exhibiting behaviour and functions. Modelling of biochemical systems can be approached by utilizing two separate but complementary strategies: top-down and bottom-up approaches.

The two approaches focus on discovering mechanisms and principles that underlie cell function and formalizing meaningful biological processes in cells. In the top-down approach, a biological cellular system is reduced systematically until essential parts remain in a minimal cellular environment. In the bottom-up approach, a whole or an aspect of a target biological system is composed from components. Therefore, the top-down based computational modelling approach simplifies the biological systems and the bottom-up based modelling approach complexifies the biological prototypical units. Bruggeman et al. provided more details about classification of the top-down and bottom-up approaches, indicating the challenges faced by modelling in systems biology and discussing limitations of these two approaches which have already led to fruitful discoveries [Brug 07].

2.4.1 Top-down approach

In the top-down approach, a large biochemical system is analyzed and decomposed for discovering molecular mechanisms. Then these discovered mechanisms are utilized to determine correlations between concentrations of molecules. Biological assumptions are generated and tested in further biochemical analysis or experiments.

The top-down based studies on cell interactions deal with large datasets and aim to obtain knowledge of biochemical systems behaviour at system level. Discoveries of behavioural patterns can support the prediction of biological mechanisms [Tayl 03, Ihme 04] and functional processes [Tana 04, Beye 06].

With respect to large omics data being ready for implementation of top-down approach, advantages of top-down approach based modelling are completion of analysis at genome level, and biochemical issues (such as metabolome, fluxome, transcriptome and/or proteome) can be also tackled [West 04]. Thus, structures of the molecular networks can also be identified [Khol 02, Vlad 04] and values of parameters in gene networks can be determined [Mole 03, Krem 04], by employing the top-down direction based modelling and analysis of biochemical systems.

2.4.2 Bottom-up approach

In the bottom-up approach, basic components and relevant information (such as kinetic laws of biochemical reactions) are utilized and integrated together from scratch, for discovering biochemical patterns within a whole system. Thus, functional properties of biochemical systems are inferred from individual components and their interactions. The bottom-up approach formulates the interactions among components in a sub-system by indicating the interactive process, for instance enzymatic reactions. Then interactions among components from different sub-systems enable the composed system to exhibit behaviours which are compared and validated with the target ones from experimental data. Therefore, small systems can be composed into a complex whole model for representing an entire biochemical system in a bottom-up based construction manner.

Some concrete biochemical pathways have been studied by employing bottom-up approach in experimental examination: signaling network downstream of the epidermal growth factor receptor [Khol 99, Suen 04, Kiya 06], modelling of central carbon metabolism in Escherichia coli [Krem 01, Schm 04, Bett 06], and Trypanosome brucei [Albe 05].

Regarding difference of resources for modelling of biochemical systems by utilizing bottom-up approach, topologies of models have been taken into account for illustrating concrete stoichiometric structures of biochemical systems. In some research, experimental examination enables precise determination of the kinetic parameters and enzymatic principles for the investigated systems. Moreover, fitting kinetic parameters by the bottom-up approach can be supported with previous modelling investigations in literature review. Therefore, some studies based on a bottom-up approach can be more precise than other approaches on modelling of biochemical systems.

2.5 Parameters of Biochemical Models

In order to study the chemical processes in living organisms, biochemistry is employed to investigate the principals of life. All the living organisms and processes are governed by the laws of biochemistry. Biochemical processes support the complexity of life, by controlling information and energy flow through biochemical signalling and metabolism. Therefore, the structures, functions and interactions of cellular components are studied in biochemistry. Furthermore, biochemical processes are main research targets, rather than individual molecules such as proteins, carbohydrates, nucleic acids and other biochemical entities.

Mike Savageau developed biochemical systems theory (BST) in the late 60s for mathematical modelling of biochemical systems, based on ordinary differential equations (ODE), in which biochemical processes are represented using power-law expansions in variables of the system [Sava 69a, Sava 69b, Sava 70]. One of major advantages of implementation of the BST is that a set of equations can be set up without knowledge of exact mechanism of each reaction in the model; moreover, biochemical models can be designed after identifying the reactants with corresponding reactional and regulatory interactions.

Models of biochemical systems are composed from interacting species, whose dynamic evolution is determined by the occurrence of biochemical reactions. Species investigated in this thesis are the protein or protein complex which work as reactants involved in biochemical reactions. A complex is grouped molecular species, such as a product of a protein binding to an enzyme which is also a protein. A biochemical model is fully characterized by the initial amount of each molecular species X_i $(1 \geq i \geq n)$ and the description of the biochemical reactions r_j $(1 \geq i \geq m)$ with their kinetic rate laws [Ball 10]. In biochemical models, the production or consumption of reactants are described by the biochemical reactions, presenting the regulations among these reactants. Biochemical reactions involve zero or more molecular species, while the species can be either reactants or products. Stoichiometric coefficients associated with biochemical reactions specify the number of molecules that are consumed or produced for each molecular species involved in the reactions.

Parameters of a biochemical model can be introduced in general by utilizing a definition of dynamics of an involved species in the model. The representation of the dynamics is given by a differential equation as follows.

$$\frac{dX_i}{dt} = \Sigma_j \mu_{ij} \cdot \gamma_j \Pi_k X_k^{f_{jk}} \tag{2.5.1}$$

where X_i represents one species of the model, for instance metabolite concentrations, protein concentrations or levels of gene expression; j represents the biochemical reaction affecting the dynamics of the species; μ_{ij} indicates the stoichiometric coefficient; γ_j indicates rate constants; and f_{jk} stands for kinetic orders.

Models representing power-law based biochemical models are different from other ODE models. In power-law models, kinetic orders can be non-integer and negative values. For instance, if there is an inhibition, a negative kinetic order indicates the inhibition on the dynamics of species by other species. Thus, power-law based models are much more flexible than other types of models for reproduction of non-linearity of the biochemical models;

and recently different kinds of biochemical models (metabolic pathways, signalling pathways and gene regulatory networks) are modelled by employing power-law expansions. Mass-action kinetics and Michaelis Menten kinetics are two widely used power-law kinetics: Mass-action kinetics takes kinetic reaction rate as a proportional value to the amounts of reactant and a kinetic constant; whereas Michaelis Menten kinetics relates the rate of enzymatic reactions to the concentration of a substrate in a model. But it should note that the Michaelis Menten kinetics only holds at the initial stage of a reaction before the concentration of the product is appreciable [Brei 08].

Parameters defined in Equation 2.5.1 are dynamic variables which enable biochemical models exhibiting behaviour (dynamics of involved species). In this thesis, we are interested in applying computational methodologies to approach and optimize these parameters by performing evolutionary modelling of biochemical systems. We take topology and kinetic rates of a biochemical model to be target investigated parameters, on which our proposed hybrid modelling framework works.

2.6 Representation of Biochemical Systems in Silico

There are different methodologies employed to describe biochemical systems in computational biology. In this chapter, we briefly introduce several popular mathematical methodologies in communities of systems and synthetic biology for illustrating biochemical processes in cellular environment.

2.6.1 Petri Nets

Preliminary qualitative and quantitative analysis of biochemical systems have been very difficult to be approached, due to inherited complexity of biochemical process. Petri nets theory [Mura 89] has been proposed for modelling biochemical systems, for instance metabolic pathways (including enzymic cascades and synergistic binding of ligands to enzymes).

Michael C. Kohn and William J. Letzkus applied the graph-theory Petri nets to illustrate a model of glycogen metabolism in 1983, by implementing formal operations on a graph of given network which leads to the identification of feedback metabolites and enzymes regulating the feedback. The systemic properties are thus isolated from the purely local regulation of individual enzymes [Kohn 83]. Venkatramana N. Reddy and other researchers focused on tackling problems of quantitative analysis of metabolic pathways [Redd 93, Redd 96] in the 1990s. Research of applying Petri nets to represent biochemical processes and indication of current research difficulties of constructing biochemical pathways by Petri nets can be referred to [Pele 05, Mats 06, Chao 07, Bald 10].

Moreover, many extensions of Petri nets, for instance coloured, timed, stochastic, continuous, hybrid, hierarchical, functional Petri nets, have been developed and applied to different scientific disciplines for both qualitative and quantitative analysis. Regarding the versatility of different Petri nets extensions, the Petri nets based modelling formalism has been utilized for modelling of biochemical systems in three types of pathways [**?**]: metabolic pathways [Kffn 00, Zeve 03, Koch 05], signaling networks [Sack 06, Chen 07, Brei 08, Hard 08]; and gene regulatory networks [Chao 04, Chao 08].

These primary research and achievements present recent implementation of Petri nets to model biochemical systems, including formal description of constructed models in Petri nets and corresponding extensions formats. The Petri nets methodology is one of the graphical theories to illustrate and model biochemical processes, and in this thesis we also focus on the utilization of Petri nets in our hybrid modelling framework.

2.6.2 SBML

Regarding reality of generating computational models of biological systems via vast and expanding quantities of data, we can employ computable file formats to present these models of biological systems. Systems Biology Markup Language (SBML) is a free and open interchange format for computer models of biological processes [SBML 12].

More standard, formal, and computable representations of biological models are required for achieving the aims of rigorously analyzing and computationally simulating biochemical processes with mathematical methods. For instance, a graphical diagram is useful to visualize and illustrate the biological relationships among entities in a model, but it is difficult to quantify the model to a computer based simulation and analysis environment. SBML is proposed and applied to tackle issues of mathematical analysis and simulation of the biological processes in silico.

In summary, SBML is a machine-readable format XML-like annotation language for representing biological models. Biological processes and entities involved in biological systems can be described by employing SBML which is suitable for representing models of cellular metabolic pathways, signaling pathways, and gene regulation networks. Details about normative definitions of features of SBML can be referred to most recent SBML specification document *SBML Level 3 Version 1 Core* [Huck 10].

2.6.3 P Systems

One of the computational models in community of computer science is a P system introduced by Gheorghe Păun [Paun 98, Paun 99, Paun 00]. The P systems perform calculations by utilizing a biologically inspired process, which is based on the structure of biological cells from the way in which chemicals interact and cross cell membranes. Furthermore, variations on the P systems led to formation of a research branch 'membrane computing'.

P systems have been primarily employed to study modelling issues by focusing on computational model characteristics, but later it was also applied to investigate modelling of biochemical systems [Arde 03, Paun 06, Gheo 08, Rome 09, Blak 11]. While being applied to model biochemical systems, a P system model is defined by using a set of membranes which contain biochemical entities and rules. These entities in a P system model determine the processes which the entities in the model may react with one another to form other products. Rules may also cause biochemical entities to pass through membranes or even cause membranes to dissolve.

Moreover, in a cellular environment, a biochemical reaction may only take place while required molecules collide and interact in a random manner. Thus rules in a P system model are implemented randomly, which results in a stochastic computation in the model and multiple simulation results being obtained in a repeated computing process. Computation in a P system model stops at a state in which no more reactions are enable. Therefore, results of a P system based simulation illustrate a biochemical process that all entities are passed to outside of the outermost membrane or into a specific membrane.

2.7 Modelling Simulators

There are different kinds of software environments developed for modelling, analyzing and simulating biochemical systems in the community of computational biology. Although different modelling simulators employ different model formats for representing biochemical systems and analyzing biochemical interactions in the models, most of these modelling simulators support importation and exportation of models under examination among different formats, for instance a SBML based model file can be imported and exported for simulation in a simulator, Snoopy, by its own model format.

In this section, we specifically focus on introduction of several popular and powerful modelling simulators for constructing models of biochemical systems, fitting kinetic rates and predicting compounds behaviour in a continuous/stochastic and qualitative/quantitative manner.

2.7.1 BioNessie

BioNessie [Liu 08] is a free, state-of-the-art platform-independent biochemical networks simulation and analysis software environment. It is developed by using Java technology and can be run on many platforms that support Java Runtime Environment (JRE) 1.5 or higher.

A full user-friendly Graphical User Interface (GUI) is provided to allow users to import, create, edit and export the biochemical models with the SBML standard. The unique Concurrent Versions System (CVS) design helps users to keep track of the version history of their SBML models during construction and subsequent modification. The core of BioNessie comprises the SOSlib (SBML ODE Solver library), which provides a programming library for symbolic and numerical analysis of a system of ordinary differential equations derived from a chemical reaction network encoded in SBML format. BioNessie can generate the changes of species amounts and parameter values over time by simulating the SBML model numerically with SOSlib. The simulation results can be generated in many ways: raw data files, plots, xml files and report text files. BioNessie is not only an editor and simulator, but also an analyzer, supporting multiple functions such as:

- Multi-thread/core enabled parameter scans
- Sensitivity analysis
- Parameter estimation (model fitting)

Cooperating with National e-Science Centre at Glasgow on the project 'BioNessieG', benefits are obtained from a wide variety of high performance computing resources across the UK through Grid technologies to support larger scale biochemical simulations in BioNessie.

2.7.2 Snoopy

Snoopy [Rohr 10, Blat, Marw 12, Liu 12] is a software tool to design and animate hierarchical graphs, among others Petri nets. The tool has been developed for using Petri nets as a common communication platform for experimentalists and theoreticians. Moreover, Snoopy is also a unifying framework for the graphical display, computational modelling, simulation, and bioinformatic annotation of biochemical networks, such as bacterial regulatory networks. Main features available in Snoopy are shown as following:

- Hierarchies by subgraphs
- Logical (fusion) nodes

- Different shapes for net elements
- Colouring of graph elements (e.g. paths or invariants)
- Automated layout by Graphviz library
- Digital signature by md5 hash function
- Animation of place/transition Petri nets
- Simulation of stochastic/continuous Petri nets
- Printing support: eps, Xfig, FrameMaker
- Import/export from/to analysis tools
- SBML import/export
- Support of web-based Petri net animation

Snoopy is in use for the verification of technical systems, especially software-based systems, as well as for the validation of biochemical systems. It is used for the design and animation of hierarchical graphs of biomolecular networks. It supports different kinds of Petri nets, and incorporates the exact Gillespie algorithm for stochastic nets and a variety of ODE solvers for continuous nets.

2.7.3 COPASI

COPASI [Hoop 06] is a software application for simulation and analysis of biochemical networks and their dynamics. It is a stand-alone program that supports models in the SBML standard and can simulate their behavior using ODEs or Gillespie's stochastic simulation algorithm. Moreover, arbitrary discrete events can be included in the simulations. A list of features in COPASI is given as following:

- Models construction
    - Chemical reaction network
    - Arbitrary kinetic functions
    - ODEs for compartments, species, and global quantities
    - Assignments for compartments, species, and global quantities
    - Initial assignments for compartments, species, and global quantities
    - SBML import and export
- Models analysis
    - Stochastic and deterministic time course simulation
    - Steady state analysis (including stability)
    - Metabolic control analysis/sensitivity analysis
    - Elementary mode analysis
    - Mass conservation analysis
    - Time scale separation analysis
    - Calculation of Lyapunov exponents
    - Parameter scans
    - Optimization of arbitrary objective functions
    - Parameter estimation using data from time course and/or steady state experiments simultaneously

- Graphical User Interface (CopasiUI)
  - Sliders for interactive parameter changes
  - Color-coded tables
  - 3D bar charts
  - Plots and Histograms
  - Network diagram visualization of results
- Command Line (CopasiSE) for batch processing
- Versions for MS Windows, Linux, Mac OS X, and Solaris SPARC
- Loading of legacy Gepasi files
- Export to Berkeley Madonna, XPPAUT, and C source code of the ODE system generated from the model
- Saving of mathematical formulas and ODEs in MathML or LaTeX

COPASI carries out analysis of the network and its dynamics, and it has extensive support for parameter estimation and optimization. It also provides means to visualize data in customizable plots, histograms and animations of network diagrams. Details about utilization of COPASI for modelling biochemical systems are given in works by Sahle, Mendes and other researchers [Sahl 06, Mend 09a, Mend 09b].

2.7.4 CellDesigner

CellDesigner [Funa 03, Funa 08] is a structured diagram editor for drawing gene regulatory and biochemical networks. Networks are drawn based on the process diagram, with graphical notation system proposed by Kitano [Kita 05], and are preserved using the SBML standard for representing models of gene regulatory and biochemical networks. Moreover, networks are able to link with simulation and other analysis packages through Systems Biology Workbench (SBW). Major features in CellDesigner are summarized as follows:

- Biochemical gene regulatory networks modeling with GUI
- Visual representation of biochemical semantics
- Comprehensive graphical notation: SBGN process diagram
- SBML compliant
- Direct integration with SBML ODE solver and Copasi
- Smooth linkage to SBW-powered simulation module
- Database connections
- Export image to image files including PDF and SVG format

CellDesigner supports simulation and parameter scan by an integration with a SBML ODE solver and Copasi. By using CellDesigner, users can browse and modify existing SBML models with reference to biochemical models databases, simulate and view the dynamics through an intuitive graphical interface.

2.8 Metaheuristics and Modelling of Biochemical Systems

2.8.1 Optimization methods and metaheuristics

Optimization methods are employed widely to formulate and solve optimization problems in science and engineering, especially the application of metaheuristics to modelling problems in biology in the last decade. Talbi introduced the details of optimization methods and summarized the classifications of theses optimization methods [Talb 09]. We briefly introduce the background of optimization methods, before discussing the implementation of metaheuristics.

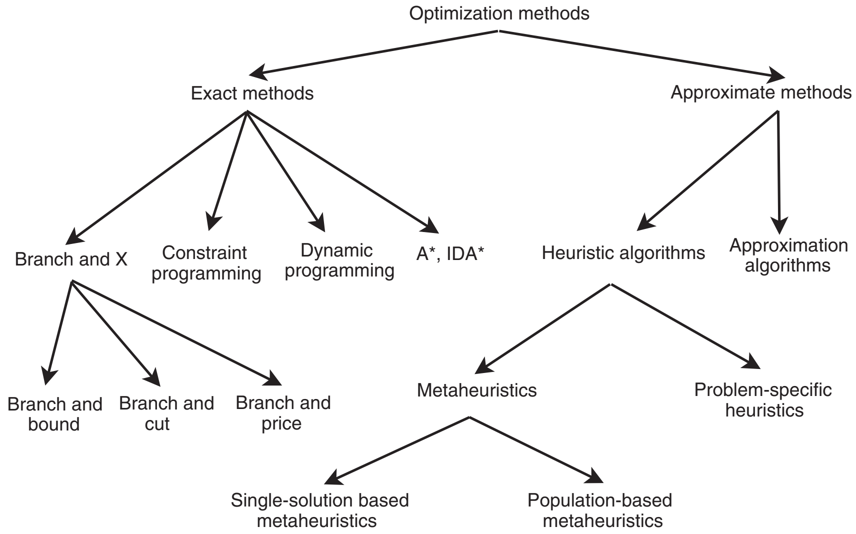


Figure 2.1: Classification of optimization methods, generated by Talbi [Talb 09].

Figure 2.1 shows the diversity of classical optimization methods which are summarized and divided into two categories: exact methods and approximate methods. The exact methods can obtain optimal solutions and guarantee their optimality, and approximate methods generate high quality solutions in a reasonable time for practical use, but there is no guarantee of finding a global optimal solution [Talb 09].

Moreover, the approximate methods can be summarized to heuristic algorithms (reasonably approaching ‘good’ problem solutions in a reasonable time) and approximation algorithms (offering problem solutions with provable quality and run-time bounds). Metaheuristics and problem-specific heuristics are two classes of the heuristic algorithms. Specific problems are addressed by the problem-specific heuristics which are tailored and designed for optimization constraints. Metaheuristics are general strategies which can be utilized to tackle optimization problems.

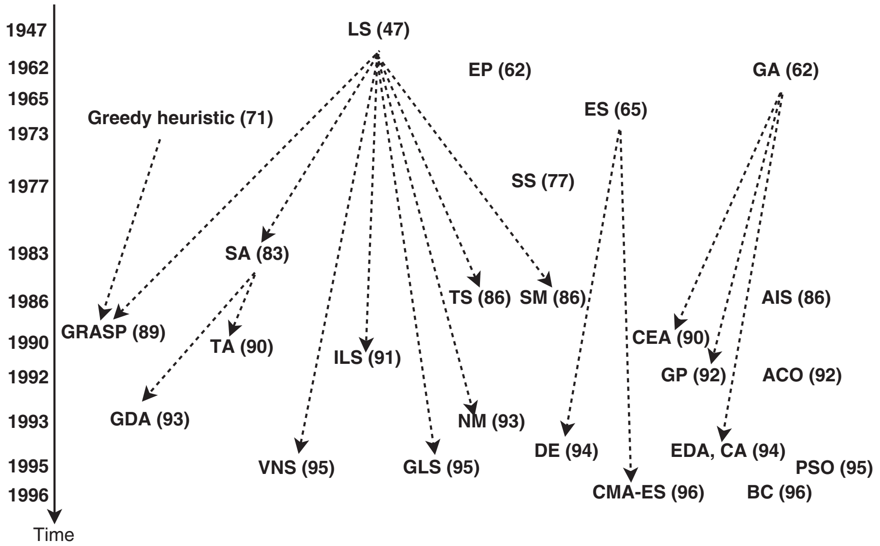


Figure 2.2: Talbi [Talb 09] summarized a genealogy of applications of the metaheuristics. Algorithms are listed by using abbreviations. Numbers in brackets are years of original applications of the algorithms. Arrows with dash lines indicate genealogical relationships among the algorithms.

There are numerous metaheuristics proposed and implemented to address practical optimization and/or machine learning problems. Classical metaheuristics include simulated annealing, tabu search, evolutionary algorithms (EAs), ant colony optimization, estimation of distribution algorithms, scatter search, path relinking, greedy randomized adaptive search procedure (GRASP), multi-start and iterated local search (ILS), guided local search, and variable neighborhood search (VNS), which have individual historical backgrounds and follow different paradigms and philosophies [Loza 10]. Figure 2.2 shows a genealogy of original applications of the metaheuristics which is summarized by Talbi [Talb 09].

Moreover, metaheuristics can be classified by criteria, such as the natural/nonnatural inspiration, with/without memory requirement, deterministic/stochastic decision process, population/single-solution based search, and iterative/greedy search process. Details of these criteria can be found as follows.

- Natural metaheuristics - being inspired from biology, swarm intelligence and physics
- Memoryless metaheuristics - not using information preserved during the search
- Deterministic metaheuristics - solving optimization problems by making deterministic decisions
- Stochastic metaheuristics - applying random rules to search process
- Population based metaheuristics - evolving a set of solutions
- Single-solution based metaheuristics - manipulating a single solution in search process
- Iterative metaheuristics - starting from non-empty complete solution(s) and transforming solution(s) at each iteration by search operators
- Greedy metaheuristics - starting from an empty solution and making a decision at each step, until generation of a complete solution

Table 2.1 presents a classification of metaheuristics which are divided by different criteria. It should note that each family of metaheuristics actually shares many search mechanisms during optimization process, therefore classification of the metaheuristics based on criteria is a demonstration of algorithms characteristics.

Table 2.1: A classification of the metaheuristics by different criteria.

MHs		Criteria								
		Nat.	*Mem.*	*Memles.*	*Det.*	*Sto.*	*Pop.*	*SinSol.*	*Ite.*	*Gre.*
EAs	DE	●				●	●		●	
	ES	●				●	●		●	
	EP	●				●	●		●	
	GA	●				●	●		●	
	GP	●				●	●		●	
AIS		●								
ACO		●								
BC		●								
PSO		●					●			
SA		●		●		●		●	●	
ILS				●	●			●		
GRASP				●						●
TS			●		●					●

Some of most used metaheuristics algorithms for modelling of biochemical systems are given as following:

- Evolutionary algorithms
    - Differential evolution
    - Genetic algorithm
    - Genetic programming
    - Evolution strategy
    - Evolution programming

- Simulated annealing
- Tabu search

As reported in the literature, summarized algorithms have been used to improve the general efficiency and precision of modelling biochemical systems in terms of topology and kinetic rates. Simulated annealing and evolution strategy are two algorithms mainly employed for our proposed hybrid modelling framework in this thesis. Section 4.5.3 and Section 4.5.2 present the details of working mechanisms and applications of SA and ES in modelling of biochemical systems.

2.8.2 Simulated annealing

2.8.2.1 Principle of simulated annealing

Simulated annealing (SA) is one of the physically inspired, memoryless, stochastic, single-solution based and iterative metaheuristics. The SA algorithm was firstly described by Kirkpatrick et al. in 1983 [Kirk 83], and it has been employed widely for addressing optimization problems with/without constraint. By analogy with a physical process of annealing in metallurgy, SA algorithm models the process of heating and lowering the system temperature iteratively to reduce the system defects and to let the system reach a minimum energy status.

In application of SA to search optimum solutions, a new solution point is generated randomly from current solution point at each step. The new solution point is estimated by an objective function and accepted to replace current solution point by an acceptance probability. The acceptance probability involves fitness of evaluated solution point and current system temperature. For a minimization problem, a better or ‘downhill’ solution point is selected in a random manner when the temperature is high; when the temperature going down, the point is selected in a strict manner. Acceptance of an ‘uphill’ solution point during search process lets SA algorithm avoid being trapped in local minima and be able to globally explore potential solutions in a large solutions space. This probability based search procedure is repeated iteratively, and it stops until stopping criterion reached:

1. Fitness of solution candidates converges to a satisfied range;

2. There is no improvement in fitness after consequent generations;

3. System reaches minimum temperature.

Here we give a high level description of SA mechanism in Algorithm 1, and details of applying SA to model or optimize biochemical systems in terms of topology and kinetic rates are illustrated in Section 4.4.

```
Require: Optimization Problem, Starting Solution Point, Objective Function, and
  Parameters of SA
Ensure: Optimized Solution
  while Stopping Criterion Not Reached do
    while Iterations Not Finished do
      Generate(New Solution Point);
      Estimate(New Solution Point);
      Accept(New Solution Point);
    end while
    Reset(Iterations);
    Check(Stopping Criterion);
  end while
  Return Optimized Solution.
```

Algorithm 1: High level description of simulated annealing algorithm.

Given ‘Optimization Problem’, ‘Starting Solution Point’, ‘Objective Function’ and ‘Parameters of SA’, SA algorithm starts the global optimization procedure from the ‘Starting

Solution Point' for creating a new neighbor solution by a random way in 'Generate()' function at each iteration. The new solution point is evaluated by a 'Estimate()' function which employs the 'Objective Function' according to the 'Optimization Problem'. Decision of accepting the new solution point is based on estimated fitness and system probability in an 'Accept()' function: if the new solution point is better than current starting solution point, the starting solution point is replaced by the new solution point; if it is not better but there is a probability allowing the system to accept a worse solution, the starting solution point is still replaced by the worse new solution point for next solution search. Anily and Federgruen have discussed the details of general probabilistic acceptance of SA [Anil 87]. Iterations number will be reset for next round of solution search at different system temperature, and system stopping criterion is checked for stopping global search to return optimized solutions for given optimization problems.

2.8.2.2 SA based structure modelling

Optimization methodologies based on SA are effective for reverse engineering problems in bioengineering. SA has been used to optimize structures of models representing biochemical systems from experimental data. For instance, gene regulatory networks can be coded in SA using an adjacency matrix to represent relationships among genes. Interactions between two genes can be illustrated by an edge with weight values, which is preserved in the adjacency matrix. We briefly review some of research employing SA to study models structures of biochemical systems.

Blower et al. [Blow 02] used simulated annealing and recursive partitioning to find combinations of molecular descriptors. Process of search based on SA was incorporated into a recursive partitioning design to produce a regression tree for biological activity on the space of structural fingerprints. Using LeadScope structural features as descriptors to mine a biological database, the merging of Recursive Partitioning and SA consistently identifies structurally homogeneous classes of highly potent anticancer agents.

Wang et al. [Wang 04] proposed a two-level simulated annealing (TLSA) to explore problems of inferring Bayesian structures which was employed to study gene regulatory networks. Aiming to find global optimized probability network models, the proposed TLSA algorithm globally searches 'Golden Networks' to generate simulated data sets and test Bayesian scores for inferring the strength of learning network structures. Case study shows that the TLSA can reach better structures with lower score, although no ordering information is available in advance. Furthermore, equivalent pattern of the optimized structures are more likely approached by the TLSA optimization algorithm.

In order to visualize automatically the topological architectures and facilitate understanding of functions of complex biochemical networks, Li and Kurata [Li 05] proposed a layout algorithm to draw the networks which are modelled as a system of interacting nodes on squared grids. The layouts of networks are produced by minimizing total cost generated from a discrete cost function between each pair of nodes. A fast algorithm involving simulated annealing heuristics is designed and implemented to minimize the discrete cost function, by which candidate layouts can be produced efficiently, and better candidates can be chosen to exhibit cluster structures clearly in relatively compact layout areas without any prior knowledge.

Guimerà and Amaral [Guim 05] proposed a methodology to extract and display information contained in complex networks. Specifically, functional modules in complex networks can be found by employing simulated annealing to maximum modularity of networks. Nodes can be classified into universal roles according to their pattern of intra-module and inter-module connections. The proposed method yields a ‘cartographic representation of the complex networks. Moreover, Guimerà et al. [Guim 07] investigated how to map the interactions between proteins and metabolites onto complex networks, and how to group nodes and links in complex biochemical networks into a small number of classes by using SA algorithm. Methodology based on SA explores partition of networks into modules that maximizes the modularity, and assess significance of the modular structure of each network for specifying essential and specific metabolic networks.

Rodrigo et al. [Rodr 07b] proposed a new tool to design transcriptional networks with targeted behavior that could be used to better understand the design principles of genetic circuits. SA optimization algorithm is implemented for exploring throughout the space of transcription networks to obtain a specific behaviour. An output transcriptional network with all the corresponding kinetic parameters is described in SBML format.

Ruz and Goles [Ruz 10] proposed a SA based framework with three simple neighborhood search strategies to learn gene regulatory networks with predefined attractors, under the threshold Boolean network model updated sequentially. The robustness of the networks is studied by employing the presented SA method for measuring the number of different updating sequences they can have without loosing the attractor. A power law between the frequency of the networks and the number of the sequences is obtained, as well as a decreasing robustness of the networks while the cycle length growing.

2.8.2.3 SA based kinetic rates optimization

The SA has been applied successfully in computational biology to estimate the parameters of constructed models representing biochemical systems.

Braun et al. [Brau 05] proposed a simple statistical parameter fitting algorithm and test the efficacy of the algorithm by using two synthetic gene networks as cases study. After measuring the deviation between experimental and simulated data by a cost function, an adaptive simulated annealing (ASA) algorithm is employed to minimize the cost function. Because the measured cost is dependent on the set of kinetic parameters for the system, parameter set returned from the minimum cost function fits the model most closely with the experimental data. With respect to well constrained systems, while the value of the cost function approaches zero, the kinetic parameter estimations should ideally approach the actual biological parameters. Therefore, parameter estimation approach based on SA methodology is feasible to recover kinetic parameter values reasonably well for highly constrained gene networks.

The ASA algorithm is also employed by Dunlop et al. [Dunl 07] in an identification framework to estimate parameters of each candidate model for multi-model selection. The ASA algorithm based parameter estimation process is integrated with model comparison process in the identification framework, which determines a best model from a set of given candidate models for well describing experimental data.

Tomshine and Kaznessis [Toms 06] presented an optimization method based on SA to locate combinations of kinetic parameters that produce a desired behavior in a genetic network. Due to inherently stochastic process of the gene expression, simulation component of SA optimization is conducted using an accurate multiscale simulation algorithm to calculate an ensemble of network trajectories at each iteration. After applying the proposed method to a three-gene repressilator, it is shown that gene network optimization is conducted by using a mechanistically realistic model integrated stochastically. Moreover, the repressilator is optimized to give oscillations of an arbitrary specified period.

Gonzalez et al. [Gonz 07] described how SA algorithm with an appropriately constructed perturbation function can be used effectively to estimate the parameters of biochemical networks modeled as S-systems from time-course biochemical data. In order to demonstrate the efficacy and general applicability of the metaheuristics, a proposed SA method is tested by studying three artificial networks designed to simulate different network topologies and behaviour, and the SA method is applied to a real-world problem by creating a working model for the *cadBA* system in *Escherichia coli*.

A mass action model of immediate-early signaling involving *ErbB14* receptors, *MAPK* and *PI3K/Akt* cascades, was constructed and analyzed by Chen et al. [Chen 09] for quantifying signal flow through *ErbB*-activated pathways. By restricting the search to a subset of 75 rate constants and initial conditions with the greatest impact on an objective function, SA is employed to search across a region of parameter space with a set of ODEs. Convergence of parameter optimization is improved substantially. Strong dependence of parameter sensitivity is found on the feature or condition under examination, which is informative with respect to mechanisms of signal propagation.

Cirit et al. [Ciri 10] presented how to modify the standard SA algorithm to generate a large ensemble of ‘good’ parameter sets rather than one ‘best’ fit. Therefore, it is feasible to obtain kinetic models of signaling networks trained on a sufficient diversity of quantitative data, which can be reasonably comprehensive, accurate, and predictive in a dynamical sense.

Czeizler et al. [Czei 11] performed parameter estimation procedures to fit both the values of the kinetic rates and initial concentrations of metabolites in an existing well validated computational model for a heat shock response. The model for the heat shock response is incorporated with several (de)phosphorylation pathways, and the quantitative control of the pathways is analyzed over entire process in terms of parameter estimation by using SA algorithm in COPASI software package.

2.8.3 Evolution strategy

2.8.3.1 Principle of evolution strategy

Evolution strategy (ES) is one of the naturally inspired, memory, stochastic, population-solution based and iterative metaheuristics, which was founded firstly by Rechenberg and Schwefel at the Technical University of Berlin [Rech 65, Rech 73, Schw 65, Schw 75].

Natural selection principle is imitated in ES simulation process by simulating ‘mutations’ and ‘survival’ of individuals in nature. Moreover, ES follows two general rules for driving individuals to achieve optimum status [Beye 02]:

1. All variables are changed at a time in a mostly small and random manner;
2. New generation of modified variables with goodness are kept, otherwise old status of the variables are rolled back as starting points for performing next modifications on the variables.

In general, there are two forms of ES: two-membered (2-m) and multimembered (m-m) ES. The difference between these two forms of ES is the number of parental and children members and corresponding selection schemes for generating new individuals. Table 2.2

briefly illustrates and compares variants of two ES forms. The symbols μ and λ in the two forms of ES stand for the number of parents and children respectively.

Table 2.2: Two-membered and multimembered forms of ES.

Form	Versions	Selection Scheme
2-m	(1+1)-ES	An offspring is selected from two parental and children individuals
m-m	(μ+1)-ES	Two of μ parental individuals at a time are selected randomly and recombined to generate an offspring by discarding the worst one
m-m	(μ+λ)-ES	$\lambda \geq 1$ children individuals are generated in a generation, and μ best out of all μ+λ individuals are selected as offspring
m-m	(μ,λ)-ES	$\lambda \geq \mu$ children individuals are generated in a generation, and selection of μ offspring is taken place among λ children individuals only, without considering fitness of μ parental individuals

```
Require: Optimization Problem, Seeds, Objective Function, and Parameters of ES
Ensure: Optimized Seeds
  Initiate(Seeds);
  while Maximum Generations Not Reached do
    while Number of Children Individuals $\lambda$ Not Reached do
      Recombine(Parental Individuals);
      Mutate(Recombined Parental Individuals);
    end while
    if ($\mu$+$\lambda$)-ES then
      Offspring = Select($\mu$+$\lambda$ Individuals);
    end if
    if (($\mu$,$\lambda$)-ES then
      Offspring = Select($\lambda$ Individuals);
    end if
  end while
  Return Optimized Solution.
```

Algorithm 2: High level description of the evolution strategy algorithm.

ES is one of the global optimization methods and is similar to other evolutionary algorithms, for instance genetic algorithms (GA), genetic programming (GP) and evolutionary programming (EP). But ES works in continuous space with an additional capability of self-adaption on the strategy parameters. A high level description of ES working process is given in Algorithm 2, which indicates the process of offspring generation by different selection scheme. More details of the self-adaptation, robustness and parallelization of ES have been presented by Bäck and Hoffmeister [Back 94]. Moles et al. [Mole 03] presented an extensive review of applying evolutionary algorithms, particularly ES, to reverse engineering regulatory networks, which indicated the outperforms of evolutionary algorithms than other methods on optimization of biochemical models. Chou and Voit [Chou 09] summarized more recent developments in parameter estimation and structure identification of biochemical and genomic systems. In Section 4.5, we illustrate the details of applying ES to optimize biochemical systems in our hybrid modelling framework.

2.8.3.2 ES based structure modelling

ES has been applied to study optimization problems in computational biology, for reverse engineering issues in terms of system structure. We briefly present some research employing ES to study the topologies of biochemical systems.

Streichert et al. [Stre 04] compared two evolutionary algorithms (genetic programming and ES) on inferring gene regulatory networks, with respect to algorithms performance on multiple problem instances with varying parameters. They found that inferring gene regulatory networks can be solved by means of ES, by fixing the network model a priori and reduce the inferring problem to a parameter optimization problem. Results of comparison shown that single problem instances are not sufficient to prove the effectiveness of a given inferring strategy and that the GP approach is less prone to varying instances than the ES.

Cao et al. [Cao 10] proposed a methodology for the automated design of cell models for systems and synthetic biology. The modelling framework was based on P systems, and a model represented by the P systems was discrete, stochastic and modular formal file standard. The automated design of biological models comprised the optimization of the model structure and its stochastic kinetic constants. Optimization was performed using an evolutionary algorithm which evolves model structures by combining different modules taken from a predefined module library and then it fine-tunes the associated stochastic kinetic constants. Four alternative objective functions for the fitness calculation within the evolutionary algorithm were investigated, namely equally weighted sum method, normalization method, randomly weighted sum method, and equally weighted product method. The effectiveness of the methodology was tested on four case studies of increasing complexity including negative and positive autoregulation as well as two gene regulatory networks.

Thomas and Jin [Thom 12] studied issues of how to couple two simple regulatory motifs, one toggle switch and one self-sustained oscillator, using an evolutionary algorithm. They evolved several complex dynamics for two different connections arrangements between the oscillator and toggle switch networks in a master/slave set up, which confirms the previously reported results achieved manually. Results indicate that it is feasible and efficient to generate complex dynamics by coupling of simple motifs using simulated evolutionary mechanisms.

Biological morphogenetic networks, such as gene regulatory networks (GRNs), are modular with independent units and often show the reuse of recurring patterns termed network motifs. Inspired by biological morphogenesis and evolution and structure of network motifs in biology, Meng and Guo [Meng 12] proposed an evolving GRN-based approach for self-organizing robotic swarms to autonomously generate dynamic patterns in unknown environments. Basic idea of the GRN-based model is that firstly several network motifs are predefined as the basic building blocks for GRNs, then covariance matrix adaptation evolution strategy (CMA-ES) is applied to evolve parameters and the structures of the GRNs model. Simulation and experimental results demonstrated that the proposed bio-inspired model is effective for complex shape generation, and the model is robust to environmental changes in complex unknown environments.

2.8.3.3 ES based kinetic rates optimization

ES has been applied successfully to estimate the parameters of constructed models representing biochemical systems.

Spieth et al. [Spie 04] introduced enhancements to evolutionary algorithm optimization process to infer parameters of non-linear system given by observed data more reliably and precisely. A method is proposed to use the advantages of flexible mathematical models to separate the inference problem into two subproblems: to find the topology or structure of the network with genetic algorithm; and to optimize parameters of a mathematical model for the given topology with evolution strategy. Simulation results show that the proposed method is suitable to infer gene regulatory systems in terms of structure and parameters.

Ji and Xu [Ji 06] implemented a C library, named libSRES, to facilitate a fast implementation of computer software for studying non-linear biochemical pathways. The library implements a (μ, λ)-ES evolutionary optimization algorithm that uses stochastic ranking as the constraint handling technique. Regarding the amount of computing time, implementation of the library may face a parameter-estimation problem. An MPI version of libSRES was provided for parallel implementation, as well as a simple user interface. The performance of libSRES has been tested on various pathway parameter-estimation problems, and performance of libSRES has been found to be satisfactory.

Zi and Klipp [Zi 06] presented a SBML based parameter estimation tool (SBML-PET). It is designed to enable parameter estimation for biological models including signalling pathways, gene regulation networks and metabolic pathways. SBML-PET supports import and export of the models in SBML format, and it can estimate the parameters by fitting a variety of experimental data from different experimental conditions. Moreover, SBML-PET has a unique feature of supporting event definition in SMBL models which can also be simulated. Stochastic ranking evolution strategy (SRES) is incorporated in SBML-PET for parameter estimation.

Fomekong-Nanfack et al. [Fome 07] showed that parameter estimation for pattern formation models can be efficiently performed using ES. They use a quantitative spatio-temporal model of a regulatory network for early development in Drosophila melanogaster as a case study. In order to estimate the parameters, simulated results are compared to a time series of gene products involved in the network obtained with immunohistochemistry. Results demonstrated that a (μ,λ)-ES can be used to find good quality solutions in the parameter estimation. Moreover, they also showed that an ES with multiple populations is 5-140 times as fast as parallel SA for the case study, and that combining ES with a local search results in an efficient parameter estimation method.

Sun et al. [Sun 12] presented a comprehensive review of parameters estimation in systems biology by metaheuristics, including implementation of ES.

2.9 Summary

In this chapter, a brief introduction of systematic modelling biochemical systems is given firstly, and then general routes of modelling biochemical systems are presented. We have illustrated what parameter variables in biochemical systems are to be investigated in our research. Biochemical systems can be modelled *in silico* by employing different model standards, for instance graphical and XML formats. We have introduced three computer models standards which are popular and useful in community of computational biology for representing biochemical systems. Moreover, four modelling simulators developed with functionalities are illustrated for tackling problems of models construction, analysis and simulation. Modelers who interest in these simulators can apply them to model different kinds of biochemical processes, for instance continuous/stochastic and quantitative/qualitative biochemical reactions.

Biochemical systems are widely represented and investigated in the community of computational biology. We introduce three well defined and implemented computer based biochemical model formats, Petri Nets, SBML and P Systems, which are in a graphical presentation or a XML based format. After introducing the description of models for representing biochemical systems, we show four popular modelling simulators in systems and synthetic biology for models construction, analysis, optimization and simulations. All these simulators can work with biochemical models constructed in aforementioned biochemical model formats by import and export functionalities. At end of this chapter, background and classification of metaheuristics have been summarized and presented, before illustrating details of two algorithms implemented in our research.

We present details of our proposed hybrid modelling framework with cases study, including basic definitions of biochemical components, genetic mutation operators and composition rules in Chapter 3, a hybrid modelling strategy in Chapter 4, investigations of modelling variants in Chapter 5, and cases study in Chapter 6.

Chapter 3

Representation and Composition of Biochemical Systems

3.1 Introduction

This chapter introduces the background of enzymatic reactions and mass action kinetics, illustrates two patterns as templates for instantiating components, declares atomic components and synthetic models, and describes two libraries to preserve the components and models for modelling biochemical systems. Composition operators and rules are illustrated to compose models of biochemical systems. Generated models of biochemical systems are maintained in terms of synthetic species, composed components and generated structures.

The whole chapter is organized as follows: Section 3.2 illustrates the principle of an enzymatic reaction and how an enzyme catalyzes the biochemical reactions; Mass action kinetics law is shown in Section 3.3 for describing the dynamics of chemical reactions. Three versions of mass action kinetics are presented in Petri net structures, as a fundamental preparation for the definitions of atomic components and synthetic models. Section 3.4

illustrates binding and unbinding patterns as templates for generation of atomic components, declares the atomic components and synthetic models in syntax and semantics. Section 3.5 shows an entity relationship diagram of a MySQL database which maintains the components and models preserved in two libraries.

Related works of applying Petri nets to model biological systems and how Petri net models can be evolved in terms of places and transitions are briefly introduced in Section 3.6. Then three composition operators and a set of composition rules are presented in Section 3.7 for modifying the Petri net models of biochemical systems. Section 3.8 discusses how to maintain constructed Petri net models to ensure the synthetic models comprise of non-conflicting species, unique components and connective topologies for further modelling. Some simple examples of composing biochemical models are demonstrated in Section 3.9, followed a brief summary of this chapter in Section 3.10.

3.2 Enzymatic Reaction

In biochemistry, a chemical reaction is a process of converting molecules of reactants into products within a specific time period. The reactants are usually known as substrates in biochemical reactions. In general, there are spontaneous and enzymatic reactions in a biochemical system.

The spontaneous reaction is a spontaneous decaying reaction, in which a substrate A decays to produce a product B, as shown in Equation 3.2.1:

$$A \rightarrow B \tag{3.2.1}$$

Moreover, due to forward and reverse reaction rates existing in the biochemical reactions, the spontaneous reaction can be reversible between the substrate and product, for instance product B decays back to form substrate A, as described in Equation 3.2.2:

$$A \rightleftarrows B \tag{3.2.2}$$

Most biochemical reactions in cells and organisms are catalyzed by specialized proteins known as enzymes. The enzymes are very important biological catalysts speeding up rates of biochemical reactions in life, by a mechanism of decreasing the amount of energy required in the reactions.

Therefore, an enzymatic reaction is a catalyzed biochemical reaction, facilitating the transformation of a set of substrates into a set of products. The catalysation of the reaction is implemented by enzyme reducing the energy which is required by the reaction to reach a higher energy transitional state [Berg 02, Voet 06].

An enzymatic reaction involves biochemical substrate(s), enzyme(s) and product(s) in a process of molecules conversion. For instance, a simple enzymatic reaction can be illustrated in Equation 3.2.3 to present interactions among one substrate A, one product B and an enzyme E.

$$A \xrightarrow{E} B \tag{3.2.3}$$

The enzymatic reaction can be taken as a basic building block of any biological dynamic system. Therefore, the enzymatic reactions can be used to describe metabolic conversions, the activation of signalling molecules and even transport reactions between various subcellular compartments [Brei 08].

3.3 Mass Action Kinetics

Mass action kinetics are used in chemistry and chemical engineering to describe the dynamics of chemical reactions [Vija 09]. Three types of mass action kinetics were introduced to reveal the catalytic mechanism of an enzyme in enzymatic reactions and metabolism [Brei 08]. Details of enzymatic reactions described by the three types of mass action kinetics are illustrated with corresponding graphic demonstrations in Petri nets as follows. Note that the symbol '|' is used to indicate a complex formed from a substrate and an enzyme.

1. *Mass Action 1 (MA1)*

   A MA1 model takes into account the mechanism by which the enzyme acts, namely by forming a complex with the substrate, modifying the substrate to form a product, and releasing the product in a disassociation. Rate constants are associated with each reaction for a consideration of kinetic properties of many enzymes. The details of MA1 [Brei 08] are shown in Equation 3.3.1.

   $$A + E \underset{\xleftarrow[k_2]{}}{\xrightarrow{k_1}} A|E \xrightarrow{k_3} B + E \tag{3.3.1}$$

   In a MA1 based enzymatic reaction, enzyme E can combine with substrate A to form an intermediate called an enzyme-substrate complex $A|E$ with rate constant k_1; the complex $A|E$ can either dissociate back to E and A with rate constant k_2, or form a product B by transforming A in a dissociation with rate k_3. Graphic presentation of the MA1 based enzymatic reaction in a Petri net is shown in Figure 3.1.

2. *Mass Action 2 (MA2)*

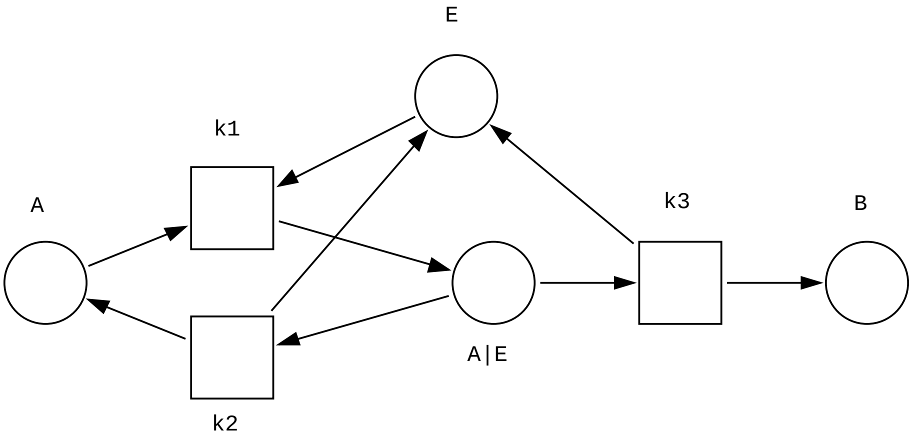


Figure 3.1: An enzymatic reaction based on MA1. Species A is combined with enzyme E to produce a complex $A|E$ by a reaction with a kinetic rate k_1; The complex $A|E$ can be decomposed back to the species A and enzyme E by a reaction with a kinetic rate k_2, or to produce a new species B and E by a reaction with k_3.

An intermediate transition state between substrate and product can exist in an enzymatic reaction. Moreover, the substrate and product bind to the same binding site with highest affinity for the intermediate. In order to approximate the intermediate transition state, an extended MA2 [Brei 08] is formulated for more detailed description of an enzymatic reaction in Equation 3.3.2.

$$A + E \underset{k_2}{\overset{k_1}{\rightleftarrows}} A|E \xrightarrow{k_3{}'} B|E \underset{k_1{}'}{\overset{k_2{}'}{\rightleftarrows}} B + E \qquad (3.3.2)$$

Only one bond is changed between the substrate and product while maintaining complexes $A|E$ and $B|E$ in the enzymatic reaction, thus association and disassociation of a complex $A|E$ and $B|E$ are related. A simple assumption can be given that kinetic rate constants are approximated in the enzymatic reaction based on MA2, for instance $k_1 \simeq k_1{}'$ and $k_2 \simeq k_2{}'$. Graphic presentation of the enzymatic reaction in a

Petri net is shown in Figure 3.2.

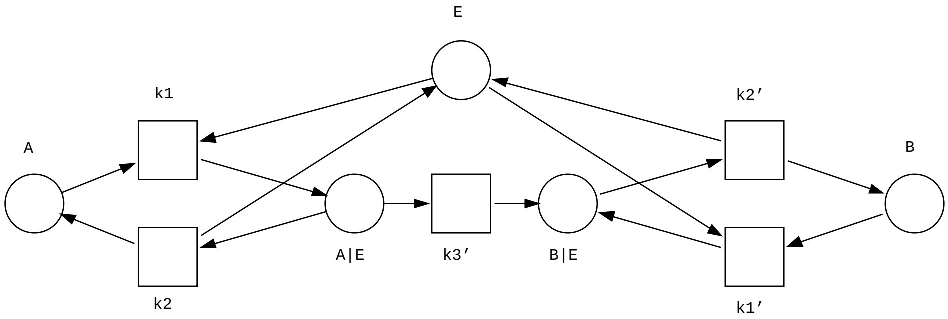


Figure 3.2: An enzymatic reaction based on MA2. Species A is combined with enzyme E to produce a complex $A|E$ by a reaction with a kinetic rate k_1; The complex $A|E$ can be decomposed back to the species A and enzyme E by a reaction with a kinetic rate k_2. There is a intermediate complex $B|E$ transferred from the $A|E$ by a reaction with a kinetic rate k'_3. The complex $B|E$ is decomposed to a new species B and E by a reaction with k'_2. The complex $B|E$ can be produced by combining the B and E.

3. *Mass Action 3 (MA3)*

In a further complete description of an enzymatic reaction, MA3 [Brei 08], a substrate can be associated with an enzyme to form a complex, and then the substrate is modified to form a product which is still associated with the enzyme in the complex. Finally the product and enzyme are released from the complex. The detailed description of above process is shown in Equation 3.3.3.

$$A + E \underset{k_2}{\overset{k_1}{\rightleftarrows}} A|E \underset{{k_4}'}{\overset{{k_3}'}{\rightleftarrows}} B|E \underset{{k_1}'}{\overset{{k_2}'}{\rightleftarrows}} B + E \qquad (3.3.3)$$

In this case the association and disassociation among substrate, enzyme, complex and product are described on different reversible stages, which may offers guidance to biochemists who could carry out further investigation on biological systems of interest. Graphic presentation of the enzymatic reaction in a Petri net is shown in Figure 3.3.

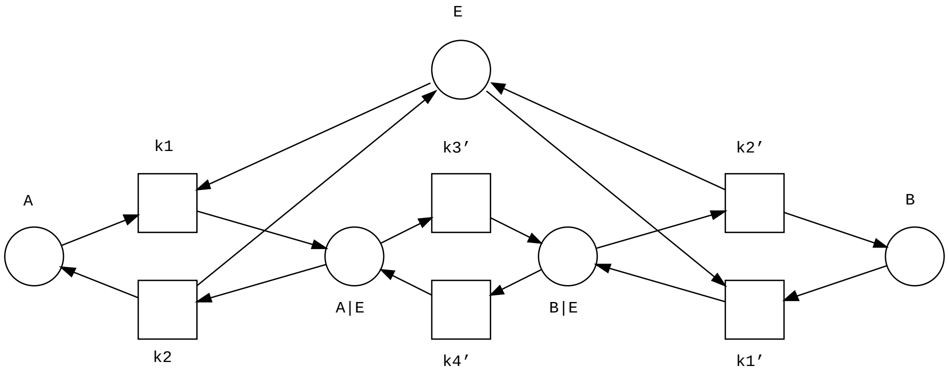


Figure 3.3: An enzymatic reaction based on MA3. As the enzymatic reaction based on MA2, a new species B can be produced, and an intermediate state of complex $B|E$ is formed from the complex $A|E$ by a reaction with a kinetic rate k_3'. Moreover, the complex $B|E$ is able to be transferred back the complex $A|E$ by a reaction with a kinetic rate k_4'.

3.4 Declarations of Component and Model

In this thesis, MA1 is employed to describe an enzymatic reaction which is used as a template to define basic components for building component-based biochemical models. Note that, the components defined by MA1 can be easily extended to the ones defined by other mass action kinetics which are introduced in section 3.3.

3.4.1 Patterns

Atomic components can be instantiated from two general patterns which are templates for components instantiation. The two general patterns describe how two species form a species, or how one species decomposes into two species. Thus, pre-defined patterns in this thesis follow a simple binary format: either two to one standard or one to two standard. Other pattern formats, for instance three (or more) species form one species and one species decomposes into three (or more) species, can be taken as development of our simple pre-defined binary patterns. Any complex biochemical reactions can be described by employing instantiations from the binary patterns, which species interact with each other by composition of instantiations from the binary patterns. Species in our defined binary patterns stands for biochemical reactant, complex or product in an enzymatic reaction. Details of the patterns are illustrated as follows.

- Binding pattern - two reactants are merged into a complex with a specific kinetic rate, as shown in Equation 3.4.1;

  $$P_1 + P_2 \xrightarrow{k1} P_3 \tag{3.4.1}$$

  where the P_1 represents a reactant acting as a substrate, P_2 denotes a reactant acting as an enzyme, and P_3 ($P_3 = P_1|P_2$) is a complex synthesized from P_1 and P_2 by using a '|' symbol to join the labels of two reactants. Graphic presentation of the binding pattern in a Petri net is shown in Figure 3.4.

  In Figure 3.4, there are two non-empty places $P1$ and $P2$, marked with $m1$ and $m2$ as initial concentration values respectively. The two places are associated by a transition $T1$ with a kinetic rate $k1$. Place $P3$ is a product of the transition.

- Unbinding pattern - a complex is disassociated back to reactants, or converted to a product and an enzyme with a specific kinetic rate, as illustrated in Equation 3.4.2.

  $$P_3 \xrightarrow{k2} P_1 + P_2 \tag{3.4.2}$$

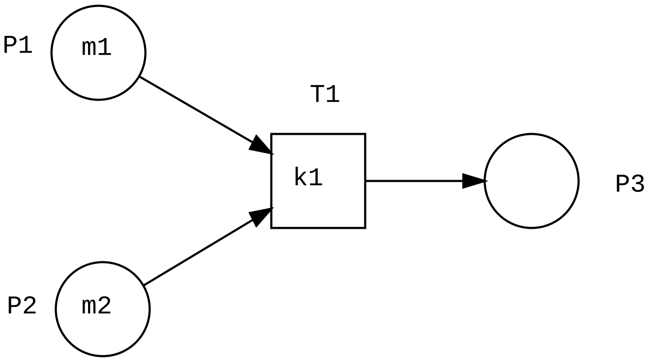
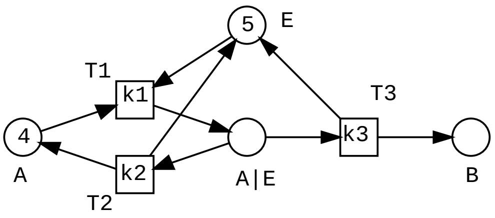


Figure 3.4: Binding pattern. $P1$ and $P2$ are two non-empty places, marked with initial concentrations $m1$ and $m2$. The two places are associated to produce a place $P3$ by a transition $T1$ with a kinetic rate $k1$.

where complex P_3 is either disassociated to two reactants P_1 and P_2 which form the complex itself, or converted into a product and an enzyme. Graphic presentation of the unbinding pattern in a Petri net is shown in Figure 3.5.

Table 3.1: A MA1 based enzymatic reaction and components.

Enzymatic Reaction and Components	Petri net
$A + E \underset{k2}{\overset{k1}{\rightleftarrows}} A\vert E \xrightarrow{k3} B + E$ $A + E \xrightarrow{k1} A\vert E$ $A\vert E \xrightarrow{k2} A + E$ $A\vert E \xrightarrow{k3} B + E$ $[A] = 4$ $[E] = 5$ $[A\vert E] = [B] = 0$	 5 E T1 k1 T3 4 k3 A k2 A|E B T2

Therefore, the enzymatic reaction described by MA1 in Equation 3.3.1 can be composition of one component instantiated from binding pattern and two components instantiated from unbinding pattern. The details of instantiated components are given as follows.

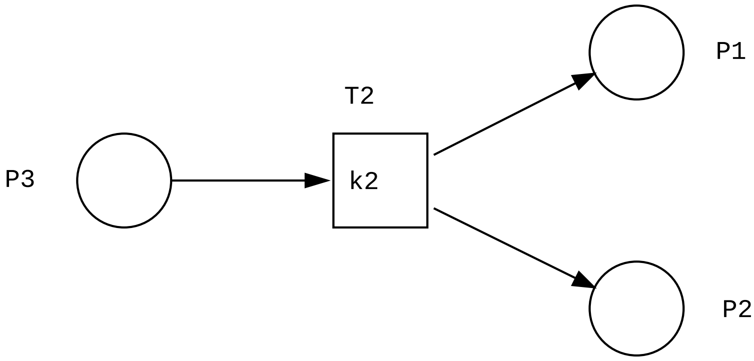


Figure 3.5: Unbinding pattern. A place $P3$ is disassociated to two places $P1$ and $P2$ by a transition $T2$ with a kinetic rate $k2$.

One instantiation of the binding pattern: $A + E \rightarrow A|E$;
First instantiation of the unbinding pattern: $A|E \rightarrow A + E$;
Second instantiation of the unbinding pattern: $A|E \rightarrow B + E$.

The instantiated components and enzymatic reaction in a Petri net are shown in Table 3.1, where concentrations of species are indicated by using labels and square brackets, such as '[A]' and '[$A|E$]'.

These two patterns informally illustrate biochemical process in components which are essential parts of an enzymatic reaction. A formal syntax and semantics of the components are given in following sections for declaration of atomic components for component-based modelling.

3.4.2 Syntax of a component

Definition 3.4.1 (*Component, Syntax*). A component for constructing biochemical models is given by $C = \langle P, T, f, v, m_0 \rangle$, which is based on the structure of Petri nets, where

- P is a disjoint set of three continuous $Places$
- T is a singleton set containing one continuous $Transition$
- $f : ((P \times T) \cup (T \times P)) \rightarrow R_0^+$ defines a set of three directed arcs, weighted by non-negative real numbers, such that there is at least one arc of the form '$p \rightarrow t$' and at least one of the form '$t \rightarrow p$'
- $v : T \rightarrow H$ assigns to the transition a firing rate function, whereby the set of all firing rate functions is $H := \bigcup_{t \in T} \left\{ h_t | h_t : R^{|{}^\bullet t|} \rightarrow R \right\}$, and $v(t) = h_t$ is for the transition $t \in T$
- $m_0 : P \rightarrow R_0^+$ gives the initial marking

Note that place names of a component can be simple (an alphanumeric string) or composite (a series of simple place names each joined by the '|' symbol). Moreover, the number of places of a component is limited to three and the number of transitions is limited to one.

Two components C_1 and C_2, instantiated from binding pattern '$P_1 + P_2 \xrightarrow{k1} P_3$' in Equation 3.4.1 and unbinding pattern '$P_3 \xrightarrow{k2} P_1 + P_2$' in Equation 3.4.2, can be described via the Def. 3.4.1 as follows.

$C_1 = \langle P_1, T_1, f_1, v_1, m_1 \rangle$ where

$P_1 = \{P1, P2, P3\}$

$T_1 = \{T1\}$

$f_1 = \{(P1 \rightarrow T1), (P2 \rightarrow T1), (T1 \rightarrow P3)\}$

$v_1 = \{T1 : k1 \times P1 \times P2\}$

$m_1 = \{P1 : m1, P2 : m2, P3 : 0\}$

$C_2 = \langle P_2, T_2, f_2, v_2, m_2 \rangle$ where

$P_2 = \{P1, P2, P3\}$

$$T_b = \{T2\}$$

$$f_2 = \{(P3 \to T2), (T2 \to P1), (T2 \to P2)\}$$

$$v_2 = \{T2 : k2 \times P3\}$$

$$m_2 = \{P3 : 0, P1 : 0, P2 : 0\}$$

Therefore, according to the Def. 3.4.1, the enzymatic reaction following MA1 kinetic law in Equation 3.3.1, '$A + E \underset{k_2}{\overset{k_1}{\rightleftarrows}} A|E \xrightarrow{k_3} B + E$', can be illustrated by composition of three instantiated components '$A + E \xrightarrow{k1} A|E$; $A|E \xrightarrow{k2} A + E$; and $A|E \xrightarrow{k3} B + E$'. The details of three composed instantiations are shown as following:

$$\begin{aligned} ER_{MA1} &= \{C_1, C_2, C_3\} \\ &= \{\langle P_1, T_1, f_1, v_1, m_1\rangle, \langle P_2, T_2, f_2, v_2, m_2\rangle, \langle P_3, T_3, f_3, v_3, m_3\rangle\} \\ &= \{\langle P', T', f', v', m'\rangle\} \end{aligned}$$

where

$$P' = \{A, E, A|E, B\}$$

$$T' = \{T1, T2, T3\}$$

$$\begin{aligned} f' = \{&(A \to T1), (E \to T1), (T1 \to A|E), \\ &(A|E \to T2), (T2 \to A), (T2 \to E), \\ &(A|E \to T3), (T3 \to E), (T2 \to B)\} \end{aligned}$$

$$v' = \{T1 : k1 \times A \times E, T2 : k2 \times A|E, T3 : k3 \times A|E\}$$

$$m' = \{A : m1, E : m2, A|E : 0, B : 0\}$$

3.4.3 Semantics of a component

Definition 3.4.2 (*Component*, *Semantics*). A component is a system of nonlinear ordinary differential equations (ODEs), illustrating the nonlinear relationship among three involved biochemical elements:

$$\frac{d[P_i]}{dt} = \sum_{t \in {}^{\bullet}P_i} f(t, P_i) \times v(t) - \sum_{t \in P_i{}^{\bullet}} f(P_i, t) \times v(t) \tag{3.4.3}$$

where P_i (i=1,2,3) is one of three continuous $Places$ in a disjoint continuous places set; t is a continuous $Transition$; pre-transitions ${}^{\bullet}P_i$ of the place P_i are all reactions producing the place, thus the continuous transition t is enabled in ${}^{\bullet}P_i$, if the markings of all places in pre-places ${}^{\bullet}t$ are available for firing the transition; the post-transitions $P_i{}^{\bullet}$ of the place P_i are all reactions consuming the place, thus the continuous transition t is enabled in $P_i{}^{\bullet}$, if the markings of all places in post-places $t^{\bullet}$ are available for firing the transition; $f : ((P_i \times t) \cup (t \times P_i)) \to R_0^+$ defines a set of three directed arcs, weighted by non-negative real numbers, such that there are three arcs associated with the continuous transition t by incoming arc $P_i \to t$ or outgoing arc $t \to P_i$; $v : t \to H$ assigns to the transition a firing rate function, whereby the set of all firing rate functions is $H := \bigcup_{t \in T} \left\{ h_t | h_t : R^{|{}^{\bullet}t|} \to R \right\}$, and $v(t) = h_t$ is for the transition t; $[P_i] : P_i \to R_0^+$ gives the concentration of place P_i, which is continuously changed over time.

It should be noted that the translation from Petri nets to the ODEs system is unique but the reverse is not guaranteed [Brei 10].

3.4.4 Syntax of a model

Definition 3.4.3 (*Model, Syntax*). A model of a biochemical system is a generalized form of a component (but with no restrictions on the number of places and transitions) and it is defined by $M = \langle P, T, f, v, m_0 \rangle$, which is based on the structure of Petri nets, where

- P is a disjoint set of at least three continuous $Places$.

- T is a set containing at least one continuous $Transition$.

- $f : ((P \times T) \cup (T \times P)) \rightarrow R_0^+$ defines a set of directed arcs, weighted by non-negative real numbers.

- $v : T \rightarrow H$ assigns to the transitions a firing rate function, whereby the set of all firing rate functions is $H := \bigcup_{t \in T} \left\{ h_t | h_t : R^{|^\bullet t|} \rightarrow R \right\}$, and $v(t) = h_t$ is for the transition $t \in T$.

- $m_0 : P \rightarrow R_0^+$ gives the initial marking.

Places in a Petri net model represent species in the target biochemical system. Markings on the places denote initial concentrations of the species. The transitions are firing rules with assigned kinetic rates. In a sense, a single component can be taken as a model of a specific biochemical system, because the model can only comprise of essential three places and one transition with regard to the syntax definition.

3.4.5 Semantics of a model

Definition 3.4.4 (*Model, Semantics*). A model is a system of ODEs, illustrating the nonlinear relationship among at least three involved biochemical elements:

$$\frac{d[P]}{dt} = \sum_{t \in {}^\bullet P \wedge t \in T} f(t, P) \times v(t) - \sum_{t \in P^\bullet \wedge t \in T} f(P, t) \times v(t) \tag{3.4.4}$$

where P is a disjoint continuous places set ($|P| \geq 3$) for the continuous $Places$ in the model; T is a continuous transitions set ($|T| \geq 1$) for the $Transitions$ in the model; $f : ((P \times T) \cup (T \times P)) \rightarrow R_0^+$ defines a set of at least three directed arcs, weighted by non-negative real numbers; $v : T \rightarrow H$ assigns to the transition a firing rate function, whereby the set of all firing rate functions is $H := \bigcup_{t \in T} \left\{ h_t | h_t : R^{|^{\bullet}t|} \rightarrow R \right\}$, and $v(t) = h_t$ is for the transition t; $[P] : P \rightarrow R_0^+$ gives the concentrations of places which are continuously changed over time.

The ODEs system derived from a model describes the continuous change of concentrations over time for the given species, and it is also a mathematical description of target biochemical system. The same as a place in a component, each place in the model gets an equation which belongs to the ODEs system. Note that the translation from a Petri net of a model to a set of ODEs is unique, but the reverse is not guaranteed [Brei 10].

3.5 Libraries of Components and Models

In order to construct models of biochemical systems by composing components, a storage place should be considered to keep synthetic components and models while modelling. A database was designed by the MySQL database technique and two libraries were developed to preserve the components and models.

Figure 3.6 shows an entity relationship (ER) diagram that describes the aforementioned database. The entity set is represented by a rectangle, and an attribute of the entity is described by an oval. The relationship between the entity sets is denoted by a diamond on the ER diagram.

There are two entity sets in the ER diagram: *Models* and *Components*. There are eight attributes in *Models*: *ID, GenerationID, PopulationID, RatesLabels, RatesConstants, Fitness*, *Structure* and *Simulation*. *Components* has six attributes: *ID, Reaction, Element1, Element2*, *Element3* and *Element4*. The relationship between *Models* and *Components* is

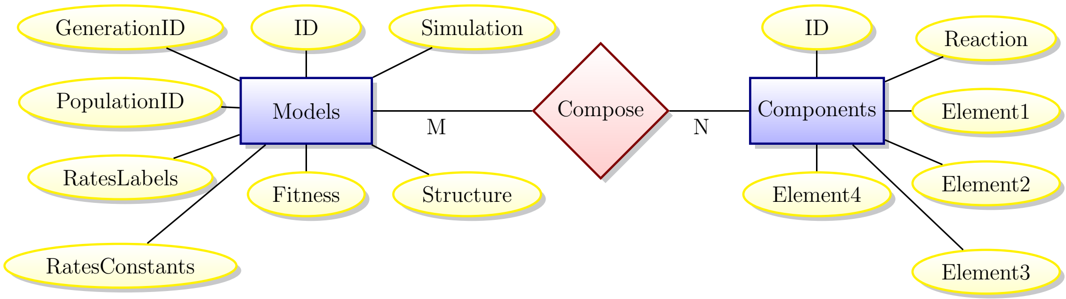


Figure 3.6: Entity relationship of models and components in the database. ‘Models’ is an entity in the database defined with following attributes: ‘ID’ is an unique number for a model under construction; Attributes ‘GenerationID’ and ‘PopulationID’ indicate the model as one of seeds in an evolutionary generation; Attribute ‘Structure’ contains information of the topology of the model; Reactions rates of the model are indicated by ‘RatesLabels’ and ‘RatesConstants’; Attribute ‘Fitness’ is the evaluation result of the model and ‘Simulation’ is for the time series data of model behaviours. Entity ‘Components’ are reactions of a model with following attributes: ‘ID’ stands for an unique reaction; ‘Reaction’ indicate the pattern of the reaction with details of substrates information; ‘Element1’, ‘Element2’, ‘Element3’ and ‘Element4’ are for the labels of substrates and kinetic rate.

‘*Compose*’. The cardinality of relationship ‘*Compose*’ is ‘M:N’, which indicates each entity in the *Models* can be associated with many entities in the *Components*, and each entity in the *Components* is associated with many entities in the *Models*. Note that the ‘many’ could be one or more and sometimes zero.

3.5.1 Components Library

Components are created at initial stage, according to the pre-defined patterns and definition in Section 3.4. A components library L_C was developed as a table in the database, to preserve the generated components as atomic building blocks for modelling biochemical systems. The library L_C maintains detailed information of these atomic components, such as labels of involved species, constants of associated kinetic rates and structures of created components.

Table 3.2: Attributes of an entity component in L_C

Index Type	Attributes	Example
Primary Key	ID	1
Index	Reaction	$P_1 + P_2 \xrightarrow{k_1} P_3$
Index	Element1	P_1
Index	Element2	P_2
Index	Element3	k_1
Index	Element4	P_3

Table 3.2 illustrates details of one component '$P_1 + P_2 \xrightarrow{k_1} P_3$' with its attributes in the library L_C. An entity component is reusable for piecewise modelling biochemical systems, and the attributes are mutable while composing models of the systems. Attribute *ID* indicates the identification of the component; attribute *Reaction* presents the structure of the component; attributes *Element1-4* show the names of the species and the kinetic rate constant of the reaction.

3.5.2 Models Library

Models can be constructed by the composition of reusable components from the library L_C. A models library L_M was developed with the component library L_C in the same database for preservation of synthetic models. The library L_M maintains information of the models, including names of species, structures, kinetic rates constants and simulation results in time series dataset.

Table 3.3 shows details of a model '$P_1 + P_2 \underset{k_2}{\overset{k_1}{\rightleftarrows}} P_3 \xrightarrow{k_3} P_2 + P_4, P_4 + P_5 \xrightarrow{k_4} P_6$' with its attributes in the library L_M. An entity model in the L_M is preserved for representing target biochemical system and supporting a further evolutionary modelling. Attribute *ID* indicates the identification of the model; attributes *GenerationID* and *PopulationID* show the stage of

Table 3.3: Attributes of an entity model in L_M

Index Type	Attributes	Example
Primary Key	ID	1
Index	GenerationID	100
Index	PopulationID	25
Index	RatesLabels	$\{k_1, k_2, k_3, k_4\}$
Index	RatesConstants	$\{0.03, 1.23, 0.6, 0.0072\}$
Index	Fitness	0.68
Index	Structure	$\{P_1 + P_2 \xrightarrow{k_1} P_3,$ $P_3 \xrightarrow{k_2} P_1 + P_2,$ $P_3 \xrightarrow{k_3} P_2 + P_4,$ $P_4 + P_5 \xrightarrow{k_4} P_6\}$
Index	Simulation	*Time Series Dataset*

the model under construction in an evolutionary modelling process; attributes *RatesLabels* and *RatesConstants* are the names and constants of the kinetic rates associated with the biochemical reactions; attributes *Fitness, Structure* and *Simulation* denote the evaluation result, structure and simulation result of the model.

3.6 Modification of Petri Nets

Study of stepwise modification of Petri nets focused on the refinement and abstraction of Petri nets by a bottom-up or top-down approach [Zhou 92], which preserved properties of Petri nets such as liveness and boundedness. The bottom-up approach was employed to merge and/or link subnets to generate a final net, and the top-down approach stepwise refined a first-level Petri net model to increase details of the net until reaching the desired level. In general, modified parts of a Petri net were places, transitions, arcs or subnets of the entire Petri net [Vale 79, Suzu 83, Zube 99, Paul 03, Gome 05].

Since Petri nets theory was utilized firstly to describe biological processes by Reddy et al. [Redd 93] in 1993, Petri net and its extensions were applied to model different types of biological pathways, such as metabolic pathways, signaling pathways, gene regulatory networks and other integrated pathways[Marw 08, Marw 11, Wagl 11]. The models of biological pathways represented in Petri nets were evolved by employing evolutionary algorithms. Mauch [Mauc 03] presented how to employ Petri nets as genomic representations for evolving a population of individuals in genetic programming. An approach was proposed by Moore and Hahn [Moor 03] to use grammatical evolution modelling gene interactions in a Petri net model. Mayo [Mayo 05] applied a method based on random hill climbing to automatically build the Petri net models for the non-linear gene interactions. Nummela and Julstrom [Numm 05] addressed the metabolic pathways prediction problem by employing a genetic algorithm and a stochastic hill-climbing step to search a space of Petri nets representing the pathways. Durzinsky et al [Durz 08] described a method to automatically reconstruct molecular and genetic networks from discrete time series data. More recently, Mayo and Beretta [Mayo 11] proposed a method based on genetic algorithms and data mining to automatically construct Petri net models representing the non-linear gene interactions.

However, the above approaches evolve an existing network model by mutating the connections among existing places and transitions without any creation of new elements during the evolutionary process, whereas our approach in this thesis is to incrementally piecewise construct a network by modifying and composing reusable components. In synthetic biology, modelling of biochemical systems is feasible to achieve desired functionalities by constructing reduced systems. But it is restricted for exploring different model structure, because of modelling process being guided by functionalities. Our piecewise modelling approach is possible to try different composition of components in a heuristic and evolutionary manner. Moreover, it enables the exploration of alternative models space in terms of topologies and kinetic rates for discovering biochemical principles, which is essential for implementation of synthetic biology and other application areas in BioModel Engineering.

Note that a model under construction in this thesis could be a single component, and a simple model can be synthesized with atomic components to form a complicate model by utilizing a set of composition operators and rules. The details of the operators and rules are illustrated in following sections.

3.7 Composition Operators and Rules

Modelling biochemical systems can be achieved by applying composition operators to modify structures of Petri net models representing biochemical systems. A set of composition operators are adapted from the evolutionary optimization [Foge 94, Beye 02] to fine tune the structures of the models. The composition operators and corresponding symbols utilized in this thesis are:

- *Addition*, represented by a symbol $\oplus$
- *Subtraction*, represented a symbol $\ominus$
- *Crossover*, represented by a symbol $\otimes$

Similar to the implementation of genetic operators in evolutionary computation, the proposed composition operators mimic the mutation of natural systems in an evolutionary process to evolve biochemical models. Furthermore, the applications of composition operators are guided by a set of composition rules during the modelling process.

In this thesis, piecewise composition rules are utilized for adding components to a model, removing components from a model and crossing two parental models to reproduce children models. Therefore, three sets of composition rules employed to guide the composition operators can be summarized briefly as follows before illustrating details of the rules:

1. *Addition Rules* are employed to add a component C_a to a model M;

   The component C_a is selected randomly from a library L_C and merged with a component C_m randomly chosen from M. The addition rules allow the component C_a to be merged with C_m into M by replacing parts of labels of the places in C_a with labels of places from C_m.

2. *Subtraction Rules* are implemented to remove components from a model M;

   The subtraction rules permit a component C_m in M to be removed by deleting transition and incident arcs of the C_m, but keeping places of C_m in M for maintenance later.

3. *Crossover Rules* are utilized to cross over two models for generating new models;

   The crossover rules let two models be cut and spliced by swapping parts of the models via an approach of '*Cut and Splice*'.

Models and components involved in the composition process are defined in Petri nets structure, therefore all the composition operators are performed on the places and transitions. Before illustrating the details of composition rules, key points about the composition are given as follows.

- Any one of the three places of a component C_a (C_m) can be randomly chosen as the composition site.

  The composition sites in the components C_a and C_m are places which are used for labels comparison. Parts of the labels of the places are replaced for the integration of the components.

- Labels of places in component C_a can be modified, but labels of places in component C_m are not changed during the composition process.

  We 'borrow' the structure of C_a to develop the topology of the model M by adding arcs, transition and synthetic species. It makes sense that only the labels of places in the added component C_a are modified with the information from the C_m to ensure the 'synthesized' species in a developed model are relevant to a primary 'version' of the model.

In this chapter, L_i ($i = 1, 2, 3$) is used to present labels of places P_i ($i = 1, 2, 3$) in a component C_a from the components library L_C for additions; L_m is used to denote the label of a place P_m in a component C_m randomly selected from a model M. The details of composition rules are illustrated with simple composition examples in the following sections.

3.7.1 Addition operator

Definition 3.7.1 (*Addition Operator,* $\oplus$). An addition operator is a function of merging a component C_a from a component library L_C with a component C_m from an existing model M to generate a new model M':

$$M \xrightarrow{C_m \oplus C_a} M'$$

The added component C_a is selected randomly from the library L_C. Another component C_m is chosen randomly from an existing model M. Since the model and components are presented in Petri net structures, the labels (names) of places (species) of the components C_a and C_m can be compared and merged according to specific rules. A set of addition rules in Section 3.7.2 is applied to guide the addition process.

The topology of model M is monotonically increased to the topology of model M', because of no removal of places and transitions while applying the addition operator. A generated model of a biochemical system should be maintained in a reasonable structure, which requires subtractions utilized on the synthetic structure of the model. A subtraction operator is presented in Section 3.7.3 to achieve the aims of controlling generated models structures.

3.7.2 Addition rules

Since the components are instantiated from the binding and unbinding patterns in Section 3.4.1, addition rules are proposed to deal with composition among components instantiated from different patterns. An overview of our proposed addition rules is given in Table 3.4.

Table 3.4: An overview of addition rules

Rules	Execution
$R^1_\oplus$	Merge S_a and S_m, if $S_m = S_a$
$R^2_\oplus$	Replace S_a with S_m, if $S_m \neq S_a$ and S_m is not a complex
$R^3_\oplus$	Decompose S_m and create a new component by parts of S_m
$R^4_\oplus$	Create a new component by S_m and S_a

Notes:
1. S_m is a species from the model for comparison;
2. S_a is a species from the added component for comparison.

Addition rules summarized in Table 3.4 are general descriptions of how to compare species from an existing model and an added component, and what operations (merging, replacement, decomposition and creation) to be executed regarding different types of species (complex or not). Details of addition rules are illustrated with examples as follows.

A component C_a is added to a model M by merging the places and transition of C_a with a component C_m from a model M. The C_a is in a binding $P_1 + P_2 \xrightarrow{k1} P_3$ or an unbinding pattern $P_3 \xrightarrow{k2} P_1 + P_2$, where L_1, L_2 and L_3 are the labels of places P_1, P_2 and P_3 respectively. The C_m is either in a binding pattern or in an unbinding pattern. When a place P_i (labeled as L_1, L_2 or L_3) is randomly selected from the C_a and compared with a place P_m (labeled as L_m) randomly chosen from the C_m, proposed addition rules are employed for performing the components addition.

- $R^1_\oplus$**: If** $L_m = L_i$ **(**$i = 1,2,3$**), the component** C_a **is added to the model** M **by adding the reaction equations of** C_a **to the set of reactions equations of** M **directly;**

  Example In Figure 3.7, there is a component C_a in the binding pattern. Place P_2 is compared with place P_m and $L_m = L_2$. The $L_m = L_2$ means ‘a species represented by a place P_2 in the component C_a exist in the model M as well’. Then reaction equations of C_a can be added to the set of reaction equations of M directly without any modification on the C_a.

- $R^2_\oplus$**: If** $L_m \neq L_i$ **(**$i = 1,2$**), one of the labels of** L_i **in** C_a **are replaced by** L_m **and reaction equations of modified** C_a **are added to the set of reaction equations of** M**;**

  Example In Figure 3.8, there is a component C_a in the unbinding pattern. The place

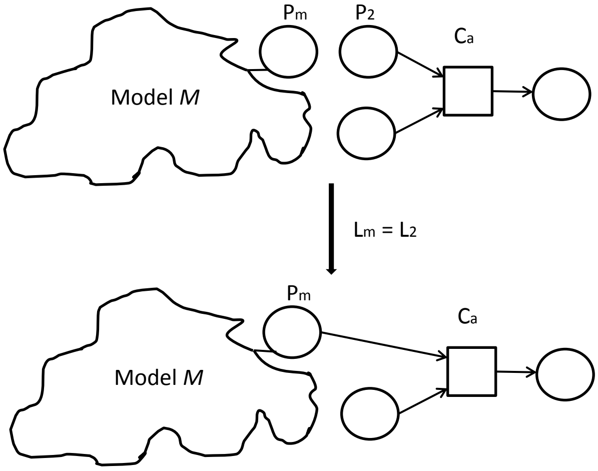


Figure 3.7: Component C_a is added to model M without modification.

P_1 in C_a is compared with place P_m and $L_m \neq L_1$. The $L_m \neq L_1$ means 'two compared species are different'. Therefore, the labels of L_1 existing in the C_a are replaced by the L_m and reaction equations of the modified C_a are added to the set of reaction equations of M.

- $R^3_\oplus$**: If** $L_m \neq L_3$ **and** P_m **is a complex, label** L_3 **in the** C_a **is replaced by** L_m**, label** L_1 **is replaced by** L_{m1} **and** L_2 **is replaced by** L_{m2}**, where** $L_{m1} \cap L_{m2} = 0$ **and** $L_{m1} \cup L_{m2} = L_m$**, and reaction equations of modified** C_a **are added to the set of reaction equations of** M**. Moreover, another component** C'_a **is created by replacing** P_3 **in the** C_a **with** P_m**, but other places in the** C_a **are not modified. Then reaction equations of component** C'_a **are added to the set of reaction equations of** M**;**

  Example In Figure 3.9, there is a component C_a in the binding pattern. The place P_3 is compared with place P_m and $L_m \neq L_3$. The $L_m \neq L_3$ means 'two compared

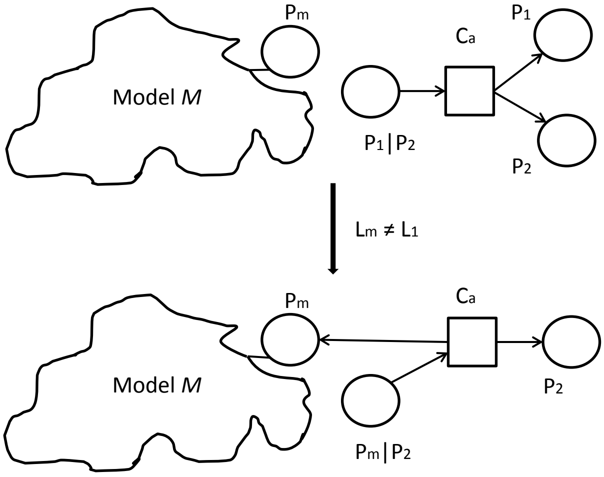


Figure 3.8: Component C_a is modified by replacing labels and added to model M.

species are different'. Because the P_m is a complex, labels L_1 and L_2 existing in the C_a are replaced respectively by two parts of L_m: L_{m1} and L_{m2}. The L_{m1} and L_{m2} are obtained by randomly splitting L_m, where $L_{m1} \cap L_{m2} = 0$ and $L_{m1} \cup L_{m2} = L_m$. The reaction equations of the modified C_a are added to the set of reaction equations of M.

- $R^4_{\oplus}$**: If** $L_m \neq L_3$ **and** P_m **is not a complex, the reaction equations of** C_a **are added to the set of reaction equations of** M **firstly, and a new component** C'_a **is created by binding** P_3 **with** P_m **to produce** $P_m|P_3$ **according to the binding pattern; then the reaction equations of synthetic** C'_a **are added to the set of reaction equations of** M**.**

Example In Figure 3.10, there is a component C_a in the unbinding pattern. The place P_3 is compared with place P_m and $L_m \neq L_3$. The $L_m \neq L_3$ means 'two compared species are different'. Because the P_m is not a complex (without a '$|$' in the label L_m),

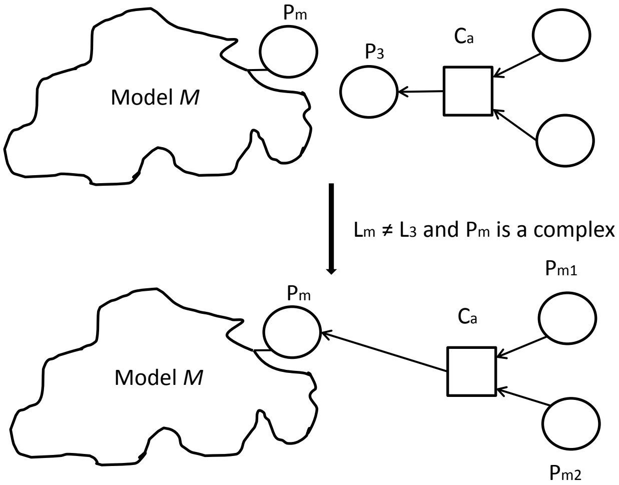


Figure 3.9: Replacement of labels in C_a by the species in M.

a new component C'_a will be created by using the P_m and P_3 in a binding pattern: $P_m + P_3 \xrightarrow{k1} P_m|P_3$. The reaction equations of the component C_a and C'_a are added to the set of reaction equations of M.

3.7.3 Subtraction operator

Definition 3.7.2 (*Subtraction Operator,* $\ominus$). A substraction operator is a function for removing a component C_m from an existing model M to generate a new model M':

$$M \xrightarrow{\ominus C_m} M'$$

In graph theory, removal of nodes (places and transitions in Petri nets) could cause recursive and uncontrolled operations to remove subgraphs. In this scenario, a fixed level(depth) of nodes search in a graph can be introduced for the subtraction. With respect to graph decomposition, we set the subtraction level to one, which means there is only one component

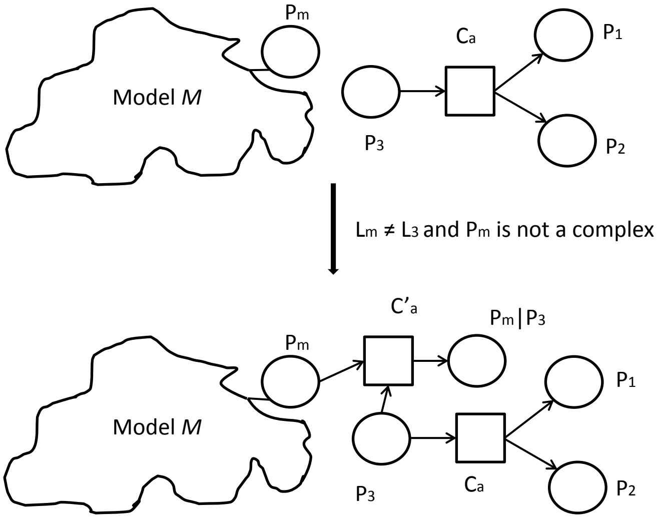


Figure 3.10: Creation of a new component C'_a by P_3 and P_m.

to be removed from the topology of a model at each time.

After performing the subtraction operator during the modelling process, the topology of the model is shrunk. The subtraction operator applied to modify the biochemical models may satisfy the principle of Occam's razor [Thor 15], which is feasible to help bioscientists find a set of simple but interesting structures of biochemical systems for further investigation in wet-lab.

3.7.4 Subtraction rules

A component C_m is selected randomly from an existing model M for the implementation of subtraction by removing transition and incident arcs. Places incident to the removed arcs are not deleted, because any removal of places could affect other transitions which are not involved in current subtraction. Table 3.5 shows an overview of subtraction rules proposed for removing parts of a biochemical model.

Table 3.5: An overview of subtraction rules.

Rule	Execution
$R^1_\ominus$	Do nothing, if there is only one component in a model M
$R^2_\ominus$	Delete transition with its incident arcs in a component C_m from a model M, if there are up to two components in the model M
$R^3_\ominus$	Delete transition with its incident arcs in a component C_m from a model M and maintain modified model M, if there are more than three components in the model M

Notes:
1. M is a model to be modified;
2. C_m is a component to be subtracted from a model M.

Subtraction of components from a model is very simple: to remove linkages among places and transitions. As shown in Table 3.5, transition and incident arcs of a component are the removed parts for performing subtraction operator. Details of subtraction rules are illustrated as follows.

- $R^1_\ominus$**: If a model M comprises only one component, subtraction operator is not implemented;**

  Since atomic component is defined as an instantiation from one of two patterns, a component is a basic and essential complete part within a model. It is obviously that a model must comprise at least one component for exhibiting species behaviour based on fundamental biochemical kinetic law, for instance MA1 in our research.

- $R^2_\ominus$**: If a model M comprises two components, one component C_m is selected randomly from the model M to subtract from the topology of the model. The subtraction is implemented by deleting transition T_m with its incident arcs in the component C_m. Another component and its reaction equations are preserved in the model M.**

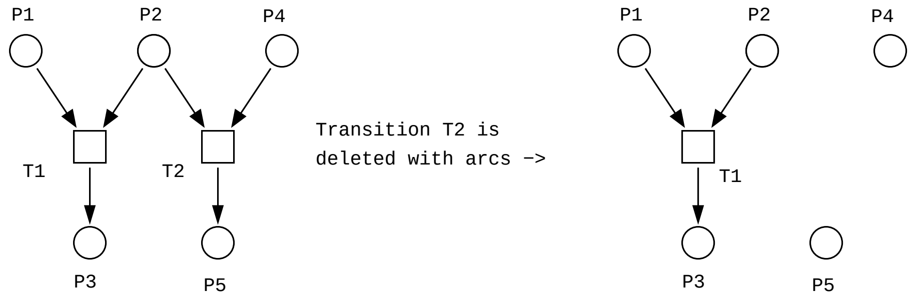


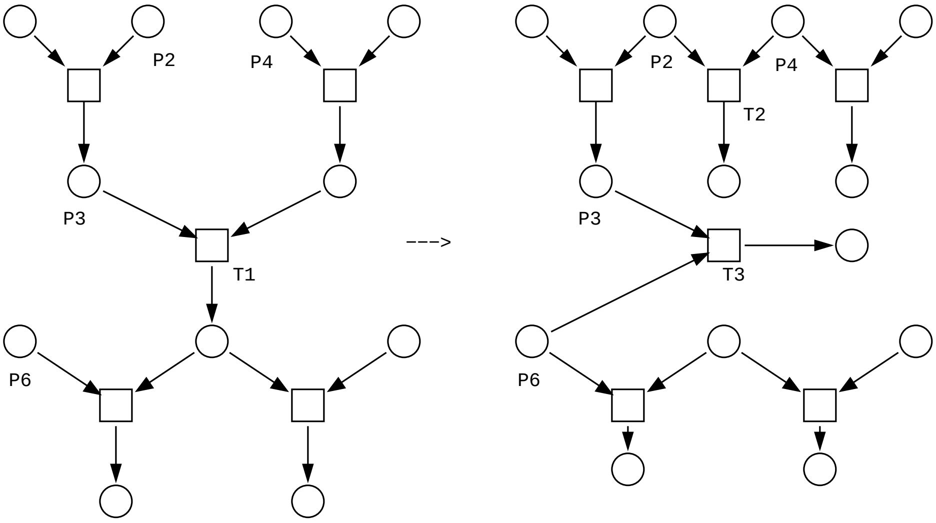
Figure 3.11: Component C_m is removed directly from the model M.

Example In Figure 3.11, transition $T2$ is removed with incident three arcs. There are two isolated places $P4$ and $P5$ which are cleaned up later, and place $P2$ is kept because of its connection to the remain parts of the model M.

- $R^3_\ominus$**: If a model** M **comprises of more than two components, a component** C_m **is selected randomly from the model** M **to subtract from the topology of the model. Then a step of maintenance is applied to check the synthesized places, added components and connectivity of the structure of the model;**

  Example In Figure 3.12, transition $T1$ is removed with incident three arcs. There are three isolated subparts in the model after applying subtraction. In a process of maintenance, places $P2$ and $P4$ are selected randomly from two isolated subparts to make a new component $T2$ by associating a complex between $P2$ and $P4$. Places $P3$ and $P6$ are selected randomly from other two isolated subparts to make a new component $T3$ by associating a complex between $P3$ and $P6$.

After removing the component C_m from the model M, species of C_m can be either incident to M or isolated from M:



Transition T1 is deleted with arcs.
Two components are created.
Transitions T2 and T3 are added.

Figure 3.12: Removal of C_m and linkages of isolated components.

- Incident places - places are incident to the model M, with incoming or outgoing arcs linking to the main parts of the topology of M;
- Isolated places - places are isolated from the model M, without any connections linking to the main parts of the topology of M.

The incident places are still functional parts of other components in the model M, and the isolated places are cleaned up automatically by a process of maintenance of the synthetic model in terms of places, components and structure.

3.7.5 Crossover operator

Definition 3.7.3 (*Crossover Operator,* $\otimes$). A crossover operator is a function of crossing two models M_1 and M_2 to produce two new models, and the two new models compete to be one survival model M' as an individual:

$$M_1 \otimes M_2 \rightarrow M'$$

In this thesis, the crossover operator is adapted from one of crossover variants in genetic algorithms: *Cut and Splice*. The mechanism of cut and splice is illustrated in Figure 3.13 by recombining two parents with different length to produce two children. Two separate crossover points are randomly selected on the parents, before swapping parts of the parents beyond the crossover points. There are two children produced from the swapping, and the characters of the parents are inherited.

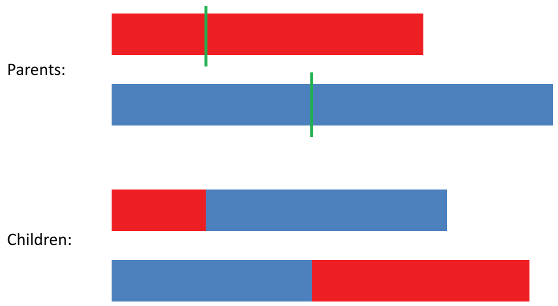


Figure 3.13: Mechanism of *Cut and Splice*.

With respect to the principle of *Cut and Splice*, the crossed models typically inherit many of the characteristics from the parental models. Therefore, it is possible to obtain a set of components with good characteristics in synthetic models. Note that the good characteristics of components in a model can be taken as the functions of producing interesting behaviour of species or composing alternative topologies of target biochemical systems.

3.7.6 Crossover rules

Models are crossed over for generating offspring models which inherit genes (components) of parental models, ensuring evolutionary progress while modelling biochemical systems. An overview of crossover rules are given in Table 3.6, describing how to cut parts of models under construction and to swap components between two models.

Table 3.6: An overview of crossover rules.

Rule	Execution
$R^1_\otimes$	Cut and swap parts of components in two models M_i and M_j ($i \neq j$)

Note: M_i and M_j are two different parental models for performing crossover operator.

Cutting and swapping components from two parental models to produce two offspring models follows the traditional '*Cut and Splice*' mechanism on individuals with binary representation. Components are instances from pre-defined patterns in this thesis and the presentation of components are not in binary format. Basic working mechanism of cut and splice works on many evolutionary modelling issues, thus we employ these evolutionary operations to evolve our models under construction by introducing genetic crossover mutation. Details of crossover rules are described as following:

- $R^1_\otimes$**: Given model** M_i **and model** M_j **(**$i \neq j$**), two cut and splice points** p_i **and** p_j **are chosen randomly in the sets of components of** M_i **and** M_j **respectively. Then components of** M_i **(**M_j**) beyond the** p_i **(**p_j**) are cut away from** M_i **(**M_j**), swapped with components of** M_j **(**M_i**) beyond the** p_j **(**p_i**) and spliced to the rest of components of** M_j **(**M_i**). Finally, two new generated models** M_i' **and** M_j' **are generated, and maintenance is applied to** M_i' **and** M_j' **to reduce duplicate components and link isolated components.**

  Example In Table 3.7, given two models M_1 and M_2 with l_1 and l_2 components

Table 3.7: Crossover between two models

Status	Models	Components (Reactions)
Before $\otimes$	M_1	$\{r^1_{M_1}, ..., r^{p_1}_{M_1}, r^{p_1+1}_{M_1}, ..., r^{l_1}_{M_1}\}$
	M_2	$\{r^1_{M_2}, ..., r^{p_2}_{M_2}, r^{p_2+1}_{M_2}, ..., r^{l_2}_{M_2}\}$
After $\otimes$	M_1'	$\{r^1_{M_1}, ..., r^{p_1}_{M_1}, r^{p_2+1}_{M_2}, ..., r^{l_2}_{M_2}\}$
	M_2'	$\{r^1_{M_2}, ..., r^{p_2}_{M_2}, r^{p_1+1}_{M_1}, ..., r^{l_1}_{M_1}\}$

respectively, two points p_1 and p_2 are selected randomly for crossover, where $1 \leq p_1 \leq l_1$ and $1 \leq p_2 \leq l_2$:

– Model M_1 - a set of components $\{r^1_{M_1}, ..., r^{p_1}_{M_1}, r^{p_1+1}_{M_1} ..., r^{l_1}_{M_1}\}$

– Model M_2 - a set of components $\{r^1_{M_2}, ..., r^{p_2}_{M_2}, r^{p_2+1}_{M_2}, ..., r^{l_2}_{M_2}\}$

After applying the crossover operation, there are two new children models M_1' and M_2' generated with different sets of components l_1' and l_2' respectively, where $l_1' = p_1 + (l_2 - p_2)$ and $l_2' = p_2 + (l_1 - p_1)$:

– Model M_1' - a set of components $\{r^1_{M_1}, ..., r^{p_1}_{M_1}, r^{p_2+1}_{M_2}, ..., r^{l_2}_{M_2}\}$

– Model M_2' - a set of components $\{r^1_{M_2}, ..., r^{p_2}_{M_2}, r^{p_1+1}_{M_1}, ..., r^{l_1}_{M_1}\}$

Since two random cut and spliced points are chosen to be separative sites in two parental models, isolated and duplicated components can exist in children models. The isolated and duplicated components result in a non-connective topology or duplicated arcs among compounds. In order to ensure models under construction are connected and reduced, more operations should be applied to maintain the generated models, Details of maintenance operations are discussed in following section.

3.8 Maintenance of Composed Models

Composition operators and rules are employed to modify the Petri net models and to synthesize new species in the models by renaming labels of places. Since the labels can be simple alphanumeric strings or a series of simple place names joined by the '$|$' symbol, the modified labels of species could create duplicate alphanumeric parts, and repeat components could be generated in the model. It is necessary to maintain the synthetic models after composition. Therefore, three aspects of constructed models should be checked and maintained: names (labels) of species (places), components and topologies.

3.8.1 Maintaining the species

During the composition process, all the components with involved species should be unique. In order to uniquely identify the species and parameters in a model, a naming convention was applied to refer species and parameters with the same names in different models without having to change the names [Rand 08]. In our proposed models composition, partial modification on the labels of places of synthetic compounds can result in duplicate alphanumeric parts joined by a symbol '$|$'. Therefore, the labels of places in a composed model will be sorted in ascending order and clarified by removing duplicate parts between the symbol '$|$'. After manipulating the names of compounds, the species in a model will be unique and clarified for comparisons with other composed places in further composition.

An example of sorting and clarifying the label of a synthetic compound is illustrated as follows. Given two labels of places L_1 and L_2, a label L_3 composed from L_1 and L_2 is synthesized by sorting and reducing the alphanumeric parts of L_1 and L_2:

- L_1: $\{A1|B2|B3|C4\}$ is the label of specie P_1, where '$A1$, $B2$, $B3$ and $C4$' are the names of other species in the model;

- L_2: $\{A1|B3|C2|C3|C4|D1\}$ is the label of specie P_2, where '$A1$, $B3$, $C2$, $C3$, $C4$ and $D1$' are the names of other species in a model;
- L_3: $\{A1|B2|B3|C2|C3|C4|D1\}$ is the label of specie P_3 composed from P_1 and P_2, where duplicate '$A1$, $B3$ and $C4$' are merged to indicate the new synthetic P_3.

Therefore, while modelling biochemical systems by our proposed species maintenance, it is possible to enable modelers identifying the uniqueness of the synthetic components of the composed models.

3.8.2 Maintaining the components

After applying addition and crossover operators to compose models of biochemical systems, the constructed models could comprise of repeat components. The duplicate components are presented in Petri nets with duplicate arcs existing among the transitions and places. The mapped ODEs system of the composed models with these duplicate components contains duplicate mathematical equations, which mathematically illustrates the corresponding models incorrectly. Consequently, the models with duplicate components should be reduced by removing duplicate reactions directly. Table 3.3 shows an example of reducing a composed model with duplicate components.

3.8.3 Maintaining the structures

When an evaluation of a generated model is carried out by simulating a set of ODEs mapped from a corresponding Petri net of the synthetic model, it is necessary to have a set of mapped ODEs consisting with target biochemical system. As introduced in Section 3.7.4

Table 3.8: Maintaining components of a synthetic model

A Synthetic Model	Duplicate Components	The Reduced Model
$M = \{P_1 + P_2 \xrightarrow{k_1} P_3,$		$M = \{P_1 + P_2 \xrightarrow{k_1} P_3,$
$P_3 \xrightarrow{k_2} P_1 + P_2,$		$P_3 \xrightarrow{k_2} P_1 + P_2,$
$P_3 \xrightarrow{k_3} P_2 + P_4,$		$P_3 \xrightarrow{k_3} P_2 + P_4,$
$P_3 \xrightarrow{k_3} P_2 + P_4,$	$P_3 \xrightarrow{k_3} P_2 + P_4$	
$P_4 + P_5 \xrightarrow{k_4} P_6,$		$P_4 + P_5 \xrightarrow{k_4} P_6,$
$P_4 + P_5 \xrightarrow{k_4} P_6,$	$P_4 + P_5 \xrightarrow{k_4} P_6$	
$P_6 \xrightarrow{k_5} P_4 + P_5\}$		$P_6 \xrightarrow{k_5} P_4 + P_5\}$

and Section 3.7.6, isolated components and subparts could exist in a composed model, after modifying the structure of the model. In this scenario, isolation of subparts in a generated model should be reconnected for mapping a set of relevant ODEs. Relevant ODEs enables a synthetic model to be simulated and the behavior of species to be fit correctly during the process of models construction.

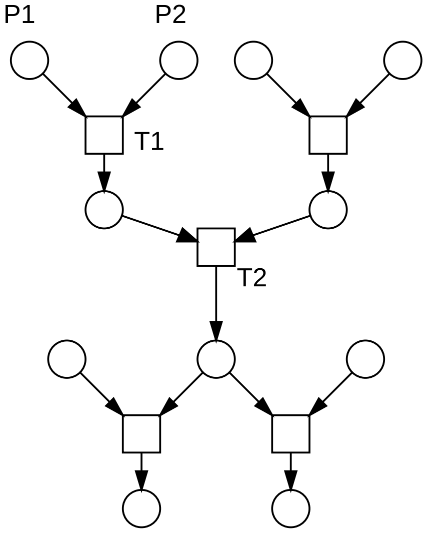


Figure 3.14: An original model for subtraction

In this thesis, we proposed an approach to maintain the connectivity of a Petri net model

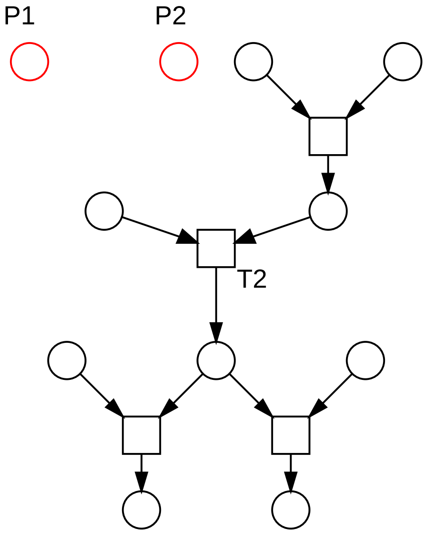


Figure 3.15: One structure of the model after subtraction

by adding a new synthetic component. The added component is created in a binding pattern by using places from the isolated components, and the component is composed to the topology of Petri net model to link isolated parts. For instance in Figure 3.12, two separate parts of the model will be linked by a component which is created via binding places P_3 and P_6 in a transition T_3 to make a new complex $P_3|P_6$. Related works of constructing connective workflow nets can be referred to [Poly 11].

An example of maintaining the structure of a generated model is given as follows. A model *M* is originally represented in Figure 3.14. If the transition T_1 is removed from *M*, two isolated places P_1 and P_2 can exist as shown in Figure 3.15. If the transition T_2 is removed from *M*, two isolated components and one connected subpart of the *M* can exist as shown in Figure 3.16.

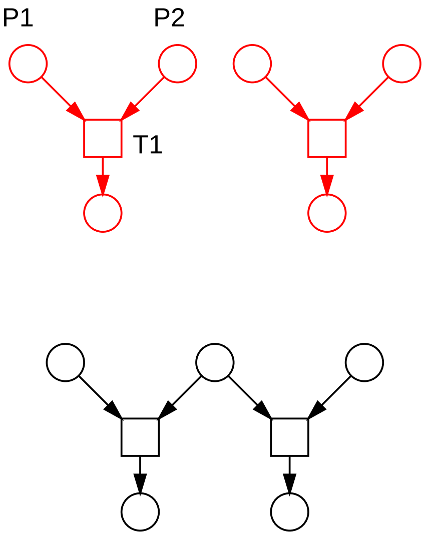


Figure 3.16: Another structure of the model after subtraction

3.9 Examples of Composing Biochemical Systems

There are three demonstration examples of component-model composition: MA1 (See Equation 3.3.1), 3-cascade pathway without feedback, and 3-cascade pathway with feedback. The details of construction and de-construction of these examples are illustrated by composing instantiated components. Note that the composition process is simplified for demonstration, and the composing of biochemical systems is carried out by a hybrid evolutionary modelling approach which is illustrated in Chapter 4.

3.9.1 Composition of an enzymatic reaction based on MA1

Given two elements, three components can be instantiated by a combinatorial method based on the binding and unbinding patterns. An enzymatic reaction based on MA1 can be generated by composing the three instantiated components. The details of patterns, components and composition of the enzymatic reaction are illustrated as following:

- **Templates for components instantiation:**
  - Binding pattern: $P_1 + P_2 \xrightarrow{k1} P_3$
  - Unbinding pattern: $P_3 \xrightarrow{k2} P_1 + P_2$
- **Input elements:** A (acting as a substrate) and E (acting as an enzyme);
- **Instantiated components:**
  - Component C1: $A + E \xrightarrow{k1} A|E$
  - Component C2: $A|E \xrightarrow{k2} A + E$
  - Component C3: $A|E \xrightarrow{k3} AP + E$
- **Composition process:**
  - Step 1: Randomly select component C2 as an initial seed from the library:

$$A|E \xrightarrow{k2} A + E$$

  - Step 2: Add component C3 to C2 by comparing $A|E$ from C2 with $A|E$ from C3. Components C2 and C3 are composed directly because of the same compared places ($A|E$):

$$A + E \xleftarrow{k_2} A|E \xrightarrow{k_3} AP + E$$

  - Step 3: Add component C1 to 'C3 $\oplus$ C2' by comparing A from C1 with $A|E$ from C2. Place $A|E$ is maintained in C2 but place A in C1 is replaced by $A|E$, where C1 is modified as '$A|E + E \xrightarrow{k1} A|E$'. This composition of adding C1 is rejected, because modified component C1 against the rule 'There must be no *Place* to produce *Place* itself, such as $P + Q \xrightarrow{k1} P$'.

- Step 4: Add component C1 to 'C3 $\oplus$ C2' by comparing E from C1 with E from C3. Components C1 and C3 are composed directly because of the same compared places (E):

$$A + E \underset{k_2}{\overset{k_1}{\rightleftarrows}} A|E \xrightarrow{k_3} AP + E$$

Then the enzymatic reaction based on MA1 kinetic law is generated by 'C1 $\oplus$ C3 $\oplus$ C2' after performing aforementioned operators.

3.9.2 Composition of a 3-cascade pathway without feedback

The composition of a 3-cascade pathway without feedback can be obtained by applying addition and subtraction operations to instantiated components. Specially, MA1 is used for the generation of components instantiation in this demonstration.

- **Templates for components instantiation:**
  - Binding pattern: $P_1 + P_2 \xrightarrow{k1} P_3$
  - Unbinding pattern: $P_3 \xrightarrow{k2} P_1 + P_2$
- **Input elements:** R, RR and RRR (which are acting as substrate) and $S1$, $P1$, $P2$ and $P3$ (which acting as an enzyme)
- **Instantiated components:** There are three input elements acting as a substrate and four elements acting as an enzyme. According to a combinatorial principle of choosing input elements for instantiating components by MA1, 36 components are generated and details of these components are shown in Table 3.9.
- **Composition process:**

Table 3.9: Instantiated components in a components library.

NO.	Component Detail	NO.	Component Detail
C1	$R + S1 \xrightarrow{k1} R\|S1$	C19	$R + P2 \xrightarrow{k1} R\|P2$
C2	$R\|S1 \xrightarrow{k2} R + S1$	C20	$R\|P2 \xrightarrow{k2} R + P2$
C3	$R\|S1 \xrightarrow{k3} RP + S1$	C21	$R\|P2 \xrightarrow{k3} RP + P2$
C4	$RR + S1 \xrightarrow{k1} RR\|S1$	C22	$RR + P2 \xrightarrow{k1} RR\|P2$
C5	$RR\|S1 \xrightarrow{k2} RR + S1$	C23	$RR\|P2 \xrightarrow{k2} RR + P2$
C6	$RR\|S1 \xrightarrow{k3} RRP + S1$	C24	$RR\|P2 \xrightarrow{k3} RRP + P2$
C7	$RRR + S1 \xrightarrow{k1} RRR\|S1$	C25	$RRR + P2 \xrightarrow{k1} RRR\|P2$
C8	$RRR\|S1 \xrightarrow{k2} RRR + S1$	C26	$RRR\|P2 \xrightarrow{k2} RRR + P2$
C9	$RRR\|S1 \xrightarrow{k3} RRRP + S1$	C27	$RRR\|P2 \xrightarrow{k3} RRRP + P2$
C10	$R + P1 \xrightarrow{k1} R\|P1$	C28	$R + P3 \xrightarrow{k1} R\|P3$
C11	$R\|P1 \xrightarrow{k2} R + P1$	C29	$R\|P3 \xrightarrow{k2} R + P3$
C12	$R\|P1 \xrightarrow{k3} RP + P1$	C30	$R\|P3 \xrightarrow{k3} RP + P3$
C13	$RR + P1 \xrightarrow{k1} RR\|P1$	C31	$RR + P3 \xrightarrow{k1} RR\|P3$
C14	$RR\|P1 \xrightarrow{k2} RR + P1$	C32	$RR\|P3 \xrightarrow{k2} RR + P3$
C15	$RR\|P1 \xrightarrow{k3} RRP + P1$	C33	$RR\|P3 \xrightarrow{k3} RRP + P3$
C16	$RRR + P1 \xrightarrow{k1} RRR\|P1$	C34	$RRR + P3 \xrightarrow{k1} RRR\|P3$
C17	$RRR\|P1 \xrightarrow{k2} RRR + P1$	C35	$RRR\|P3 \xrightarrow{k2} RRR + P3$
C18	$RRR\|P1 \xrightarrow{k3} RRRP + P1$	C36	$RRR\|P3 \xrightarrow{k3} RRRP + P3$

- Step 1: Randomly select C3 as an initial seed:

$$R|S1 \xrightarrow{k3} RP + S1$$

- Step 2: C3 $\oplus$ C2:

$$R + S1 \xleftarrow{k_2} R|S1 \xrightarrow{k3} RP + S1$$

- Step 3: C3 $\oplus$ C2 $\oplus$ C19:

$$R + S1 \xleftarrow{k_2} R|S1 \xrightarrow{k3} RP + S1$$

$$R + P2 \xrightarrow{k1} R|P2$$

- Step 4: C3 $\oplus$ C2 $\oplus$ C19 $\oplus$ C1:

$$R + S1 \;\overset{k_1}{\underset{k_2}{\rightleftarrows}}\; R|S1 \xrightarrow{k_3} RP + S1$$
$$R + P2 \xrightarrow{k1} R|P2$$

- Step 5: C3 $\oplus$ C2 $\oplus$ C19 $\oplus$ C1 $\ominus$ C19. Transition of $\xrightarrow{k1}$ in C19 component '$R + P2 \xrightarrow{k1} R|P2$' is removed with incident arcs directly. Places $P2$ and $R|P2$ are cleaned up after checking the topology connectivity of remain parts:

$$R + S1 \;\overset{k_1}{\underset{k_2}{\rightleftarrows}}\; R|S1 \xrightarrow{k_3} RP + S1$$

- Step 6: C3 $\oplus$ C2 $\oplus$ C19 $\oplus$ C1 $\ominus$ C19 $\oplus$ C10. Places R in component C10 is replaced by RP:

$$R + S1 \;\overset{k_1}{\underset{k_2}{\rightleftarrows}}\; R|S1 \xrightarrow{k_3} RP + S1$$
$$RP|P1 \xleftarrow{k1} RP + P1$$

- Step 7: C3 $\oplus$ C2 $\oplus$ C19 $\oplus$ C1 $\ominus$ C19 $\oplus$ C10 $\oplus$ C11. Places R in component C11 is replaced by RP:

$$R + S1 \;\overset{k_1}{\underset{k_2}{\rightleftarrows}}\; R|S1 \xrightarrow{k_3} RP + S1$$
$$RP|P1 \;\overset{k_1}{\underset{k_2}{\leftrightarrows}}\; RP + P1$$

- Step 8: C3 $\oplus$ C2 $\oplus$ C19 $\oplus$ C1 $\ominus$ C19 $\oplus$ C10 $\oplus$ C11 $\oplus$ C11. Places R in component C11 is compared with $RP|P1$ in M, then '$RP|P1 \xrightarrow{k2} R + P1$' is created and added. Component C11 '$R|P1 \xrightarrow{k2} R + P1$' is added:

$$R + S1 \;\underset{k_2}{\overset{k_1}{\rightleftarrows}}\; R|S1 \xrightarrow{k_3} RP + S1$$

$$R + P1 \xleftarrow[k_2]{} RP|P1 \;\underset{k_2}{\overset{k_1}{\leftrightarrows}}\; RP + P1$$

$$R + P1 \xleftarrow{k2} R|P1$$

- Step 9: C3 ⊕ C2 ⊕ C19 ⊕ C1 ⊖ C19 ⊕ C10 ⊕ C11 ⊕ C11 ⊖ C11. Transition of $\xleftarrow{k2}$ in component C11 '$R + P1 \xleftarrow{k2} R|P1$' is removed with incident arcs directly. Place $R|P1$ is cleaned up after checking the topology connectivity of remain parts:

$$R + S1 \;\underset{k_2}{\overset{k_1}{\rightleftarrows}}\; R|S1 \xrightarrow{k_3} RP + S1$$

$$R + P1 \xleftarrow[k_2]{} RP|P1 \;\underset{k_2}{\overset{k_1}{\leftrightarrows}}\; RP + P1$$

The 1st cascade layer is generated by Step 1-9, and the 2nd and 3rd cascade layers can be generated in a similar manner, for instance after *N* Steps and *N*+*M* Steps respectively as follows.

- Step *N*: The 2nd cascade layer is generated:

$$RR + S1 \;\underset{k_2}{\overset{k_1}{\rightleftarrows}}\; RR|S1 \xrightarrow{k_3} RRP + S1$$

$$RR + P2 \xleftarrow[k_2]{} RRP|P2 \;\underset{k_2}{\overset{k_1}{\leftrightarrows}}\; RRP + P2$$

- Step *N*+*M*: The 3rd cascade layer is generated:

$$RRR + S1 \;\underset{k_2}{\overset{k_1}{\rightleftarrows}}\; RRR|S1 \xrightarrow{k_3} RRRP + S1$$

$$RRR + P3 \xleftarrow[k_2]{} RRRP|P3 \underset{k_2}{\overset{k_1}{\leftrightarrows}} RRRP + P3$$

The 3-cascade pathway without feedback is generated after two more steps of composition as follows.

- Step *N+M+1*: ‘1st-cascade $\oplus$ 2rd-cascade’ is composed by replacing $S1$ in 2nd-cascade with RP:

$$
\begin{aligned}
R + S1 \underset{k_2}{\overset{k_1}{\rightleftarrows}} R|S1 \xrightarrow{k_3} RP + S1\\
R + P1 \xleftarrow[k_2]{} RP|P1 \underset{k_2}{\overset{k_1}{\leftrightarrows}} RP + P1\\
RR + RP \underset{k_2}{\overset{k_1}{\rightleftarrows}} RR|RP \xrightarrow{k_3} RRP + RP\\
RR + P2 \xleftarrow[k_2]{} RRP|P2 \underset{k_2}{\overset{k_1}{\leftrightarrows}} RRP + P2
\end{aligned}
$$

- Step *N+M+1+1*: ‘1st-cascade $\oplus$ 2rd-cascade $\oplus$ 3rd-cascade’ is composed by replacing $S1$ in 3rd-cascade with RRP:

$$
\begin{aligned}
R + S1 \underset{k_2}{\overset{k_1}{\rightleftarrows}} R|S1 \xrightarrow{k_3} RP + S1\\
R + P1 \xleftarrow[k_2]{} RP|P1 \underset{k_2}{\overset{k_1}{\leftrightarrows}} RP + P1\\
RR + RP \underset{k_2}{\overset{k_1}{\rightleftarrows}} RR|RP \xrightarrow{k_3} RRP + RP\\
RR + P2 \xleftarrow[k_2]{} RRP|P2 \underset{k_2}{\overset{k_1}{\leftrightarrows}} RRP + P2
\end{aligned}
$$

$$RRR + RRP \underset{\underset{k_2}{\longleftarrow}}{\overset{k_1}{\longrightarrow}} RRR|RRP \xrightarrow{k_3} RRRP + RRP$$

$$RRR + P3 \xleftarrow[k_2]{} RRRP|P3 \underset{\underset{k_2}{\longrightarrow}}{\overset{k_1}{\longleftarrow}} RRRP + P3$$

Then the 3-cascade pathway without feedback is generated after application of addition and subtraction operations.

3.9.3 Composition of a 3-cascade pathway with feedback

More composition steps can be applied to a generated 3-cascade pathway without feedback to compose a 3-cascade pathway with feedback, by adding components C1 and C2 from the Table 3.9 in previous section.

- **Composition process:**
  - Step 1: 'a 3-cascade pathway without feedback $\oplus$ C1'. R in C1 is replaced by $RRRP$ which is from the 3-cascade pathway without feedback:

$$R + S1 \underset{\underset{k_2}{\longleftarrow}}{\overset{k_1}{\longrightarrow}} R|S1 \xrightarrow{k_3} RP + S1$$

$$R + P1 \xleftarrow[k_2]{} RP|P1 \underset{\underset{k_2}{\longrightarrow}}{\overset{k_1}{\longleftarrow}} RP + P1$$

$$RR + RP \underset{\underset{k_2}{\longleftarrow}}{\overset{k_1}{\longrightarrow}} RR|RP \xrightarrow{k_3} RRP + RP$$

$$RR + P2 \xleftarrow[k_2]{} RRP|P2 \underset{\underset{k_2}{\longrightarrow}}{\overset{k_1}{\longleftarrow}} RRP + P2$$

$$RRR + RRP \underset{\underset{k_2}{\longleftarrow}}{\overset{k_1}{\longrightarrow}} RRR|RRP \xrightarrow{k_3} RRRP + RRP$$

$$RRR + P3 \underset{k_2}{\longleftarrow} RRRP|P3 \substack{\xleftarrow{k_1} \\ \\ \xrightarrow[k_2]{}} RRRP + P3$$

$$RRRP + S1 \xrightarrow{k1} RRRP|S1$$

- Step 2: ‘a 3-cascade pathway without feedback $\oplus$ C1 $\oplus$ C2’. R in C2 is replaced by $RRRP$ from the ‘3-cascade pathway without feedback $\oplus$ C1’:

$$R + S1 \substack{\xrightarrow{k_1} \\ \\ \xleftarrow[k_2]{}} R|S1 \xrightarrow{k_3} RP + S1$$

$$R + P1 \underset{k_2}{\longleftarrow} RP|P1 \substack{\xleftarrow{k_1} \\ \\ \xrightarrow[k_2]{}} RP + P1$$

$$RR + RP \substack{\xrightarrow{k_1} \\ \\ \xleftarrow[k_2]{}} RR|RP \xrightarrow{k_3} RRP + RP$$

$$RR + P2 \underset{k_2}{\longleftarrow} RRP|P2 \substack{\xleftarrow{k_1} \\ \\ \xrightarrow[k_2]{}} RRP + P2$$

$$RRR + RRP \substack{\xrightarrow{k_1} \\ \\ \xleftarrow[k_2]{}} RRR|RRP \xrightarrow{k_3} RRRP + RRP$$

$$RRR + P3 \underset{k_2}{\longleftarrow} RRRP|P3 \substack{\xleftarrow{k_1} \\ \\ \xrightarrow[k_2]{}} RRRP + P3$$

$$RRRP + S1 \substack{\xrightarrow{k_1} \\ \\ \xleftarrow[k_2]{}} RRRP|S1$$

Then a 3-cascade pathway with feedback is generated after Steps 1 and 2 by composing component C1 and C2 to a 3-cascade pathway without feedback.

3.10 Summary

In this chapter, we have presented binding and unbinding patterns as two templates for instantiating components. An enzymatic reaction can be decomposed into three components instantiated from the two patterns. Moreover, a set of MA1 enzymatic reactions is employed to present biochemical systems in this thesis, but reactions based on other two mass action kinetics MA2 and MA3 can be also utilized to investigate more complex biochemical systems in further research. The atomic components and synthetic models are defined in syntax and semantics for modelling biochemical systems. Two libraries are proposed and implemented in a MySQL database to preserve the components and models respectively.

We have presented how to modify Petri net models of biochemical systems by using a set of composition operators and rules. The composition operators are adapted from evolutionary algorithms in computer science, which allows synthetic models to inherit characteristics of parental models. The composition rules proposed in our research guide the implementation of composition operators to modify the Petri net models, which makes sure composed models are biological relevance and controllable. Moreover, plausible structures of Petri net models can be generated by our proposed composition operators and rules. These alternative models present target biochemical systems in a different view, and biologists in wet-lab would interest in these synthetic alternative models in further experimental investigation.

In order to obtain models with non-conflicting species, unique components and connective topologies after composing models, we have illustrated how to maintain these synthetic Petri net models by manipulating the places, components and structures of models in a maintenance procedure.

Some simple examples of modelling biochemical systems have been demonstrated by applying composition operators and rules to compose atomic components. More details of implementations of the composition operators, composition rules and model maintenance are illustrated in a hybrid piecewise modelling framework in Chapter 4.

Chapter 4

Hybrid Modelling of Biochemical Systems

4.1 Introduction

This chapter firstly introduces related works of modelling biochemical systems via different types of hybrid approaches, points out the importance of hybrid modelling biochemical models in terms of structure and kinetic rates, and presents approaches of piecewise hybrid composing biochemical systems. The hybrid modelling approaches focus on different aspects of biochemical systems: one approach is a one dimension hybrid model generator based on SA algorithm for manipulating model topology and kinetic rates separatively; another approach is a two dimensions hybrid piecewise modelling framework, which shows an integration of ES and SA on a two-layer modelling environment for composing biochemical models in terms of both topology and kinetic rates.

Section 4.2 introduces related works of modelling biochemical systems which has focused on utilizing different metaheuristics and computational models to construct biochemical systems in computational biology. The employed metaheuristics include memetic algorithms, simulated annealing and genetic algorithm. S-systems and P systems are used to describe the computational models. Gene regulatory networks and other transcriptional networks have been investigated as test cases for topology construction and kinetic rates optimization.

Section 4.3 introduces a general framework of modelling biochemical systems in computational biology and illustrates our modelling strategy in this thesis. The basic conceptions of our hybrid modelling methodologies are illustrated before the one and two dimensions modelling approaches are illustrated.

Section 4.4 presents our one dimension hybrid models generator developed by employing SA to construct structures and optimize kinetic rates separatively. In the one dimension hybrid approach, topology mutation is performed iteratively by piecewise adding components to a model seed under construction, and kinetic rates associated with reactions of a model are mutated by the Gaussian distribution and globally optimized by SA. Moreover, there are two ways to refine kinetic rates at each iteration while applying the SA to the modelling process: only one kinetic rate associated with one biochemical reaction is mutated; or all kinetic rates associated with all biochemical reactions are mutated. Our proposed one dimension hybrid models generator can approach these two kinetic rates optimization.

Section 4.5 presents a two dimensions hybrid piecewise modelling framework, which is proposed and implemented with the aims of automatically and intelligently modelling biochemical systems from scratch in an integrated two-layer environment in terms of both topology and kinetic rates. Evolution strategy (ES) is employed to compose models by adding components to (or removing components from) the model candidates. SA is utilized to perform optimization of kinetic rates of the reactions in models. Swapping between ES and SA implementation is performed by exchanging models information between a ES based outer layer and a SA based inner layer in a hybrid framework.

A brief summary of this chapter is given in Section 4.6 which summarizes the working mechanisms of the one dimension and two dimensions hybrid piecewise modelling approaches.

4.2 Related Works of Hybrid Modelling

Models can be constructed in systems biology to predict and explain exhibiting behaviour of biochemical systems, or as templates for designing novel biochemical systems in synthetic biology. It is still an open question regarding how to build and verify models of the biochemical systems, involving intelligent methods and tractable computational tools.

Traditionally the structures of models are inferred from various experimental observations, and the kinetic rates are estimated computationally regarding kinetic laws [Brei 08, Gilb 09]. Given static topologies of models representing the biochemical systems, it is feasible to fit kinetic rates of the models to drive behaviour of models coinciding with observations of given physical systems [Feng 04, Mari 04, Manc 11]. It is also feasible to construct biochemical models by identification of alternative topologies of the target biochemical systems, and then to optimize the topologies with which kinetic rates constants are associated, generating models with similar behaviour to target biochemical systems [Fran 04, Vysh 08].

As topologies and kinetic rates associated with biochemical reactions are both very crucial for biochemical systems exhibiting observed behaviour, it is necessary to model the systems in terms of both topology and kinetic rates by a hybrid method. One of the challenging aims of hybrid modelling research is to develop a robust method for automated models construction from descriptions of the observed or desired behaviour of target biochemical systems, by manipulating both topology and kinetic rates in an integrated and iterative manner. Some previous research has been carried out in hybrid modelling of biochemical systems with respect to the topology and kinetic rates issues, for instance a memetic method and S-systems based inference of gene regulatory networks [Spie 04], an ODEs and SA based optimization of small transcriptional networks and kinetic parameters [Rodr 07a, Rodr 07b], a nested GA and P systems based modelling framework [Cao 10, Rome 08].

Previous research of hybrid modelling mainly relies on constructing models topologies, starting from existing biochemical networks. In these related works, initial topologies of models seeds are modified and evaluated by different metaheuristics, with optimizing kinetic rates. Whereas our developed hybrid modelling approach in this thesis is to incrementally piecewise construct a network from a single component, which starts modelling from a simple structure to a complex one. Moreover, kinetic rates associated with the structure under construction can be optimized in different stages of developing topology in an evolutionary and automatic manner.

A brief introduction of general modelling framework is given in Section 4.3, before our one dimension hybrid models generator is presented in Section 4.4 and two dimensions hybrid piecewise modelling approach is presented in Section 4.5. The simple models generator is designed regarding SA mechanism, and it is implemented to generate topology and optimize kinetic rates separatively. The two dimensions hybrid modelling approach is proposed by taking mechanisms of ES and SA into account while piecewise constructing topology and globally optimizing kinetic rates.

4.3 General Framework of Hybrid Modelling

4.3.1 General framework

Modelling of biochemical systems driven by target behaviour can be illustrated in Figure 4.1. Given a biochemical system with information of observed behaviour from experimental examination in wet-lab, a synthetic model can be modelled by manipulating topology and kinetic rates associated with biochemical reactions in the model seed. Behaviour of species in the synthetic model can be described and used for a comparison between the target modelled biochemical system and the synthetic model. Feedback from the comparison results can be provided to biologists in wet-lab for further experiments, and refinement can be suggested and passed to modelers in dry-lab to modify properties of the synthetic model, such as topology and kinetic rates, for improving quality of the synthetic model. Similar work of designing biochemical systems by computer-aid methodologies has been investigated. Cooling et al. focused on how to use standardization of biological parts to develop libraries of standard virtual parts in the form of mathematical models that can be combined to inform system design. An online Repository was presented to use a collection of standardized models that can readily be recombined to model different biological systems using the inherent modularity [Cool 10].

We apply metaheuristics to evolve topology and optimize kinetic rates of models while composing representations of target biochemical systems in PNs format. The behaviour of biochemical systems and synthetic models utilized by our modelling approach are time series data which is the change of species concentration over simulation time. Comparison of behaviour between target and generated model is approached by measuring behaviour difference, which provides positive or negative information about composed models under

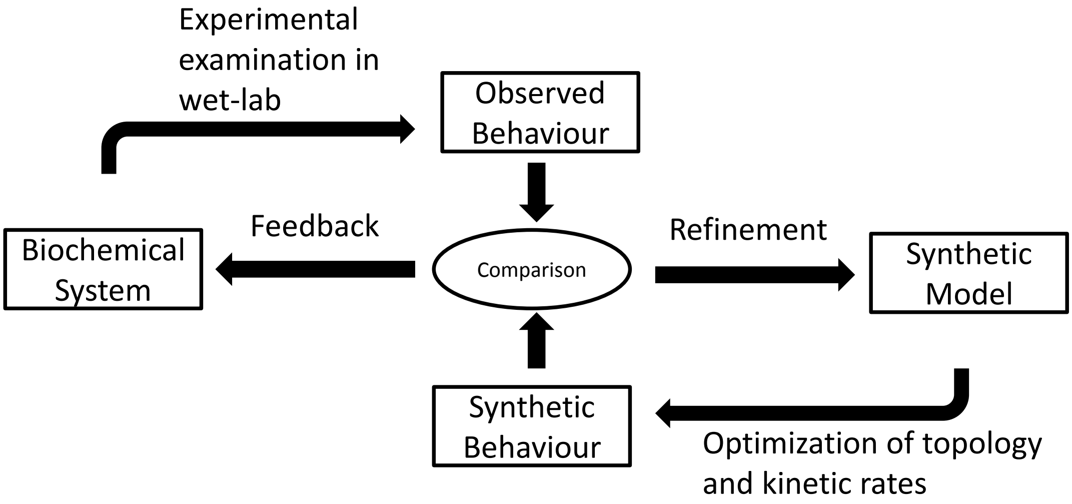


Figure 4.1: A general framework of modelling biochemical systems.

estimation. Then modelling approach can fine tune the models in terms of topology and kinetic rates, for exhibiting improved synthetic behaviour.

4.3.2 Hybrid approach

In this thesis, we apply two metaheuristics, evolution strategy (ES) and simulated annealing (SA), in computer science to tackle issues of modelling biochemical systems in terms of topology and kinetic rates. Since two aspects (topology and reaction rates) of a biochemical system are investigated by employing two different algorithms in a hybrid manner, our hybrid modelling is illustrated in Figure 4.2.

In general, if *ModelConstructionMethod(Topology, Rates)* is applied to *M(T, R)*, indicating that topology and rates of a biochemical model are constructed by different methods. We can have different combinations of hybrid application of methods to the topology and rates: *M*(T_{ES}, R_{SA}) and *M*(T_{SA}, R_{ES}). In this thesis, we investigate *M*(T_{ES}, R_{SA}), which is a hybrid implementation of ES and SA on topology and rates respectively. As shown

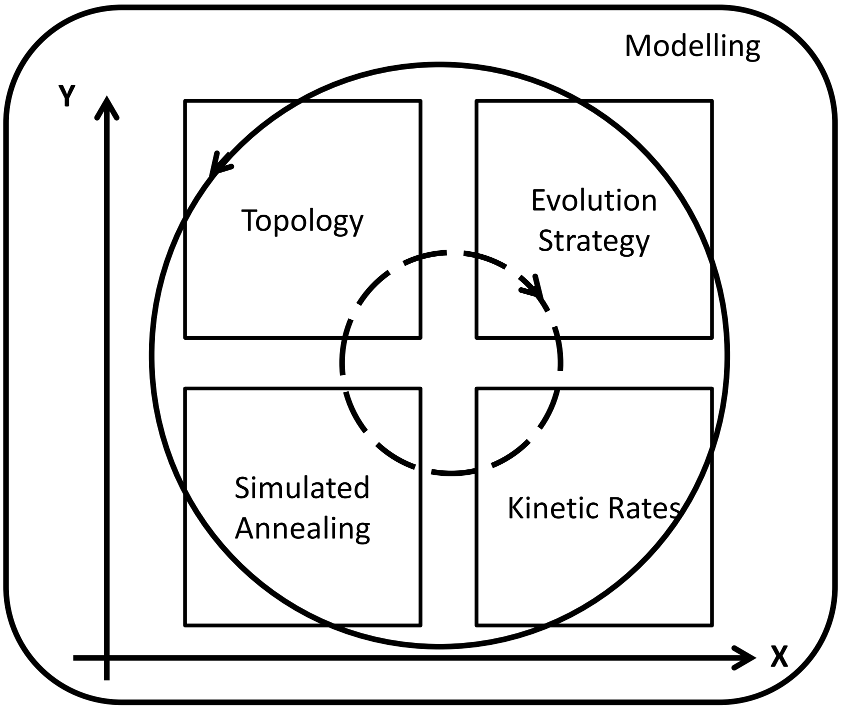


Figure 4.2: Construction of systems operates over two aspects: topology and kinetic rates. Each aspect can be taken as one dimension which needs to be manipulated by the same/different method.

in Figure 4.2, while modelling biochemical systems, the construction operates over two aspects of the systems: topology and kinetic rates. We can take each aspect as one dimension which needs to be manipulated by the same/different method. In this thesis, we have two hybrid approaches which apply different methods to different dimensions while modelling biochemical systems: a one-dimensional (1D) approach and a two-dimensional (2D) approach.

In 1D approach, the algorithm is applied to tackle one problem at each time, for instance SA can be used to fit kinetic rates on x-axis by a combination of ‘SA+Kinetic Rates’, or construct topology on y-axis by a combination of ‘SA+Topology’; and ES can be utilized to develop topology by a combination of ‘ES+Topology’ on x-axis, or optimize kinetic rates by a combination of ‘ES+Kinetic Rates’ on y-axis.

In 2D approach, two algorithms are used to solve modelling issues in a combinatorial manner. For example, in the clockwise direction, ES can be used to fit kinetic rates of a model and the topology of the model is constructed by SA, with these modelling stages being repeated by using ES and SA in turn; or in the anticlockwise direction, ES is employed to develop topology of a model under construction and kinetic rates associated with biochemical reactions are optimized by SA, then modelling operations swap between implementations of ES on topology and SA on kinetic rates until satisfying the termination criteria.

ES is a population-based metaheuristics and it is good at introducing alternative solutions with a probability, we utilize ES to tackle the topology composition in our 2D hybrid modelling approach. Moreover, SA is a single-solution based metaheuristics which obtains optimal solutions by a global search, we employ SA to optimize the kinetic rates associated with reactions in the models under composition. Thus, a combination of ‘ES+Topology’ and ‘SA+Kinetic Rates’ is fundamental hybrid mechanism in the research of modelling biochemical systems by our 2D hybrid approach in this thesis.

4.4 A 1D Hybrid Modelling Approach

SA has been employed to set up a modelling environment in a 1D hybrid models generator for the construction of biochemical systems. The 1D hybrid models generator has two functions: to piecewise build models of biochemical systems, and to iteratively optimize kinetic rates in given biochemical models. The topologies of biochemical systems are constructed in the models generator by manipulating pre-defined components and adding the components to model seeds. The kinetic rates of given biochemical models are optimized in the models generator globally and iteratively.

4.4.1 Topology generation

4.4.1.1 Development of topology

The topology of a model is constructed by assembling pre-defined components together to form a complex structure representing target biochemical system. During the development of topology, components are added to develop the topology incrementally, but kinetic rates associated with reactions in these composed components are not modified. Interactions among species of a model can be represented by arcs in components which are instantiated from the PNs templates defined in Chapter 3. The iterative addition of arcs from added components to a model seed develops the topology of the seed, and the topology space is explored by using the global search mechanism of SA.

An algorithm *BNRSA* (Biochemical Network Reconstruction based on Simulated Annealing) is proposed and implemented in the models generator to illustrate how piecewise developing topologies of models by adding reusable components in a SA based 1D hybrid modelling approach [Wu 10]. The pseudo-code in Algorithm 3 describes the details of the algorithm *BNRSA*.

Given a library preserving reusable components and an initial setting (initial and minimum temperatures, cooling rate, iterations number and initial concentrations of species) for running the 1D hybrid models generator, the piecewise topology construction starts as follows.

A component is selected randomly from the components library as an initial biochemical model seed. Another component is chosen randomly from the library to develop the

```
Require: $D_T$, $S$, $T_0$, $T_{min}$, $CoolingRate$, $N$ and $M_{initial}$
Ensure: $DeltaDistance$, $ModelTopology$ and $SimuResult$
while $T > T_{min}$ do
    while $N \neq 0$ do
        $NewTopology \leftarrow$ Add($Component$, $OldTopology$);
        $\Delta$ C = Cost($NewTopology$)-Cost($OldTopology$);
        if $\Delta C < 0$ then
            $OldTopology \leftarrow NewTopology$;
        else
            if $exp(-(\Delta C/T)) > Random(0, 1)$ then
                $OldTopology \leftarrow NewTopology$;
            end if
        end if
        $N \leftarrow (N - 1)$;
    end while
    Reset $N$
    $T \leftarrow (CoolingRate \times T)$
end while
```

Algorithm 3: Algorithm *BNRSA* (Biochemical network reconstruction based on simulated annealing).

model seed by addition of species and reactions. A new developed model topology is estimated on the cost which is the difference of species behaviour between developed and target model. The calculation of behaviour difference is based on the Euclidean distance equation. The behaviour difference between $NewTopology$ and $OldTopology$ is computed and compared by a 'Cost(Topology)' function.

$$\Delta \text{ C} = \text{Cost}(NewTopology)\text{-Cost}(OldTopology)$$

The 'Cost(Topology)' function is implemented by simulating the given topology with information of species concentrations and kinetic rates, which provides behaviour information of the given topology in time series data format. According to the probabilistic mechanism of SA, there are two methods to accept a new generated topology representing the model under construction:

1. If the cost of $NewTopology$ is less than the $OldTopology$, that is $\Delta C < 0$, then the $OldTopology$ is replaced by the $NewTopology$;
2. If the $NewTopology$ is worse than the $OldTopology$, that is $\Delta C \geq 0$, but there is a probability $exp(-(\Delta C/T))$ satisfying a condition $exp(-(\Delta C/T)) > Random(0, 1)$, where $Random(0, 1)$ is a random double value between zero and one, then the $OldTopology$ is still replaced by the $NewTopology$.

The addition of components and evaluation of developed topologies are repeated N iterations at each system temperature T. The temperature T is lowered by a cooling mechanism '$CoolingRate \times T$' for driving SA system to reach a minimum temperature T_{min}. When a frozen state of SA system is approached, the models generator working on the development of model structure stops to return a final developed topology representing the target biochemical system.

Note that the kinetic rates associated with reactions in the generated topology by the models generator are not modified during the modelling process. Section 4.4.2 illustrates an investigation of employing SA to optimize kinetic rates in given models with fixed topologies, driving species behaviour to approach desired ones in the target biochemical systems.

4.4.1.2 Experimental results

Signalling pathways play a pivotal role in many key cellular processes [Elli 02]. The abnormality of cell signalling can cause the uncontrollable division of cells, which may lead to cancer. For instance, the *Ras/Raf-1/MEK/ERK* signalling pathway (also called the *ERK* pathway) is one of the most important and intensively studied signalling pathways, which transfers the mitogenic signals from the cell membrane to the nucleus [Yeun 00]. In the

ERK pathway, the *Raf-1* kinase inhibitor protein (*RKIP*) inhibits the activation of *Raf-1* by binding to it, disrupting the interaction between *Raf-1* and *MEK*, thus playing a part in regulating the activity of the *ERK* pathway [Yeun 99]. Figure 4.3 shows a graphical representation of the *ERK* signaling pathway regulated by *RKIP*.

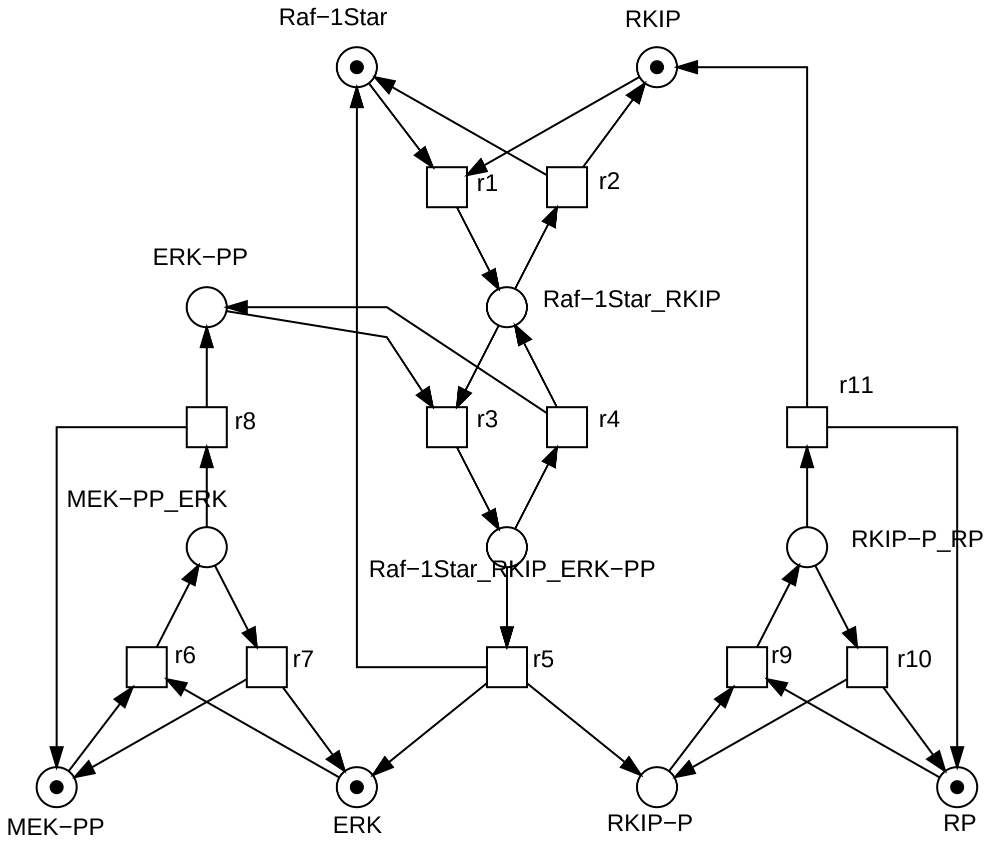


Figure 4.3: A graphical representation of the *ERK* signaling pathway regulated by *RKIP* [Cho 03]

A number of computational models have been developed in order to understand the role of *RKIP* in the pathway and to develop new therapies ultimately [Cho 03, Cald 04]. A well studied model of the *RKIP* inhibited *ERK* pathway described by Cho et al. [Cho 03] is used as an example to test our 1D hybrid simple models generator, with the aims of piecewise constructing and global searching the model topology based on SA algorithm.

Gilbert et al. [Gilb 06] have shown that analysis based on a discrete Petri net model of the *ERK* signaling pathway regulated by *RKIP* can be used to derive the sets of initial concentrations required by the corresponding continuous ODE model, and no other initial concentrations produce meaningful steady states. We used the state 13 derived from the analysis, mapping from the qualitative values of [0,1] to the original quantitative values of [0,2.5] in the model of the *RKIP* inhibited *ERK* pathway given by Cho et al. [Cho 03]. Table 4.1 shows the details of the initial concentrations of species.

Table 4.1: Initial concentrations of species.

Species	μM
$Raf1$	2.5
$RKIP$	0
$Raf1 \mid RKIP$	0
$Raf1 \mid RKIP \mid ERKPP$	0
ERK	0
$RKIPP$	2.5
$MEKPP$	2.5
$MEPP \mid ERK$	0
$ERKPP$	2.5
RP	2.5
$RKIPP \mid RP$	0

For the implementation of SA algorithm, to find the minimum of a given fitness function depends on many parameters. The parameters have a significant impact on the effectiveness of generated solutions for a given optimization problem [Kirk 83]. Because there is not a general way to find the best setting for initial parameters of SA, we apply an empirically derived setting to our test. The setting of parameters we used for SA platform is listed in Table 4.2, which can be investigated and optimized for specific modelling of biochemical pathways in further research.

Table 4.2: Setting of SA parameters for topology generation.

Parameter	Meaning	Value
$T_{Initial}$	Initial temperature	50
T_{Min}	Minimum temperature	0.01
α	Temperature cooling rate	0.95
N	Iterations at each temperature	10

We employed the *BioNessie* [Liu 08] platform to simulate model of the *RKIP* inhibited *ERK* pathway, and generated time course data as a set of target behaviour of species in the model. The information of behaviour in time course data format is used to drive the modelling process by comparing the behaviour distance of species between generated and target model. The measurement of behaviour distance is obtained by employing the Euclidean distance function.

The topology of target *RKIP* inhibited *ERK* signalling pathway is developed from scratch by iteratively adding components to an initial model seed. After iterative additions in the model generator, we can obtain a constructed model which has a similar topology to the target one. A 'similar' topology described in this thesis presents a topology which has major common species and their interactions of the target topology. Some species and interactions may be missed in the similar topology, as well as extra species not in target topology being generated with interactions.

In Table 4.3 we give a comparison between one generated and target model in terms of species. Compared to the original 11 species in the target model of the *RKIP* inhibited *ERK* signalling pathway, there are two species missed from our generated model: '$Raf1 \mid RKIP \mid ERKPP$' and '$RKIPP \mid RP$'. The symbol '|' in the names of species indicates that these species are complex associated from different species. In addition to the nine matched species in both generated and target model, there are also another nine 'new'

Table 4.3: Comparison of generated and target species in topology.

Species	Target Model	Generated Model
$Raf1$	✓	✓
$RKIP$	✓	✓
$Raf1 \mid RKIP$	✓	✓
$Raf1 \mid RKIP \mid ERKPP$	✓	-
ERK	✓	✓
$RKIPP$	✓	✓
$MEKPP$	✓	✓
$MEKPP \mid ERK$	✓	✓
$ERKPP$	✓	✓
RP	✓	✓
$RKIPP \mid RP$	✓	-
$ERKPP \mid Raf1$	-	✓
$RKIPP \mid Raf1$	-	✓
$ERK \mid Raf1$	-	✓
$ERKPP \mid MEKPP$	-	✓
$RKIPP \mid MEKPP$	-	✓
$RKIPP \mid Raf1$	-	✓
$ERK \mid RP$	-	✓
$RKIP \mid RP$	-	✓
$ERKPP \mid RP$	-	✓

species generated in the developed model. But these new synthetic species are not in the target model of the *RKIP* inhibited *ERK* signalling pathway.

Therefore, our model generator can construct target model piece by piece, by adding pre-defined components. The main parts of the topology of target model can be obtained. Extra structure information of the target model is provided with new synthetic species. Biologists may be interested in the new synthetic species, because these new specie could exist in concrete biochemical system but are not being observed or measured in wet-lab experiments.

4.4.2 Kinetic rates optimization

4.4.2.1 Optimization of kinetic rates

Regarding extremely complicate and interconnected relationship among species in biochemical systems, it is very difficult to understand the system behaviour without clearly comprehending the mechanism of enzymatic reactions and associated quantitative kinetic rates, even when there is general knowledge about the topologies of the biochemical systems. Moreover, kinetic rates are not always possible or easy to measure in wet-lab experiments, because of experimental constraints, cost and time. Therefore, it is important to quantitatively study the kinetic rates *in silico* by computational methodologies, especially after obtaining the model topologies for the biochemical systems.

Given a model with fixed topology, kinetic rates associated with reactions can be optimized by employing SA algorithm in the models generator to globally explore the rates space for the model exhibiting desired behaviour. Experimental data of the biochemical systems, for instance behaviour of species, can be used to drive the optimization of kinetic rates by comparing behaviour difference between generated and target model. The difference of behaviour contributes to an objective function for the estimation of optimized kinetic rates.

We proposed Algorithm 4 to describe optimization of kinetic rates in a given model for obtaining desired behaviour of the biochemical system. Given a vector of kinetic rates $K(M)$ for a model with fixed topology, kinetic rates in the $K(M)$ are modified in the models generator by employing SA. After initiating the parameters of SA system on initial and minimum temperatures, cooling rate, and iterations number, Gaussian distribution $N(\mu, \sigma)$ is utilized in a function 'Modify($K^{N}_{t=T_0}(M)$, $N(\mu, \sigma)$)' to manipulate values in $K(M)$ at SA system temperature T_0 and Nth iteration.

```
Require: $T_0$, $T_{min}$, $CoolingRate$, $M_0$, $N$, $M$ and $K^N_{t=T_0}(M)$
Ensure: $K(M)$
while $T_0 > T_{min}$ do
    while $N \neq 0$ do
        $K^N_{t=T_0}(M)' \leftarrow$ Modify($K^N_{t=T_0}(M)$, $N(\mu, \sigma)$);
        $\Delta$ C = Cost($K^N_{t=T_0}(M)'$)-Cost($K^N_{t=T_0}(M)$);
        if $\Delta C < 0$ then
            $K^N_{t=T_0}(M) \leftarrow K^N_{t=T_0}(M)'$;
        else
            if $exp(-(\Delta C/T)) > Random(0, 1)$ then
                $K^N_{t=T_0}(M) \leftarrow K^N_{t=T_0}(M)'$;
            end if
        end if
        $N \leftarrow (N - 1)$;
    end while
    Reset $N$
    $T_0 \leftarrow (CoolingRate \times T_0)$
end while
```

Algorithm 4: Algorithm *KROSA* (Kinetic rates optimization based on simulated annealing).

All modified kinetic rates in the $K(M)$ at each iteration are evaluated by comparing the behaviour distance between the given and target model. The behaviour distance is calculated by using a cost function ‘Cost($K(M)$)’ based on Euclidean distance.

$$\Delta C\text{=Cost(}K^N_{t=T_0}(M)'\text{)-Cost(}K^N_{t=T_0}(M)\text{)}$$

Modification of the vector $K(M)$ is accepted or rejected by following a classical SA probabilistic mechanism of solutions acceptance. The process of optimizing $K(M)$ stops when the system temperature t reaching a minimum temperature T_{min} by cooling rate $CoolingRate$, and returns a final modified vector $K(M)$ with optimized kinetic rates for given model exhibiting desired behaviour.

4.4.2.2 Experimental results

We use the model of *RKIP* inhibited *ERK* signalling pathway introduced in Section 4.4.1.2 as a study case for simulations of kinetic rates optimization. The topology of target model is fixed without modification, while the associated kinetic rates are optimized by employing SA in the models generator.

The values of kinetic rates of the target model for fitting are assigned with the rates constants of state 13 in the model investigated by Gilbert et al. [Gilb 06], as shown in Table 4.4, which are in accordance with the range given in the original paper by Cho et al. [Cho 03].

Table 4.4: Original kinetic rates.

Kinetic Rate	Initial Value	Kinetic Rate	Initial Value
k1	0.53	k7	0.0075
k2	0.0072	k8	0.071
k3	0.625	k9	0.92
k4	0.00245	k10	0.00122
k5	0.0315	k11	0.87
k6	0.6		

A set of ODEs mathematically representing the target model can be used for simulations on optimized kinetic rates. Details of the ODEs are described in Table 4.5 as follows.

Figure 4.4 presents all the behaviour of species in the model of *ERK* signaling pathway regulated by *RKIP*, which is generated by simulation on a set of given ODEs and a group of original kinetic rates.

In the models generator, the values of kinetic rates are fine tuned by Gaussian distribution $N(\mu, \sigma)$ with mean μ and standard deviation σ. Furthermore, there are two ways to optimize the kinetic rates in a given model at each iteration in SA system: to mutate one rate only and to mutate all the rates.

$$
\begin{aligned}
\frac{d[Raf1]}{dt} &= k2 * [RKIP|Raf1] + k5 * [ERKPP|RKIP|Raf1] \\
&\quad -k1 * [Raf1] * [RKIP] \\
\frac{d[RKIP]}{dt} &= k2 * [RKIP|Raf1] + k11 * [RKIPP|RP] \\
&\quad -k1 * [Raf1] * [RKIP] \\
\frac{d[RKIP|Raf1]}{dt} &= k1 * [Raf1] * [RKIP] + k4 * [ERKPP|RKIP|Raf1] \\
&\quad -k2 * [RKIP|Raf1] - k3 * [RKIP|Raf1] * [ERK|RP] \\
\frac{d[ERK|RP]}{dt} &= k4 * [ERKPP|RKIP|Raf1] + k8 * [ERK|MEKPP] \\
&\quad -k3 * [RKIP|Raf1] * [ERK|RP] \\
\frac{d[ERKPP|RKIP|Raf1]}{dt} &= k3 * [RKIP|Raf1] * [ERK|RP] \\
&\quad -k4 * [ERKPP|RKIP|Raf1] - k5 * [ERKPP|RKIP|Raf1] \\
\frac{d[RKIPP]}{dt} &= k5 * [ERKPP|RKIP|Raf1] + k10 * [RKIPP|RP] \\
&\quad -k9 * [RKIPP] * [RP] \\
\frac{d[ERK]}{dt} &= k5 * [ERKPP|RKIP|Raf1] + k7 * [ERK|MEKPP] \\
&\quad -k6 * [ERK] * [MEKPP] \\
\frac{d[RP]}{dt} &= k10 * [RKIPP|RP] + k11] * [RKIPP|RP] \\
&\quad -k9 * [RKIPP] * [RP] \\
\frac{d[RKIPP|RP]}{dt} &= k9 * [RKIPP] * [RP] - k10 * [RKIPP|RP] \\
&\quad -k11 * [RKIPP|RP] \\
\frac{d[ERK|MEKPP]}{dt} &= k6 * [ERK] * [MEKPP] - k7 * [ERK|MEKPP] \\
&\quad -k8 * [ERK|MEKPP] \\
\frac{d[MEKPP]}{dt} &= k7 * [ERK|MEKPP] + k8 * [ERK|MEKPP] \\
&\quad -k6 * [ERK] * [MEKPP]
\end{aligned}
$$

Table 4.5: A set of ODEs for the simulations of optimized kinetic rates.

1. To mutate one kinetic rate associated with one biochemical reaction at each iteration.

   In this scenario, we are interested in fitting one specific biochemical reaction at each iteration, while other kinetic rates associated with rest of reactions are fixed without modifications. The single-reaction based optimization of kinetic rates can offer an opportunity to fit a specific rate in the biochemical system which is difficult to measure or observe in wet-lab experiments.

   Figure 4.5 shows results of fitting one kinetic rate $k1$ from the vector $K(M)$, which simulations are based on the set of given ODEs. The value of initiated $k1$ is firstly assigned with a value from the range of (0, 1] randomly, and then it is modified by

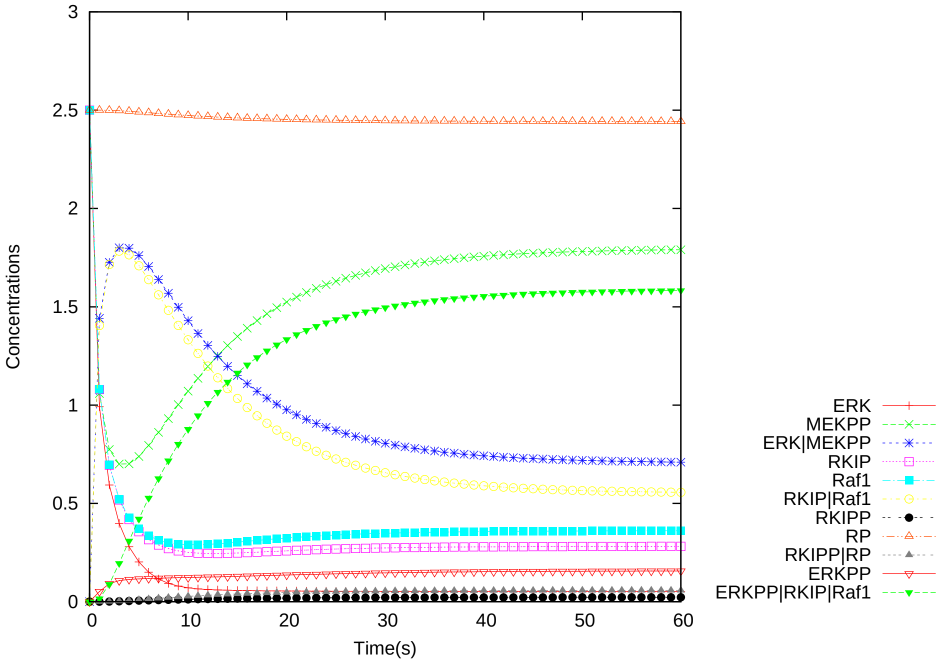


Figure 4.4: Behaviour of all species in *ERK* signaling pathway regulated by *RKIP*.

Gaussian distribution $N(\mu, \sigma)$, where $\mu = k1$ and $\sigma = 0.00001$. The parameters of implementing SA are set as following: initial and minimum temperatures are 100 and 1 respectively, cooling rate is 0.95 and iterations number is 10.

The optimized value of $k1$ in the given model of *ERK* signaling pathway regulated by *RKIP* is 0.64, which is close to the original value 0.53 as shown in Table 4.4.

2. To mutate kinetic rates associated with all reactions at each iteration.

Due to complicated interactions among species usually existing in a given model, all kinetic rates associated with biochemical reactions are important and relevant to exhibiting species behaviour. It is also very difficult to estimate or fit constants of kinetic rates of a given system within uncertain ranges. Thus our approach of optimizing all the rates at the same time enables the comprehensive study of kinetic

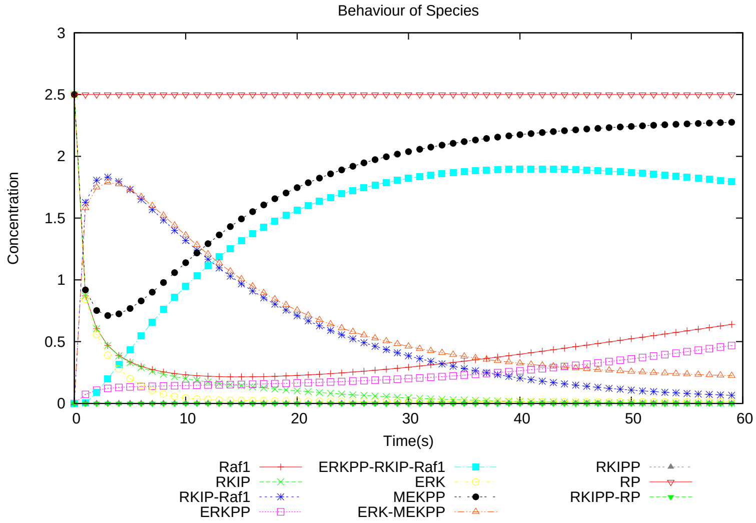


Figure 4.5: Behaviour of a *RKIP* model from optimization of one kinetic rate.

rates.

Table 4.6: Comparison between initial and fitted kinetic rates.

Kinetic Rate	Original Value	Fitted Value
k1	0.53	0.67
k2	0.0072	0.17
k3	0.625	0.22
k4	0.00245	0.85
k5	0.0315	0.77
k6	0.6	0.63
k7	0.0075	0.53
k8	0.071	0.28
k9	0.92	0.29
k10	0.00122	0.20
k11	0.87	0.31

Figure 4.6 shows the results of model behaviour after fitting all the rates in the vector $K(M)$ by a random walk in a range of (0, 1]. The rates values in $K(M)$ are modified by Gaussian distribution $N(\mu, \sigma)$, where $\mu = K_i(M)$, i is the ith kinetic rate and $\sigma = 0.00001$ for fine tuning all the given rates. The parameters of implementing SA

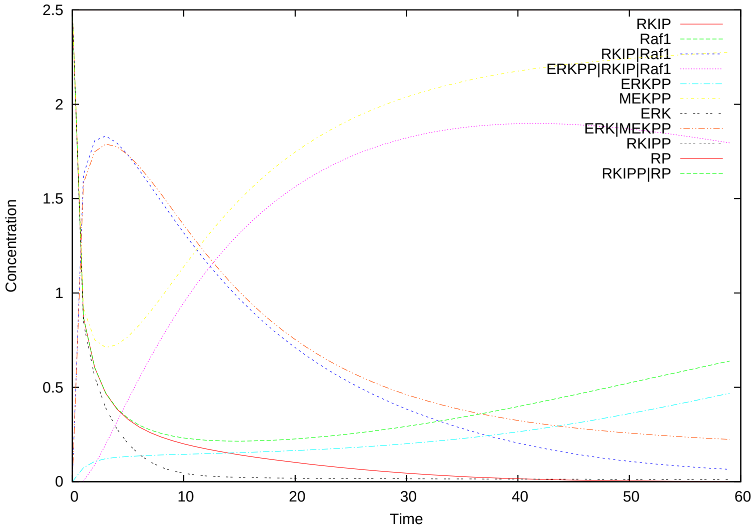


Figure 4.6: Behaviour of a *RKIP* model from optimization of all kinetic rates.

are set as following: initial and minimum temperatures are 100 and 1 respectively, cooling rate is 0.95 and iterations number is 10.

Compared to target species behaviour in *ERK* signaling pathway regulated by *RKIP* in Figure 4.4, it is clear that our models generator can fit the kinetic rates of a biochemical system by employing Gaussian distribution and SA, driving the behaviour of species in the model to exhibit similarly to the target ones. Table 4.6 shows a comparison of fitted kinetic rates obtained from our models generator and original kinetic rates given by Cho et al. [Cho 03].

4.4.3 Discussion

Models of biochemical systems can be obtained by employing SA metaheuristics to add components together and the topology of a model under construction is developed from simple to complex incrementally. Kinetic rates associated with reactions in these synthesized models can be optimized globally by utilizing SA, especially for the estimation of kinetic rates which are difficult to be obtained in wet-lab.

There are two major issues existing in developed models, while applying SA to construct topology and optimize kinetic rates, which details of modelling issues are illustrated as follows.

1. Model topology is developed incrementally without control, due to lack of removal operations on the structure while modelling.

   Without implementation of removing components from a model under construction, the topology of the model is expanded by linking subnetworks represented in PNs incrementally. Regarding the probabilistic acceptance mechanism of SA, a model assembled iteratively by the addition operator could be in a highly interconnected and complicated structure. These models with intricate topologies need to be reduced to simple ones by controlling the number of components of the models. The removal operations can be carried out by removing places (species) and transitions (reactions) of the components represented in PNs, for controlling the topologies in a reasonable size in accordance with the target biochemical systems.

2. Synthetic species are created without supervision, due to the biological meaningless of addition rules applied to manipulate components.

   Regarding the mechanism of addition operator in composition rules applied to models generation, it is easy to increase linkages among species between the added components and the model seed under construction. The linkages are obtained by merging names of species from different components directly. Moreover, added components are instantiated from two pre-defined templates by applying a combinatorial mechanism to a set of input species. That means synthetic species in components before and after components addition are not tested or supervised by biologists or following biochemical knowledge. Thus synthesized species in generated models could not exist in target biochemical systems. Models with unexpected synthesized species are difficult to be checked and validated in wet-lab.

More sophisticate operations should be introduced and investigated to manipulate models in terms of topology, such as components subtraction and models crossover. Instantiations of components and composition rules need to be developed for synthesizing species and merging components in a sophisticated manner to prevent generation of meaningless species. In this thesis, subtraction and crossover operators are proposed and implemented to tackle aforementioned modelling issues while composing biochemical models. Details of these operators are illustrated in Section 4.5.

4.5 A 2D Hybrid Modelling Approach

In this thesis, we aim to solve a topology construction problem by iteratively piecewise assembling components represented by quantitative PNs from a user pre-defined library, combined with optimizing kinetic rates associated with biochemical reactions. We developed a 2D hybrid piecewise modelling approach which integrates ES and SA together, for piecewise composing topologies of models and globally optimizing kinetic rates of the models.

Regarding application of metaheuristics to modelling of biochemical systems, there are some questions which need to be discussed before illustrating the details of our hybrid modelling approach.

1. Why using ES and SA, but not other metaheuristics?

The ‘No Free Lunch (NFL) theorems’ is the first reason we choose ES and SA from a set of metaheuristics for the investigation of evolutionary modelling. The NFL theorems are described as ‘An algorithm performs well on a certain class of problems, then it necessarily pays for that with degraded performance on the set of all remaining problems’ by Wolpert and Macready [Wolp 97]. The NFL theorems show that any pair of algorithms has identical average performance on the static and time dependent optimization problems. In other words, if an algorithm *A* performs better than another algorithm *B* over some class of optimization problems, then the algorithm *B* must perform better than the algorithm *A* over a set of all other optimization problems. Therefore, we can take the point of view that there is not a general and universal optimization scheme suitable for any optimization problems.

In addition, metaheuristics have been employed to study the modelling of biochemical systems in computational biology, for instance GA and GP. It is still necessary to investigate different metaheuristics and their applications to model biochemical systems in terms of topology and kinetic rates, for a complementary and overall research of utilizing metaheuristics in computational biology. That is why we choose ES and SA as our methodologies to set up a hybrid modelling environment and model biochemical systems in a piecewise manner.

2. What is the major difference of applying ES, SA and GA to the modelling process?

   In general, ES and GA are both population-based optimization methodologies. They can start from a set of solution candidates and evolve these candidates to approach optimal solutions for the optimization problems. The major difference between GA and ES is that GA stresses chromosomal operators, whereas ES emphasizes behavioural changes at the level of the individual [Foge 94]. We are interested in the change of behaviour of manipulated individuals by a hybrid piecewise modelling approach in this thesis, therefore it is better to employ ES to address the evolutionary modelling issues.

SA is a single-solution based global optimization metaheuristics. It is easy to shift the search strategy from global optimum to local optimum via a controllable parameter, imitating a temperature cooling scheme in the physical environment. The evolved model candidates in the hybrid modelling framework are usually very complicate, because of uncertain kinetic rates with/without knowledge of models topologies. It is essential to start the fitting of kinetic rates for each model candidate from a global level to a local level, especially while the topologies of these models are being mutated on a population-based modelling platform. That is why we choose SA to examine the kinetic rates of each model under construction in the hybrid piecewise modelling framework.

3. Why hybridizing ES and SA, not applying ES and SA in a serial manner for building and optimizing models in terms of topology and kinetic rates, separatively?

   Metaheuristics are often inspired from natural environment and very powerful in sorting out optimization problems. Dozen of metaheuristics and their variants have been developed and utilized to tackle the optimization problems in the real world. It is definitely useful to apply one metaheuristics to the optimization problems. But another promising way to get much valuable optimization results is to develop hybrid metaheuristics and investigate the implementation of these hybridized metaheuristics, which concerns the combination of several search algorithms with strong specialization in intensification and/or diversification [Loza 10].

> Therefore, we hybridize ES and SA in a two-layer piecewise modelling framework which iteratively composes the structures of models and optimizes the kinetic rates in a combinatorial manner. The aims of hybridizing ES and SA are to tackle the problems of manipulating models in terms of topology and kinetic rates via an intelligent and automatic swapping mechanism, and to find a potential trade-off between composing models structures and optimizing reactions rates heuristically.

Regarding characteristics of ES and SA metaheuristics, two different but switchable layers are designed and developed for applying ES to mutate model topology and SA to optimize kinetic rates in a hybrid manner. Details of the proposed 2D hybrid piecewise modelling approach are presented in following sections.

4.5.1 A general flowchart of the hybrid modelling

A hybrid evolutionary and heuristic piecewise modelling approach has been developed by hybridizing two metaheuristics algorithms on two layers: topologies of the models representing a target biochemical system are evolved by employing ES at outer layer, and SA is applied to optimize kinetic rates associated with the reactions in these evolved models at inner layer. The operations of evolving topologies and optimizing rates are switchable on the two layers, and information of models under construction is exchanged for simulation of the models and evaluation of the modifications on topology and kinetic rates.

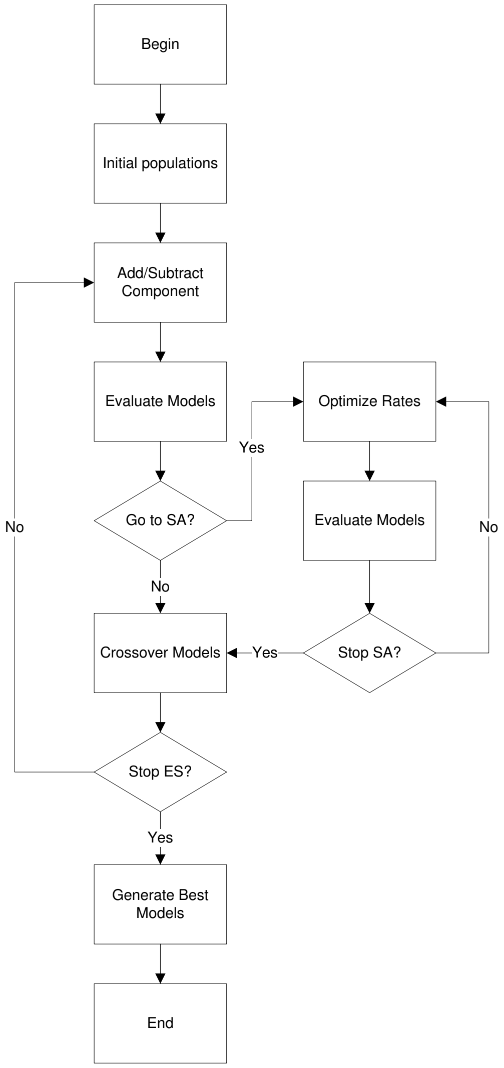


Figure 4.7: A flowchart of hybridizing ES and SA to model biochemical systems.

A general flowchart is shown in Figure 4.7 to illustrate the hybridizing between ES and SA for models construction. As shown in the modelling flowchart, the modelling process is based on a scheme of piecewise composing components iteratively. A set of initial model seeds is given to compose components, and the composed models are mutated and evaluated at ES outer layer. Before going to crossover these composed models at the end of the modelling process at ES outer layer, kinetic rates of these composed models are optimized globally at SA inner layer. The optimization of topologies stops after a pre-defined number of generations at ES outer layer, and the optimization of kinetic rates stops after the system reaches a minimum temperature at SA inner layer. A set of best synthetic models is returned at the end of the hybrid piecewise modelling process, providing information of alternative models with similar behaviour to the target system.

With respect to related works of hybrid modelling biochemical systems [Cao 10], our work differs from them in terms of underlying representation of biochemical models: we use Petri nets and they use P-systems. Moreover, we can perform incremental piecewise addition of basic components resulting in new compounds during the modelling process, as well as genetic operations due to our use of ES with mutation operators, while their approach is confined to genetic operations.

4.5.2 Topology construction based on ES outer layer

Outer layer of the hybrid modelling approach is designed for implementation of ES to compose topologies of models under construction. A classical (μ+λ)-ES [Beye 02] is utilized to piecewise assemble components from components library to the models iteratively, where μ and λ are the numbers of parental and children individuals, respectively. The (μ+λ)-ES starts from an initial population of model individuals which are single components selected randomly from the components library.

Three composition operators (*Addition, Subtraction* and *Crossover*) are applied to modify topologies of the individuals. Because the composition operators are adapted from evolutionary algorithms in computer science which are well studied for mimicking natural selections, it is feasible to employ these operators to evolve biochemical models in terms of topology. In general, the *Addition* operator is used to integrate components to an existing model. The *Subtraction* operator is utilized to subtract components of a model by removing transitions with incident arcs of these components. The *Crossover* operator is employed to apply a ‘cut and splice’ method to swap parts of two models under construction to generate new models. In Section 3, the composition operators and corresponding composition rules are illustrated in detail.

```
Require: $CompLib$, $ModLib$ and $ComposRules$
Ensure: $BioN_{Best}$
Initiate the population;
while Not reach maximum generation (ES layer) do
    for Each individual in the population do
        Modify the topology of individual by Addition $\oplus$ or Subtraction $\ominus$;
        Check the topology of modified individual;
        Evaluate the modified individual;
        Optimize kinetic rates of modified model (SA layer);
    end for
    Cross over the individuals by Crossover $\otimes$;
    Select offsprings for next generation;
end while
Return $BioN_{Best}$.
```

Algorithm 5: A ES based outer layer for model topology composition.

Algorithm 5 shows the pseudo-code for model topology composition at ES outer layer. Before constructing the models of biochemical systems, two libraries *CompLib* and *ModLib* are set up for preserving instantiated components and composed models, respectively. The atomic components in the library *CompLib* are instantiated from binding and unbinding patterns as defined in Section 3.4. Preserved components are based on information of input substrates, and a combinatorial mechanism is applied to generate components among these substrates. Moreover, the components are reusable in *CompLib*, and the library *CompLib* is accessible during the modelling process for components selection and composition with model individuals. The library *ModLib* preserves synthetic models which are alternative models for illustrating target biochemical systems in terms of topologies and behaviours. Composition rules *ComposRules* are applied to compose components to the model seeds in an initial population.

The ES outer layer in the hybrid modelling approach is in charge of modifying the structures of models by composition operators and rules. After being modified on the topologies, models are checked for connective and redundant components. Then the models are evaluated by using Euclidean distance function in an objective function to measure the behaviour distance of species between generated and target model.

Kinetic rates of these composed models are optimized at SA inner layer, whereas topologies of these composed models at this layer are fixed without modification. The details of implementation of SA at inner layer are described in following Section 4.5.3.

Before stopping topologies construction at ES outer layer, there is a crossover operation applied to synthetic models. The aims of applying crossover operation are to mate model individuals in the same population and to allow model offsprings inheriting genetic chromosomes (good biochemical reactions and species) for the next generation. At the end of the piecewise hybrid modelling approach, a group of best models in terms of similar behaviour to the target biochemical system is returned and preserved in models library *ModLib* for further investigation.

4.5.3 Kinetic rates optimization based on SA inner layer

SA is a heuristic optimization algorithm for searching for a global optimal solution in a very large solutions space, avoiding local optimum solutions. In our previous work [Wu 10], we have applied SA to piecewise construct and explore the topologies of models representing biochemical systems. In this thesis, SA is integrated within an ES based outer layer as an inner layer to optimize the kinetic rates of composed models obtained from ES outer layer. The topologies of these synthetic models are fixed at SA inner layer, while corresponding kinetic rates are optimized.

```
Require: $M$, $K(M)$, $IterNum$, $\alpha$, $T$ and $T_{Min}$
Ensure: $M$ and $K'(M)$
  while $T > T_{Min}$ do
    while $IterNum! = 0$ do
      Mutate $K(M)$ by Gaussian distribution $N(\mu, \sigma)$;
      Evaluate the model $M$;
      Accept $M$ based on the Metropolis algorithm;
    end while
    Reset $IterNum$;
    Lower $T$ by $\alpha$;
  end while
  Return $M$ with optimized kinetic rates in $K'(M)$.
```

Algorithm 6: SA based inner layer for model kinetic rates optimization.

Algorithm 6 shows the pseudo-code for optimizing kinetic rates at SA inner layer. The kinetic rates associated with biochemical reactions in a given model M are coded in a vector $K(M) = (k_1^t, k_2^t, ..., k_l^t)$, where l is the number of reactions, t is the current SA system temperature $t = T$, and k_i^t is a constant rate of the ith biochemical reaction r_i ($i = 1, 2, ..., l$). The vector $K(M)$ is mutated by the Gaussian distribution $N(\mu, \sigma)$ with $IterNum$ iterations at each system temperature. The mutated $K(M)$ of the model is evaluated at each iteration, by comparing the Euclidean distance of species behaviour between the model M and the target pathway.

The evaluated model M with optimized $K(M)$ is accepted or rejected, according to a classical Metropolis mechanism. Accepted M is preserved as a new start seed for the next run of $K(M)$ optimization. The same model M with different rates values in $K(M)$ is optimized at different SA system temperatures by a cooling rate α. The whole optimization process stops when system temperature reaches a minimum temperature T_{Min}.

Due to probabilistic behaviour of random procedure of SA [Anil 87], a mutated vector $K(M)$ which causes a bad estimated fitness of the model M could be generated. Therefore, it is possible to have a model returned from SA inner layer after optimizing associated kinetic rates in a fixed topology that is worse than the one passed from ES outer layer to SA inner layer in the hybrid modelling approach.

4.6 Summary

In this chapter, one dimension and two dimensions hybrid modelling approaches are developed and illustrated for piecewise modelling biochemical systems in terms of topology and kinetic rates.

The one dimension hybrid modelling approach is implemented in a simple models generator, which is developed by using SA to iteratively expand the model structure, and to globally explore the kinetic rates values of biochemical reactions. The main advantages of the models generator are to build models structures from scratch for describing target biochemical systems and optimizing kinetic rates iteratively by single-reaction and all-reactions based methods. Previous research of employing one metaheuristics to model biochemical systems has focused on mutating structures to obtain models exhibiting desired systems behaviour, and research of optimizing kinetic rates has been carried out by fitting rates associated with a small group of biochemical reactions. The simple models generator developed in this thesis improve the topologies construction by a piecewise modelling methodology and the kinetic rates optimization by an overall rates exploration.

The two dimensions hybrid modelling approach is performed in a two-layer piecewise modelling framework, which integrates ES and SA together for evolutionary composing topologies and globally optimizing kinetic rates in a hybrid manner. Because ES is a population-based heuristical evolutionary algorithm, it is feasible to evolve a set of model candidates by using mutation operators on the topologies. While evolving the topologies of models, the kinetic rates of each model can be optimized iteratively by SA which is a single-solution based heuristic algorithm. The two dimensions hybrid piecewise modelling approach benefits the process of modelling biochemical systems, regarding structures and rates at the same time, which is very difficult to be tackled in wet-lab experiments.

The two dimensions hybrid piecewise modelling approach can be developed with respect to different modelling variants on topology and kinetic rates. In addition, a grid technique based parallelize methodology can be introduced to improve the sequential simulation process, which can speed up the simulation performance by using multiple processors. Details about the modelling variants and the parallel implementation are illustrated in Chapter 5.

Chapter 5

Variants of Hybrid Modelling Approach

5.1 Introduction

This chapter describes variants of the two dimensions hybrid piecewise modelling approach, including implementation of a parallelization technique, methods of evaluating composed models and synthesized topologies, and modelling variants in terms of topology and kinetic rates. The whole chapter is organized as follows.

Section 5.2 firstly introduces the motivation of parallelizing our proposed 2D hybrid modelling approach. Then the GridGain is applied to parallelize the hybrid modelling and simulation process. Two flowcharts are presented to illustrate assignments of different jobs (mutation of models topologies and optimization of kinetic rates) to different working nodes in the GridGain pool. An example of parallel modelling is investigated to illustrate improved modelling performance by employing the GridGain. The improved performance of hybrid modelling includes reduced simulation time, which is quantitatively measured and discussed by a comparison between the sequential and parallel implementation. Further issues of parallel modelling, for instance idle nodes in the GridGain pool while modelling, are pointed out and discussed. With respect to characteristics of the parallel technique, some potential solutions for addressing aforementioned modelling issues are suggested.

Section 5.3 describes methods to evaluate synthetic models during the models construction. In order to estimate the quality of composed models, behaviour of species in synthetic models are compared with the ones in target biochemical systems. Two methods of computing the behaviour distance are given: *Average* method and *Maximum* method. The average method calculates the mean of behaviour distance among compared species in an objective function, and an average fitness value is returned to represent the quality of analyzed model. The maximum method chooses a species with maximum behaviour difference from a set of compared species, which is only behaviour difference calculated in the objective function, and a fitness value is returned to indicate the quality of the evaluated model. Moreover, regarding the piecewise modelling process, it is possible to obtain species which are generated in synthetic models but not existing in target biochemic system. In this scenario, a mechanism of giving reward and penalty to fitness values in the objective function is included as a complement of behaviour distance measurement based on the Euclidean distance function. The included reward and penalty measurement supports an overall estimation of the generated models during the modelling process.

Section 5.4 introduces exploration of the topologies space by the proposed hybrid modelling methodology. Two mathematical methods are presented to quantitatively measure common interactions between generated and target model. Exploration of topologies space provides an opportunity for obtaining different structures of models for biochemical systems. The models with different interactions among biochemical entities can reveal working mechanisms in biochemical systems which are difficult to observe or verify in wet-lab.

According to specific modelling aims, variants of the proposed hybrid piecewise modelling can be explored, for instance to obtain similar or alternative topologies, desired behaviour and optimized kinetic rates. There is a large variety of ways in which evolutionary methods can be designed for performing genetic operators, comparing species behaviour and evaluating generated models. Section 5.5 presents how to investigate the advantages and disadvantages of some of the variants for the piecewise modelling, with an emphasis on the effect of mutation operators and evaluation criteria of the overall hybrid methods.

Section 5.6 gives a brief summary of the suggested variants of the 2D hybrid piecewise modelling approach. Further discussion about the development of modelling variants is given with simulation results in Chapter 6.

5.2 A GridGain based Parallelized 2D Hybrid Modelling Approach

GridGain [Grid] is a leading JVM-based distributed computing middleware which works on any managed infrastructure. Since first release of GridGain in 2007, GridGain enables users to easily build highly scalable real-time computing and data intensive distributed applications that work on many different infrastructures, such as a small local cluster, private grid, and large private, public and hybrid clouds. Two fundamental technologies are integrated into one product, which supports the co-located parallelization of process and data access:

- Computational Grid
- In-Memory Data Grid

In this thesis, we are interested in applying the computational grid of GridGain to parallelize the hybrid modelling process. Therefore, the computational grid is employed to parallelize the hybrid modelling and details of the computational grid is introduced as follows.

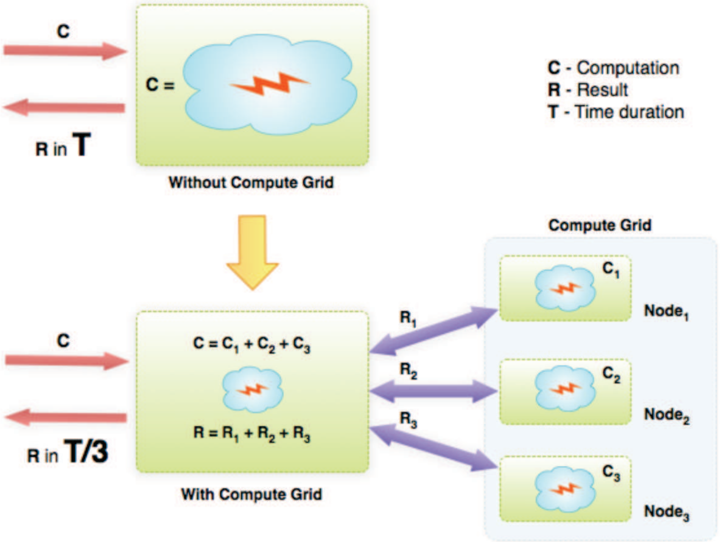


Figure 5.1: Implementation of computational grid in GridGain [Grid].

In general, computational grid technology provides methodologies for distribution of processing logic. Figure 5.1 shows how to split an original one computational task into multiple subtasks, executing these subtasks in parallel on any managed infrastructure and aggregating (reducing) results back to one final result. Compared to the implementation of computation without a grid technique, the final result can be returned in $T/3$ processing time (if there are three nodes in the GridGain pool, and the original total processing time for only one node is T). Therefore, GridGain is one of best parallel environment for parallelizing the hybrid modelling process. The motivation and performance of applying GridGain to develop our 2D hybrid modelling approach is illustrated firstly in following sections.

5.2.1 Motivation

While modelling target biochemical systems, the model candidates in a population pool are independent, before crossing over and mating with other ones. Operations are applied to improve the models in terms of topology and kinetic rates: topology of each individual model is evolved by addition and subtraction operators on ES platform, associated kinetic rates of the model are optimized globally on SA platform, and estimation of mutation and optimization of the model is carried out by mapping PNs of the model to a set of ODEs for simulation. With respect to characteristics of independent models, a parallelization technique can be applied to tackle heavy computation issues existing in sequential simulation process. Details of reasons causing heavy computation in sequential simulation process are described as following:

- Piecewise compose components to models under construction by adding and subtracting operators on ES platform

  A model under construction is presented in PNs format. Addition and subtraction of components requires the import and outport of the PNs model before and after the topology modification, which takes time to update the corresponding vector of models on the modelling platform.

- Globally search for the kinetic rates of each model under construction by fine tuning rates values on SA platform

  Models generated and passed from ES outer layer to SA inner layer are used to optimize the kinetic rates without modifying topologies. These kinetic rates associated with biochemical reactions are fine tuned by employing Gaussian distribution, and corresponding modification on rates values are evaluated by comparing behaviour distance of current optimized model and target model. Both modification and evaluation of kinetic rates are repeated in an iterative manner, which takes time to calculate the real values.

- Iteratively map PNs models to a set of ODEs for evaluating topologies mutation and rates optimization

  All models under construction are described by PNs format as pre-defined in this thesis. An ODE simulator is used to simulate synthetic models to obtain time course data for describing species behaviour in these models. It takes time to map the PNs to a corresponding set of ODEs for quantitatively computing mathematical descriptions of models.

With regard to advantages of parallel technique, the GridGain can improve simulation performance by speeding up the processes of mutating models topologies, searching for kinetic rates values, and mapping ODEs to simulate mathematical models for generating species behaviour data.

5.2.2 Parallelized modelling process

Our hybrid modelling process is improved by using the GridGain to parallelize topologies mutation and kinetic rates optimization. Figure 5.2 shows a pair of sequential and parallelized hybrid modelling process.

Sequential hybrid modelling process applies mutation operators to modify topology of each individual model by ES, and then associated kinetic rates are optimized by using SA. After all individual models are manipulated on topologies and rates, a crossover operator is applied to cut and splice two individuals for generating offsprings in next generation.

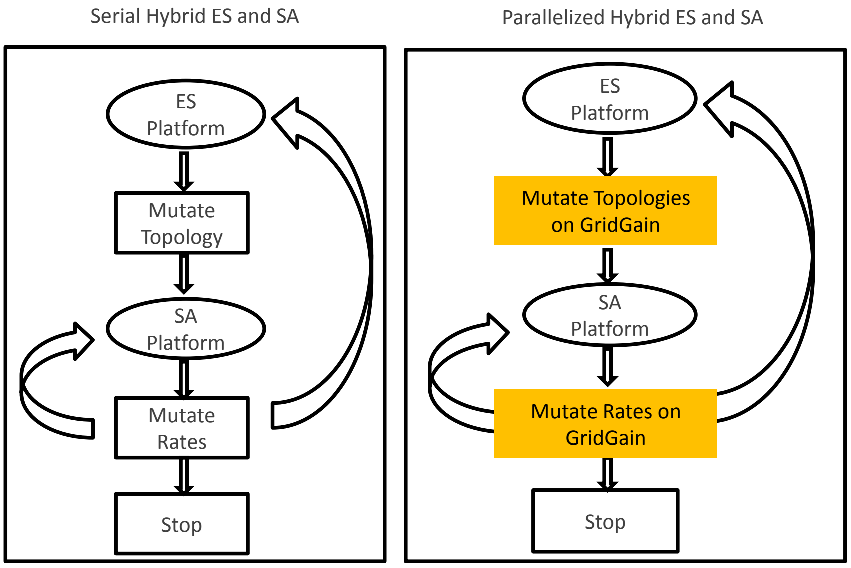


Figure 5.2: Sequential and parallelized hybrid modelling processes.

Parallelized hybrid modelling process allows all individual models to be mutated on topology at the same time on different nodes in a GridGain pool. Associated kinetic rates of each individual model are optimized after the topology mutation, by calling nodes in the GridGain pool. At the end of the parallelized modelling process, all the individual models are copied to each node, and the crossover operation is applied to parallel and genetically produce offsprings on the nodes.

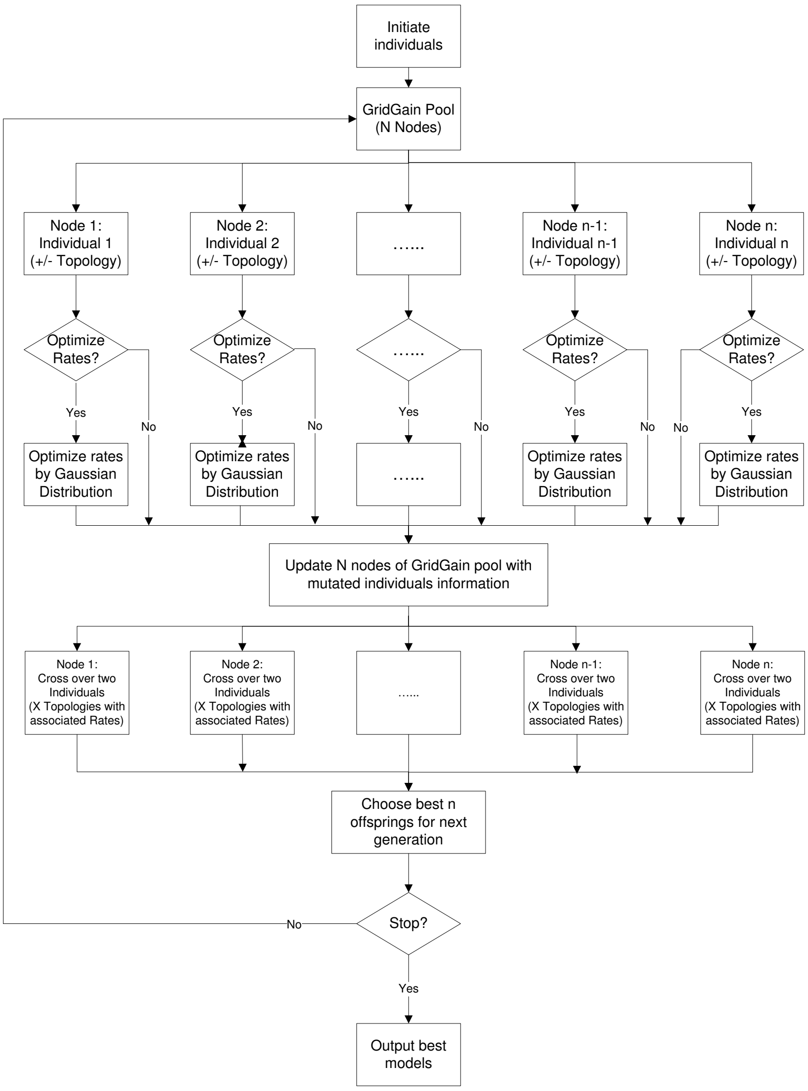

Initiate individuals
GridGain Pool (N Nodes)
Node 1: Individual 1 (+/- Topology)
Node 2: Individual 2 (+/- Topology)
……
Node n-1: Individual n-1 (+/- Topology)
Node n: Individual n (+/- Topology)
Optimize Rates?
Yes
No
Optimize rates by Gaussian Distribution
Update N nodes of GridGain pool with mutated individuals information
Node 1: Cross over two Individuals (X Topologies with associated Rates)
Node 2: Cross over two Individuals (X Topologies with associated Rates)
Node n-1: Cross over two Individuals (X Topologies with associated Rates)
Node n: Cross over two Individuals (X Topologies with associated Rates)
Choose best n offsprings for next generation
Stop?
No
Yes
Output best models

Figure 5.3: Implementation of GridGain to individual models.

Figure 5.3 describes the details of applying GridGain to tackle problems of sequentially mutating and optimizing models under construction in terms of both topology and kinetic rates. The GridGain based parallel modelling contributes to the improvement of constructing biochemical systems by speeding up the process of adding or subtracting topologies and optimizing kinetic rates of models on different nodes at each generation. The estimation of composed models is obtained by firstly mapping PN models to sets of ODEs, and then simulating ODEs based mathematical representation of models in a parallel manner. Before moving from the current evolutionary generation to the next generation, individual models are mated by cut and splice on reactions (containing substructures and rates) to obtain new offsprings. Finally, a set of best generated models is generated at the end of GridGain based parallel hybrid piecewise modelling process.

5.2.3 Parallel performance

In order to evaluate performance of applying GridGain to speed up the modelling process, we have employed the *RKIP* pathway as our test case and carried out five runs of parallel modelling with different number of working nodes in GridGain environment.

Initial setting of running parallel simulation at each run is the same for instantiating components, indicating compared species, running ES and SA algorithms, and applying addition, subtraction and crossover operators. Details of setting are listed as follows: a set of fixed compared species '*RKIP*, *Raf1* and *RKIP—Raf1*'; parameters of SA 'Initial temperature=10, Cooling rate=0.8, Minimum temperature=1, Iteration numbers=10'; initial settings of ES and SA 'Maximum Generations=500, Individuals=50, subtraction at every two generations, crossover with the best model, optimization of rates at every 100 generations, objective function is based on Euclidean distance function'.

The only difference among these five runs is the initial population of model seeds, because population is initiated by randomly selecting components as models seeds from the components library.

Figure 5.4 shows that simulation time can be reduced by using more computing nodes

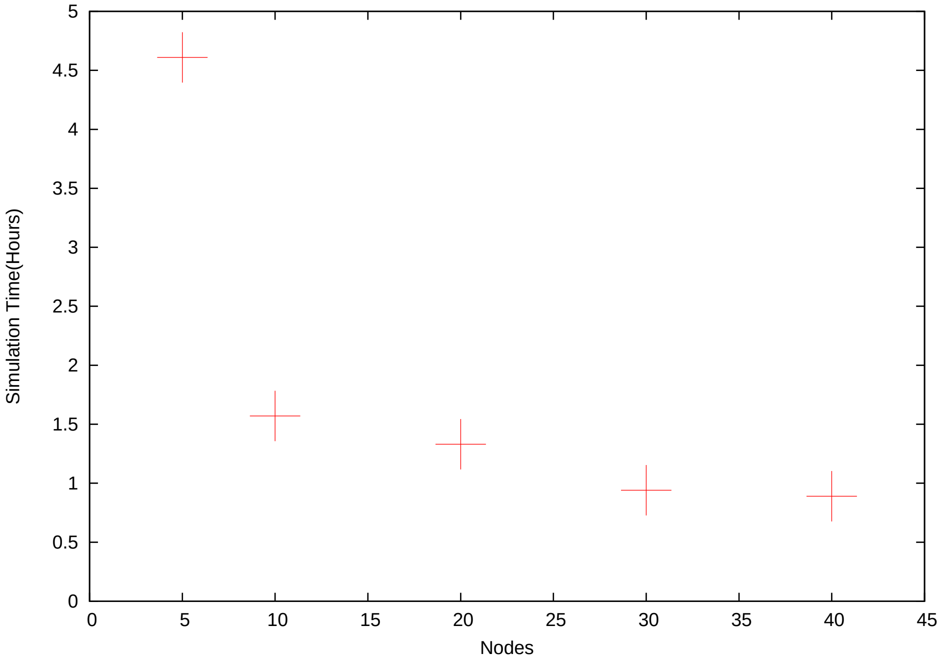


Figure 5.4: Performance of GridGain implementation.

in the GridGain pool. Here, the nodes are cores of processor, which are used as computing nodes in the GridGain pool. A big task can be divided into subtasks according to the number of available working nodes, and then the subtasks on the nodes can be executed to obtain partial results which are integrated into a final result returned for further operations.

5.2.4 Discussion

Parallelization can benefit the simulation process by manipulating the models under construction from the same generation on different nodes in the GridGain pool. While applying GridGain to parallelize the piecewise modelling process, further research can be investigated to address two important issues which exist in current parallel implementation:

1. How to handle ‘idle’ nodes, when loaded jobs are finished on these nodes;

2. How much benefit we can obtain, when the GridGain technique is applied to parallelize a heavy task by splitting a big job to small jobs, executing the small jobs, combining and returning the results.

A working node in the GridGain pool is in charge of dealing with modelling issues: a subtask of mutating topology or optimizing kinetic rates. Due to different size of the model, the processing period on each node is different. Therefore, subtasks on the nodes can be finished in different simulation stages. When the subtasks assigned to the nodes are finished and there is no other subtasks waiting for assignment to be proceed, these nodes are idle in the pool. These idle nodes wait for other busy nodes finishing subtasks assigned to them. When all the subtasks on nodes are finished at current generation, nodes are reset and assigned new subtasks for next run of simulation.

In some extremely scenario, if there is a large models population, only one node still works for the subtask assigned to itself but other nodes are idle, the whole modelling process at current generation is held and the modelling process has to wait for the last busy node finishing the subtask. This scenario makes a low performance of parallel simulation, which degrades the benefit of parallelization.

One of the methods to tackle the above issue of idle nodes is to introduce a cloud technique to assign nodes, according to requirements of subtasks. Moreover, idle nodes can be released for other subtasks. During the modelling process, there are more feasible models generated from a small set of initial model seeds. The mechanism of releasing idle nodes allows different number of nodes on the parallel platform can be used. Adaptive number of available nodes accompanying dynamic (increased or decreased) number of plausible models can broadly explore the solutions space while modelling biochemical systems.

In general, estimation on the cost of executing one big task and the cost of dealing with multiple subtasks can reveal how much improvement we can obtain by utilizing the GridGain parallel technique. Thus if the cost of task assignment is higher than the benefits obtained from parallelization, the implementation of GridGain is not suitable for improving hybrid modelling process.

5.3 Evaluation of Composed Models

A synthetic model is evaluated by comparing its behaviour with target biochemical system. A behaviour is presented by time series data which is measured concentration values of species spaced at uniform time intervals. The species behaviour in a target system can be obtained from a reference biochemical model or by observations of a biochemical system from the wet-lab.

5.3.1 Behaviour comparison

Given a set of reference data for the behaviour of target system M_T, there are *N* generated time series $X_T = (X_1, X_2, ..., X_N)$ which represent the behaviour of *N* species, $N \geq 1$. There are *P* data points in each time series $X_i = (x_i^1, x_i^2, ..., x_i^P)$, $i = 1, ..., N$. There are *M* time series $X_G = (\hat{X}_1, \hat{X}_2, ..., \hat{X}_M)$ describing the behaviour of *M* species in a constructed model M_G, and there are *P* data points for each time series $\hat{X}_j = (\hat{x}_j^1, \hat{x}_j^2, ..., \hat{x}_j^P)$, $j = 1, ..., M$. The intersection between M_T and M_G of species is defined by $X_C = X_T \cap X_G = (X_1, X_2, ..., X_n)$, $1 \leq n \leq N$. Therefore, behaviour difference between the M_T and M_G is calculated by averaging the difference of behaviour of each species in X_C by a paired comparison of the *P* data points.

$$d_{M_T,M_G}(X_k) = \frac{1}{\eta}\sum_{k=1}^{\eta}\sqrt{\sum_{t=1}^{P}(x_k^t - \hat{x}_k^t)^2}. \tag{5.3.1}$$

As shown in Equation 5.3.1, the difference of behaviour for one species X_k, $X_k \in X_C$, is measured by the Euclidean distance function, where η is the total number of compared species in X_C.

Because species in X_C are selected for behaviour comparison, the difference of each compared pair of species behaviour could be in a different scale. There are two ways to compute the overall behaviour distance between generated and target model as final estimated value representing the quality of composed model: *Average* method and *Maximum* method. In general, the average method focuses on the average behaviour distance of all compared species from X_C as the estimated value representing the model under evaluation, but the maximum method chooses the maximum behaviour difference of one species in X_C to represent the quality of the evaluated model. Details of the two methods are given as follows.

5.3.1.1 Average method

The average method allows generated models to be evaluated by measuring behaviour of all involved species in the models without bias. Behaviour distance of each species in a composed model is computed firstly, and then an average value of these behaviour distance is calculated for describing the distance between generated and target model.

As shown in Equation 5.3.1, η is the total number of compared species from a vector X_C. In average method, evaluation of behaviour distance is based on $\eta = |X_C|$. Thus, all the behaviour information of species are utilized during the model evaluation process, which is useful and precise in the scenario of compared species being specified in advance.

5.3.1.2 Maximum method

In maximum method, one species with maximum behaviour difference from the vector X_C is used to evaluate the general distance between composed and target model. Behaviour distance of all the species in X_C is measured firstly, and then the maximum behaviour distance of one species in X_C is utilized in Equation 5.3.1 for measuring the whole composed model, thus $\eta = 1$.

The benefit of using maximum method to evaluate a composed model is to drive the modelling process quickly by rejecting models with the worst performance in terms of furthest behaviour distance. During the initial stages of piecewise modelling, it is easy to generate some species whose behaviour distance is far away from the ones in target model, because of incorrect interactions among species and kinetic rates associated with the biochemical reactions. Therefore, a quick model evaluation could be obtained by avoiding the acceptance of synthetic models which consist of species with furthest behaviour distance.

5.3.2 Reward and penalty

While evaluating the generated model, the species for behaviour comparison can be specified by the user. A vector X'_C can be used to preserve these specified species, where $|X'_C| = n'$ and n' is the number of species in X'_C. Due to indication of compared species in advance, there could be some synthetic entities in a generated model M_G but not in the target model M_T. Therefore, if a substrate is specified for comparison in M_G, whereas the species does not exist in M_T, then M_G should be punished for a constraint of further modelling. If a species for comparison exists both in M_T and M_G, a reward can be given to M_G for an encouragement of correct modelling.

A *Reward and Penalty* function $\Phi(X_k)$ in Equation 5.3.2 is proposed to improve the models evaluation, as the reward and penalty is a complement of the Euclidean distance function for measuring the total behaviour distance.

$$\Phi(X_k) = \begin{cases} -\varepsilon_1, & \text{If } X_k \in X_G \wedge X_k \notin X_T \\ \varepsilon_2, & \text{If } X_k \in X_G \wedge X_k \in X_T \end{cases} \tag{5.3.2}$$

where ε_1 and ε_2 are reward and penalty values, respectively. Both of ε_1 and ε_2 are non negative real values and defined by users at the initial stages. The returned result of $\Phi(x)$ can partly contribute to the final fitness value of a model under evaluation in the objective function $F(x)$ in Equation 5.3.3 as described in following section.

5.3.3 Objective function

With regard to comparison of behaviour difference and a mechanism of reward and penalty, composed models can be evaluated by utilizing Equation 5.3.3 which consists of the Equation 5.3.1 and Equation 5.3.2 for an overall estimation of the fitness during the construction process.

$$f(M_G) = d_{M_T,M_G}(X_k) + \frac{1}{\eta}\sum_{k=1}^{\eta} \Phi(X_k) \tag{5.3.3}$$

where $\eta = n$ if the compared species are from the intersection X_C; and $\eta = n'$ if the compared substrates are from the specific X'_C. In this thesis, modelling of biochemical systems is a minimization problem, therefore the smaller the evaluated fitness value, the better the generated model.

5.4 Exploration of Topologies Space

Alternative topologies can be explored while modelling biochemical systems for understanding the relationships among the compounds. Obtained alternative topologies can be provided to biologists who work in wet-lab to study the suggested models by experimental methods. In our previous work [Wu 12], generation of alternative models has been investigated by employing ES algorithm to explore the models space. A set of alternative topologies with similar behaviour to the target ones has been obtained from our 2D hybrid piecewise modelling approach.

5.4.1 Construction and evaluation of composed topologies

While utilizing the 2D hybrid piecewise modelling approach to construct models for interesting biochemical systems, returned synthetic topologies enable the models exhibiting similar behaviour to the target ones in biochemical systems. Regarding interactions among species in the models, generated topologies can be classified into three categories without respect to values of kinetic rates associated with these interactions:

1. Composed topology is the same as the target one: $T_{Composed} = T_{Target}$;

2. Composed topology covers most of the target one: $T_{Composed} \cap T_{Target} \neq 0$;

3. Composed topology is an alternative topology: $T_{Composed} \neq T_{Target}$.

The models with the same or major parts of a target topologies are usually used to verify the modelling of biochemical systems *in silico*. But there must be primary biochemical knowledge about the biochemical systems in the interactions of species, in order to compare the generated and target model in terms of topology directly. Moreover, the number of species involved in the biochemical system should be a constant, namely no covered species existing in the system. In this scenario, the aim of constructing models for target biochemical systems is to reconstruct structures of the systems and verify the feasible piecewise components composition. Estimation of these synthetic models on topologies is obtained by computing the coverage of interactions among species between the synthetic and target model.

The 2D hybrid piecewise modelling approach allows generation of models with different topologies to the target systems, while exhibiting similar species behaviour to the target ones. In biological experiments, biologists may be interested in biochemical systems with different topologies which produce close behaviour observed on a system level. It is important to investigate and discover different working mechanisms, especial on the multiple regulatory interactions among genes, proteins and complex, for overall understanding the biochemical systems. Therefore, generation of alternative topologies provides an opportunity to unveil the biochemical systems under investigation in an efficient and precise manner.

The evaluation of these three types of generated model topologies can be performed by quantitative estimation in terms of coverage of interactions among biochemical entities. The details of the quantitative evaluation are illustrated in following sections.

5.4.2 Quantitative evaluation of topologies

In order to evaluate the synthetic model structures quantitatively, two measures are employed: *Compression* and *Coverage*. Both measures vary from 0 (worst) to 1 (best). If either compression or coverage is low for a particular model, it indicates the topology of generated model is very different from the target biochemical system, even if their behaviours are similar.

5.4.2.1 Compression

Compression (adapted from [Braz 98] and [Gilb 03]) measures the percentage of matched common arcs between synthetic and target model, which details are given as follows:

$$Compression = \frac{|Intersection|}{Max(|Target|, |Generated|)} \tag{5.4.1}$$

where $|Intersection|$ represents the number of matched arcs between target and generated topology, $|Target|$ is the number of arcs in the target topology, $|Generated|$ denotes the number of arcs in the generated topology, and $Max(|Target|, |Generated|)$ is the bigger number of arcs between the target and generated model.

5.4.2.2 Coverage

Coverage calculates the ratio of matched arcs in the target model and it is given by:

$$Coverage = \frac{|Intersection|}{|Target|} \tag{5.4.2}$$

where $|Intersection|$ represents the number of matched arcs between target and generated topology, and $|Target|$ is the number of arcs in target topology.

5.5 Variants of Hybrid Piecewise Modelling

In our previous research [Wu 12], a 2D hybrid piecewise modelling approach has been proposed and investigated. The 2D hybrid piecewise modelling approach is a hybrid two-layer design applied to model biochemic systems by iteratively assembling components from a user pre-defined library and globally optimizing kinetic rates. The hybrid modelling process is briefly described as follows: firstly, the topologies of models representing biochemical systems are piecewise composed and evolved by utilizing ES algorithm at an outer layer; then SA algorithm is employed at an inner layer to optimize kinetic rates associated with reactions of these synthetic models. Implementations of ES and SA swap, after a predefined number of iterations or generations. At the end of modelling process, a set of best generated models is returned, offering alternative topologies with similar behaviour to the target system.

Regarding different modelling processes in terms of mutating topology and optimizing kinetic rates, variants of the 2D hybrid piecewise modelling can be explored for specific modelling aims, for instance generation of similar or alternative topologies, desired behaviour and optimized kinetic rates. Due to a large variety of ways in which evolutionary methods can be designed, for performing genetic operators, comparing species behaviour and evaluating generated models during the construction process, we investigate the advantages or disadvantages of some variants for the piecewise modelling, with an emphasis on the effect of genetic operators and evaluation criteria of the overall hybrid methods. Five sets of specific modelling variants are compared and general descriptions of these variants are given as follows.

1. Methods related to the data driven, involved in evaluating the composed models:

- Fixed: behaviour of a fixed set of species to be compared
- Dynamic: behaviour of a dynamic set of species to be compared

2. Methods of survival selection:
   - SES: standard (1+1)-evolution strategy
   - PES: probabilistic (1+1)-evolution strategy, probabilistically accept a worse model
3. Methods of applying mutation operator (mutation consists in adding and/or subtracting a component to/from the topology):
   - Fixed: a fixed frequency of switching the addition/removal of a component to/from the model
   - Random: a random way of switching the addition/removal of a component to/from the model
4. Methods of performing crossover:
   - Best: each individual mates with the best individual in the population
   - Random: each individual mates with a randomly selected individual from the population
5. Methods of evaluating generated models in an objective function:
   - ED: the objective function represents the Euclidean distance function
   - ED+RP: the objective function is a combination of a reward and penalty mechanism and the Euclidean distance function

Variants of these five sets are compared in performance of producing high quality models with similar behaviour, best fitness, compression and coverage. Before we demonstrate the details of generated models, these compared variants are described in detail as follows.

5.5.1 Methods of driving models composition

Time series data presenting behaviour of species in a target biochemical system is used to drive the modelling process via reducing the behaviour distance between generated and target model. Given a target biochemical system and a generated model which consist of *N* and *M* species respectively, there are two sets of time series data describing species behaviour in the target and generated model:

$$X_T = (X_1, X_2, ..., X_N), \text{ where } N \geq 1$$
$$X_G = (\hat{X}_1, \hat{X}_2, ..., \hat{X}_M), \text{ where } M \geq 1$$

There is a set of species in a vector X_C which contains species for comparison of behaviour between the generated and target model. It is easy to understand that compared species in X_C can be selected via a fixed or dynamic method: modelers can investigate interesting species in target biochemical system by using a fixed method to drive the modelling process, whereas a dynamic method allows modelers to drive the modelling process by an adaptive manner in terms of matched species between generated and target model.

5.5.1.1 Fixed method

In the fixed method, the species in a fixed set X_C^F are specified by users at initial stage as follows:

$$X_C^F = (X_1, X_2, ..., X_n)$$

where X_i $(i = 1, 2, ..., n)$ is the species assigned for comparison; *n* $(1 \leq n \leq N)$ is a non-variable constant indicating the number of species in X_C^F for the whole model evaluation; *N* is the total number of species in X_T of target biochemical system.

The species specified by user are referred to a target biochemical system. Therefore, all the information (names, concentrations and behaviour in time series data format) of these compared species is provided without uncertainty. Regarding the process of piecewise modelling, a composed model X_G which is constructed at initial stages or evolved by mutation after many generations could only consist of lesser species than the target model. Thus some of the specified species for comparison in X_C^F could be missed in the X_G. In this scenario, the difference between generated and target model will be computed by using an objective function based on Euclidean distance equation or a reward and penalty function which are introduced in Section 5.3.

5.5.1.2 Dynamic method

In the dynamic method, the species for comparison in a dynamic set X_C^D are generated and preserved according to the existence of species in both generated and target models during the modelling process. Thus the species will be the common species from X_T and X_G, which is given as:

$$X_C^D = X_T \cap X_G = \{X_1, X_2, ..., X_N\} \cap \{\hat{X}_1, \hat{X}_2, ..., \hat{X}_M\}$$

The number of species in X_C^D will be a dynamic variable in a range of [0, N]: if there is no common species in both generated and target model, $|X_C^D| = 0$; if all the species in X_T are also generated in X_G, $|X_C^D| = |X_T| = N$; otherwise, $0 < |X_C^D| < N$.

5.5.2 Methods of selecting survival models

Inspired by SA algorithm, a probabilistic evolution strategy (PES) is proposed, which differs from the standard evolution strategy (SES). Regarding the probabilistic mechanism, PES can accept worse models by a probability while searching the solutions space. This may be helpful in avoiding local optima. Theoretically, a global optimum model could be approached for a target system, if an optimization algorithm is run for an enough amount of time.

5.5.2.1 SES method

SES is a traditional evolutionary process, selecting model candidates as offsprings for further evolution in following generations. The criteria for survival models is based on improved fitness. Thus if fitness value of one mutated model is better than the fitness value of the model before mutation, the mutated model with improved fitness values can be survival.

The main process of SES can be referred to Algorithm 5, and the details of selecting offsprings can be illustrated as following: firstly, a model M_t is mutated as a new model M_{t+1}; then models M_t and M_{t+1} are evaluated by an objective function to obtain fitness values $f(M_t)$ and $f(M_{t+1})$, respectively. If $f(M_{t+1}) \geq f(M_t)$, model M_{t+1} survives and replaces model M_t as an offspring for further modelling; otherwise, the mutated model M_{t+1} is rejected and M_t is mutated again for generating a new model mutation for estimation.

5.5.2.2 PES method

PES mimics the natural annealing process, such as a physical process of annealing in metallurgy, for enabling the search of optimum models in a large solutions space. The basic idea of PES is to introduce an acceptance probability into the stages of choosing survival models, which is integrated within the normal model selection stages of SES. Regarding the probabilistic process of SA, it is reasonable to involve a probability of accepting worse models during the modelling process. The search for optimum models can benefit from probabilistic acceptance of worse models, avoiding local optimal traps.

A brief description is given to illustrate the process of accepting and discarding worse models based on a probability during the modelling process:

1. Initiate model seeds in population;

2. For a model M_t in the population, Mutated(M_t) $\rightarrow M_{t'}$;

3. Evaluate(M_t) $\rightarrow f(M_t)$;

4. Evaluate($M_{t'}$) $\rightarrow f(M_{t'})$;

5. Calculate fitness difference Δ C = $f(M_{t'})$-$f(M_t)$;

6. If $\Delta C \geq 0$, Model $M_{t'}$ is an improved synthetic model and $M_{t'}$ is accepted to replace M_t as a new offspring;

7. If $\Delta C < 0$ and $e^{-\frac{\Delta C}{T}} > Random(0, 1)$, Model $M_{t'}$ is a worse synthetic model, but $M_{t'}$ is still accepted to replace M_t as a new offspring;

8. Else Model $M_{t'}$ is rejected and M_t is kept as an offspring;

9. Repeat steps 2 to 8 to mutate, evaluate and compare other models in the population in the same way for generation of other new offsprings.

The probabilistic acceptance of worse models involves the systems temperature T of the PES system and enables the modelling process jumping from local optima to a global optimum. While the system temperature T decreasing, the probability from the $e^{-\frac{\Delta C}{T}}$ should be a decreasing values between 0 and 1, which constrains the acceptance of worse models.

5.5.3 Methods of implementing mutation operators

The mutation operators consist of addition/subtraction of components to/from models. The addition operator is utilized by linking components with existing models, and subtraction operator is used by removing the transitions and associated arcs of the PNs of the components in the models. The addition and subtraction operators applied to mutate the models during the modelling process can be implemented by a fixed method or a random method. The fixed and random methods allow the piecewise modelling to start the composition of components from scratch but with different frequency of adding and subtracting components. The topologies of models under construction can be developed by implementations of addition and subtraction operators.

5.5.3.1 Fixed method

In the fixed method, the two mutation operators can be performed in turn, for instance being applied to the models at every two generations. The fixed method allows users to construct models with simple topologies: defining a high frequency of using the subtraction operator for removing components from the models under construction. Otherwise, complicated models can be developed after performing too many components additions. Moreover, a model under construction should contain at least one component, therefore a single-component based model will be skipped while a fixed method is utilized.

5.5.3.2 Random method

In the random method, addition and subtraction are applied to models at every generation randomly. In this scenario, the topologies of models are composed with more components (species and reactions among these species) or simplified by removing species and linked reactions from the PNs of these models. Mutation of model candidates in the population is randomness, which allows the process of searching optimal topologies without bias. The only issue of randomly applying addition or subtraction operators is that a single-component based model could be mutated by the subtraction operator. Therefore, regarding the constraint of at least one component in the model, the subtraction would not be carried out continually but skipped from a model with only one component.

5.5.4 Methods of performing crossover operator

The crossover operator mates two individual models under construction by a cut and splice method. New offsprings are generated from the combination of parental models in terms of components (reactions and species). The parental models and offsprings compete and only one of them can be survival as a model candidate in the population for evolution in next generation. There are two ways to perform the crossover operator: best and random methods.

5.5.4.1 Best method

In the best method, each model under construction from the population is recombined with a model with best fitness from the same population. It is inspired by the elitism based individuals selection in genetic algorithm. As implemented in genetic algorithm, elitism is a selection method which copies (a set of) best chromosome(s) to new population firstly, and then the rest of chromosomes are selected in other classical ways, such as Roulette Wheel selection, Rank selection and Steady-state selection. The elitism based mechanism of selection can increase the evolutionary performance rapidly, by preventing the lost of best found problem solutions.

The best method of implementing crossover operator mimics the elitism based selection of model candidates. The best method enables the creation of new models population by crossing over an elitist model with other models from the same population. The fitness values of models under construction can converge quickly, because of introduction of best chromosomes from elitist models into the evolved model candidates. Specially, if a model under crossover is a best model in the population while implementing the best method to choose model for crossover, the model will be preserved directly as a survival offsprings for next run of evolution.

One potential problem of applying the best method is that the search easily trapping into local optimal solutions. The models are evolved for mutation with bias of choosing specific elitist models during the construction. If chosen elitist models are local optimal solutions, genetic chromosomes (components with reactions and species) of these models are inherited to offsprings. A promising way for addressing local optimal solutions traps is to employ PES method which is introduced and discussed in previous sections. By using PES, worse and local optimal models are accepted or rejected regarding a probability, which sorts out aforementioned problems of trapping into local optima.

5.5.4.2 Random method

In the random method, each model in the population will be crossed over with another model chosen randomly from the same population without considering the fitness. The crossing over between two models for generation of offsprings follows the mechanism of random selection in nature. It is feasible to approach optimal models by evolving model candidates in a reasonable number of generations, with respect to successful implementation of evolutionary algorithms to drive the modelling process in computational biology.

While applying the random method to choose a model for crossover, it is easy to choose a model itself for the crossover, especially in a small size population. Therefore, if a current evolved model is selected randomly for crossing over with itself, the random model selection will be executed again until a different model being reached in the same population. This mechanism of crossover between different models prevents modelling process from applying meaningless operations to evolve models, because it does not benefit the evolution by swapping components from the model itself.

5.5.5 Methods of evaluating models

The difference between generated and target model is calculated by employing an objective function. In the objective function, there are two methods of evaluating the composed models: a Euclidean distance (ED) based method, and a Euclidean distance with a reward and penalty mechanism (ED+RP) based method. These two evaluation methods can deal with estimations of models involving compared species which are not both existent in generated and target model. Evaluation in the objective function is based on a classical estimation of behaviour difference which is computed between two sets of time series data representing behaviour of generated and target model.

5.5.5.1 ED method

As mentioned in Section 5.3, a basic evaluation method is to calculate the behaviour distance of species in generated and target model by employing traditional ED equation. The ED is an ordinary distance between two points on the time series data for the species behaviour from generated and target model. Moreover, the distance between the two points on the behaviour data is the absolute value of their numerical difference.

Therefore, several points on a pair of time series data sets for one species behaviour between generated and target model can be specified for the measurement, for instance every specific simulation time in minutes and corresponding species concentration. These specific behaviour data points are used to quantitatively estimate the difference of generated and target model in terms of one specified species behaviour. Other species behaviour could be included and calculated in the objective function based on ED equation for the models evaluation. In this scenario, the objective function can include the overall calculation of behaviour difference among all the given species behaviour in different sets of time series data.

The premise of applying ED equation to the models evaluation is that all the compared species should both exist in generated and target model. With respect to the piecewise modelling process, there is a chance that some synthetic models do not consist of specified species for comparison during the models construction. Therefore, a sophisticated evaluation method should be developed, for instance giving a penalty to invalid compared species. The development of models evaluation with reward and penalty is described in following section.

5.5.5.2 ED+RP method

A formal model estimation method involving a mechanism of giving reward or penalty to generated models is defined and illustrated in Section 5.3. The inclusion of the reward and penalty in an objective function is intended to prioritize individual models whose components are among the ones existing in the target model. For instance, if a species is generated in a synthetic model and the species is also among the ones existing in the target model, fitness will be improved by giving a reward value; otherwise, the fitness will be penalized by giving a penalty. Regarding different behaviour scales of target biochemical systems under construction, different values of the reward and penalty can be implemented. According to our preliminary experiments, we choose 0.01 and 1000 as the reward and penalty values respectively in our cases study.

5.6 Summary

In this chapter, we have presented variants of the 2D hybrid piecewise modelling approach in details. The developments of modelling approaches include implementation of a grid technique to parallelize the sequential modelling and simulation process, two mathematical methods of evaluating constructed models on the topologies, and variants of the modelling in terms of topologies mutation and kinetic rates optimization.

The basic aim of applying the GridGain to modelling process is to improve the performance of simulation. Because it takes time to calculate the mapped ODEs of the composed models and to estimate the mutations of kinetic rates in these models, the modelling process can be very slow. The GridGain can support the assignment of different modelling jobs, for instance mutating topologies, optimizing kinetic rates and mating models from the same population, to working nodes in the GridGain pool for a parallel jobs execution. The jobs on the nodes are executed independently and results from the nodes are summarized for further operations. Therefore, sequential modelling process can be improved by the GridGain to obtain good modelling performance. A parallel case study with simulation results is given to demonstrate the improved performance based on the GridGain.

Composed models can be evaluated by different methodologies, for instance regarding specific species or all the species in a model. These different methods of evaluating synthetic models support the investigation of specific species in target biochemical systems, whereas it is difficult for biologists in wet-lab to perform the same species estimation.

Regarding complicated mechanisms in biology and high interactions among biochemical entities, it is difficult to investigate topologies of biochemical systems in a biological experiments manner. Therefore, it is necessary to explore the topologies space, for obtaining knowledge of target biochemical systems in terms of signalling cascades and reactions rates. In order to measure the quality of generated models topologies, two mathematical measurements are used to calculate the ratio of common arcs between generated and target model.

This chapter describes variants of the hybrid piecewise modelling in terms of modelling topology and optimizing kinetic rates with different criteria. These variants are proposed and illustrated in details of working mechanisms. The advantages and disadvantages of these proposed variants are investigated by comparing and analyzing simulation results obtained from implementations of these variants in Chapter 6.

Chapter 6

Cases Study

6.1 Introduction

In this chapter, we have applied the 2D hybrid piecewise modelling approach to model two signalling pathways. Synthetic models of two given signalling pathways can be composed automatically from scratch, driven by target behaviour of the pathways.

We evaluate synthetic models by comparing similarity of behaviour of species in the composed and target model, analyzing the convergence of fitness values of synthetic models, and calculating compression and coverage scores of synthetic models for quantitative analysis. Moreover, we have shown that alternative models topologies of given signalling pathways can be obtained by employing the 2D hybrid piecewise modelling approach. In biology, alternative structures of biochemical systems are always important and valuable for understanding the signalling transduction paths.

We developed the 2D hybrid piecewise modelling approach in Chapter 5 by considering different variants, for instance different implementation of target data driven, individuals selections, mutation operators and models estimation methods. Synthetic models are composed by utilizing different implementations of modelling variants and their combinations.

In this chapter, we statistically analyze synthetic models composed by five paired modelling variants. A summary of the performance of these compared different modelling variants in terms of generating similar or alternative topology and similar behaviour is given. Conclusions about effects of modelling variants focusing on specific modelling aspects describe whether a modelling variant performs better, worse or the same as another one it is directly compared with.

6.2 RKIP Pathway

Signalling pathways play a pivotal role in many key cellular processes [Elli 02]. The abnormality of cell signalling can cause uncontrollable division of cells, which may lead to cancer. There is one of the most important and intensively studied signalling pathways: *ERK* pathway (the *Ras/Raf-1/MEK/ERK* signalling pathway) which transfers the mitogenic signals from the cell membrane to the nucleus [Yeun 00]. The *ERK* pathway is de-regulated in various diseases, ranging from cancer to immunological, inflammatory and degenerative syndromes and thus represents an important drug target.

A brief illustration of regulations among proteins and complex based on signalling transduction in the *ERK* pathway is given as follows. *Ras* is activated by an external stimulus, via one of many growth factor receptors; it then binds to and activates *Raf-1* to become *Raf-1**, or activated *Raf*, which in turn activates *MAPK/ERK Kinase* (*MEK*) which in turn activates *Extracellular signal Regulated Kinase* (*ERK*). Cell differentiation is controlled by following cascade of protein interactions: *Raf-1* → *Raf-1** → *MEK* → *ERK*.

The effect of regulation is dependent upon the activity of *ERK*. The *Raf-1* kinase inhibitor protein (*RKIP*) inhibits the activation of *Raf-1* by binding to it, disrupting the interaction between *Raf-1* and *MEK*, thus playing a part in regulating the activity of the

ERK pathway [Yeun 99]. A number of computational models have been developed in order to understand the role of *RKIP* in the pathway and ultimately to develop new therapies [Cho 03, Cald 04].

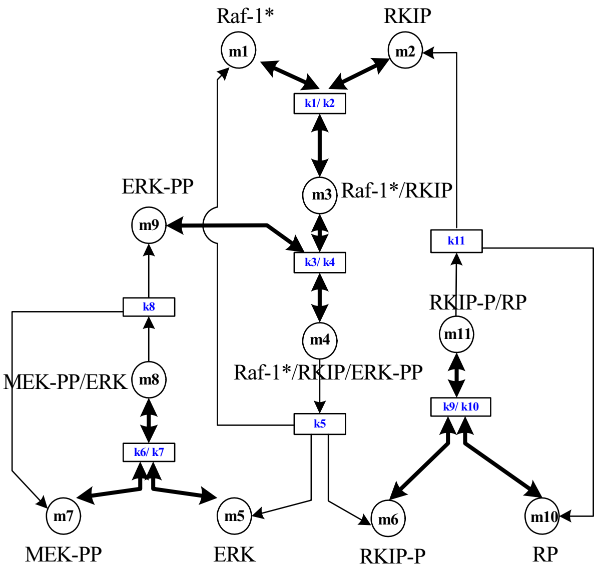


Figure 6.1: A graphical representation of the *ERK* signaling pathway regulated by *RKIP*, reproduced from Cho [Cho 03]: a circle represents a state for the concentration of a protein and a bar indicates a kinetic parameter of reaction to be estimated. The directed arc (arrows) connecting a circle and a bar represents a direction of a signal flow. The bi-directional thick arrows represent an association and a dissociation rate at same time. The thin unidirectional arrows represent a production rate of products.

A concrete example, the '*RKIP* pathway' which is a subset of the *ERK* signalling pathway, is employed as our first case study in this thesis. A graphical PNs representation of the *RKIP* pathway is shown in Figure 6.1 which is suggested by Cho et al. [Cho 03]. We employed this graphical *RKIP* pathway as a target biochemical system for testing our hybrid piecewise modelling approach. In Figure 6.1, a state of a protein concentration is represented by a circle; a bar indicates a kinetic parameter of a biochemical reaction to be estimated; A direction of a signal flow between protein and reaction is illustrated by a directed arc connecting the circle and bar; association and disassociation rates are represented by the bi-directional thick arrows, and the thin unidirectional arrows represent a production rate of products.

Simulation results suggest that it is feasible to employ our 2D hybrid piecewise modelling approach with its variants to model biochemical systems from scratch and obtain models with similar or alternative topologies exhibiting similar behaviour as the ones in the target biochemical systems. Analysis of simulation results is illustrated in details as follows.

6.2.1 Generation of similar behaviour

One of main aims of applying the hybrid methodology to model target biochemical systems is to construct synthetic models which exhibit similar behaviour to the ones in target biochemical systems. In our simulations on the test case '*RKIP* pathway', a group of best models is generated by piecewise composing components to a set of given model seeds under construction, and evolving the composed models in terms of topology and kinetic rates.

Similar behaviour of species among these synthetic models are obtained, regarding species behaviour given in the target *RKIP* pathway. There are 11 species in the target *RKIP* pathway, but more or less proteins or complex could be generated in the composed models, with respect to piecewise modelling process. We mainly compare the behaviour of species existing in both generated and target model. The similarity of compared species behaviour are shown in the following figures. Some behaviour of species of composed models from a group of best returned models are very similar to the target ones. But some behaviour of species from a small subgroup of returned models are not similar, due to different topologies and kinetic rates in these generated models.

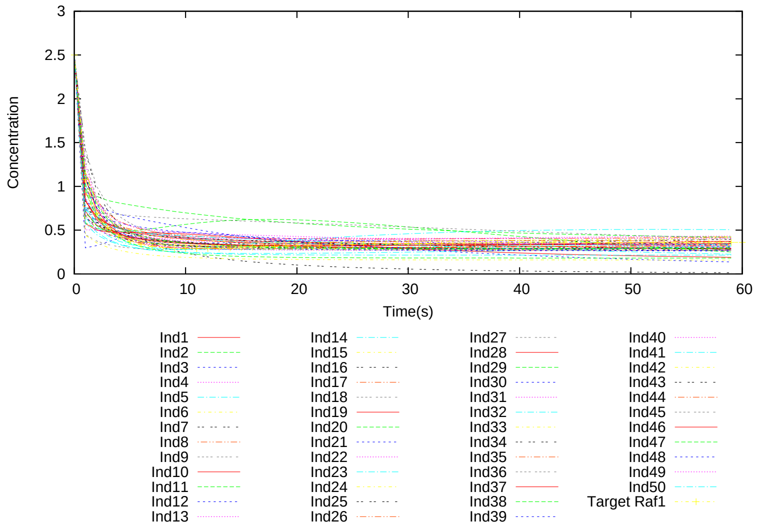


Figure 6.2: Comparison of species *Raf1* behaviour.

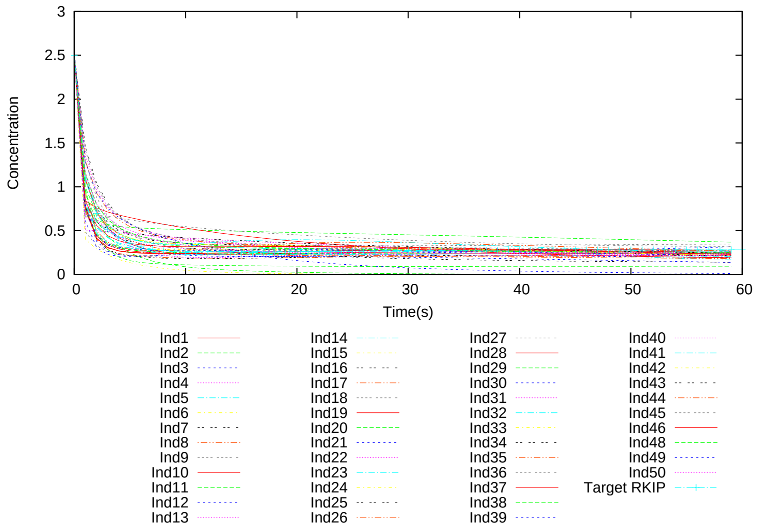


Figure 6.3: Comparison of species *RKIP* behaviour.

Figure 6.2 and Figure 6.3 show comparison of behaviour of species *Raf1* and *RKIP* between target RKIP pathway and 50 generated models. From the diagrams, it is clear that most of synthetic models exhibiting similar behaviour of species *Raf1* and *RKIP* to the target ones in *RKIP* pathway.

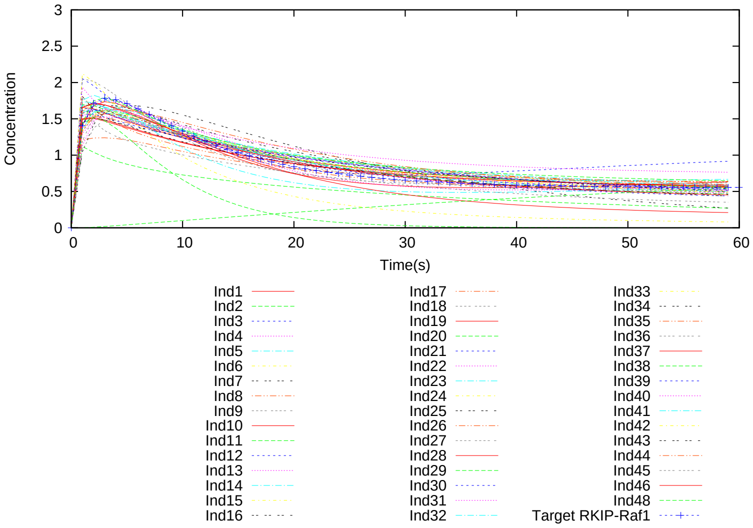


Figure 6.4: Comparison of species *RKIP*|*Raf1* behaviour.

Because biochemical reactions between species *Raf1* and *RKIP* are very important for signal transduction in the pathway, it is necessary to investigate interactions among species *Raf1*, *RKIP* and the complex *RKIP*|*Raf1* formed by binding *Raf1* and *RKIP*. The interactions can be described in the following two biochemical reactions: binding reaction '$Raf1 + RKIP \rightarrow RKIP|Raf1$' and unbinding reaction '$Raf1 + RKIP \leftarrow RKIP|Raf1$'.

Moreover, behaviour of complex *RKIP*|*Raf1* provided from the target *RKIP* pathway is one of the species behaviour for driving during the modelling process. Composed models can be investigated for generation of the two binding and unbinding reactions by comparing the behaviour of species *RKIP*|*Raf1*.

Figure 6.4 shows that most of generated models exhibit similar behaviour of the complex *RKIP*|*Raf1* as the target one. The generation of nonsimilar *RKIP*|*Raf1* behaviour in the figure suggests that two binding and unbinding reactions may be interrupted by other biochemical reactions associated with *Raf1* and *RKIP* in corresponding composed models, which could be investigated for the details in terms of topology.

In *RKIP* pathway, the same mechanism of binding and unbinding interactions exists in two biochemical reactions between species *ERK* and *MEKPP*: '$ERK + MEKPP \rightarrow ERK|MEKPP$' and '$ERK + MEKPP \leftarrow ERK|MEKPP$'.

As shown in Figure 6.5, Figure 6.6 and Figure 6.7, behaviour of species *ERK*, *MEKPP* and complex *ERK*|*MEKPP* in generated models from the hybrid piecewise modelling framework are also similar to the target ones.

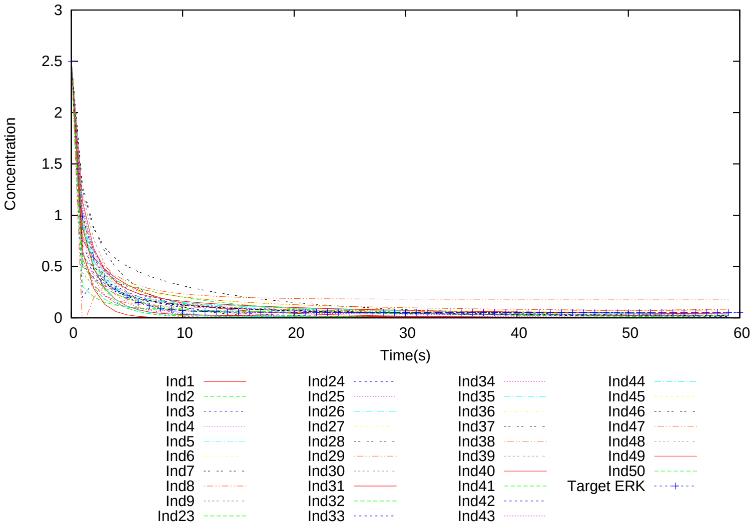


Figure 6.5: Comparison of species *ERK* behaviour.

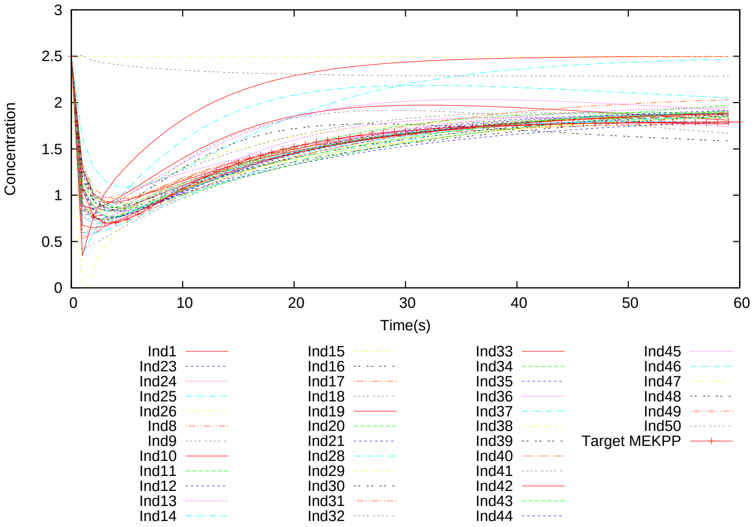


Figure 6.6: Comparison of species *MEKPP* behaviour.

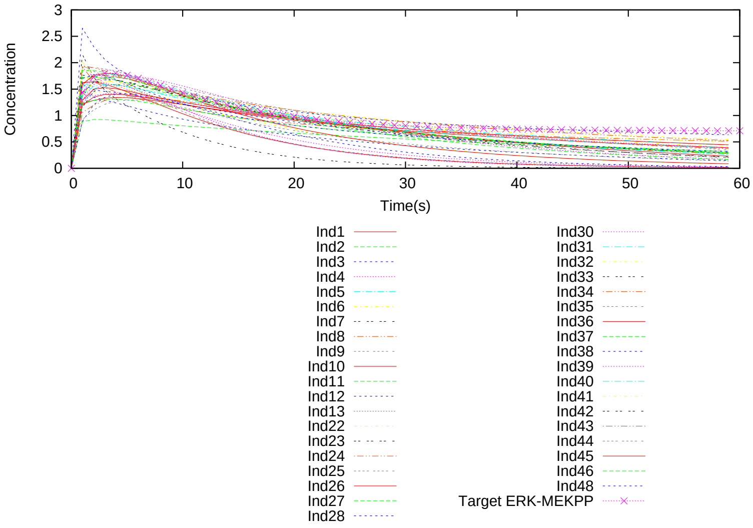


Figure 6.7: Comparison of species *ERK*|*MEKPP* behaviour.

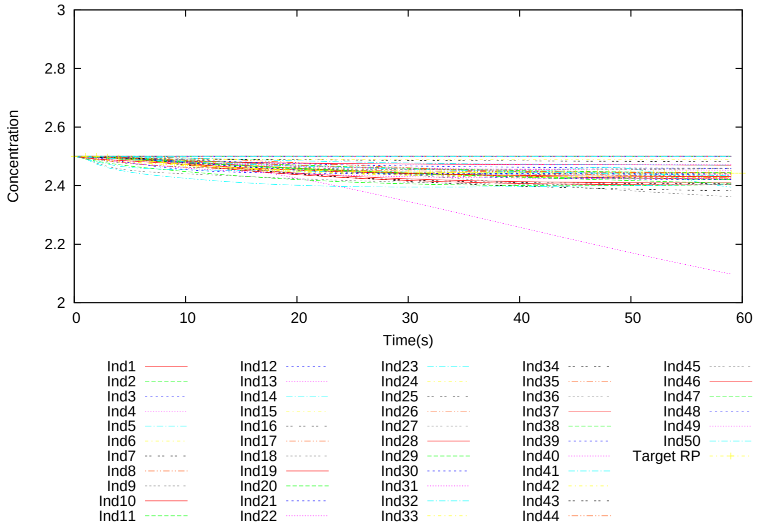


Figure 6.8: Comparison of species *RP* behaviour.

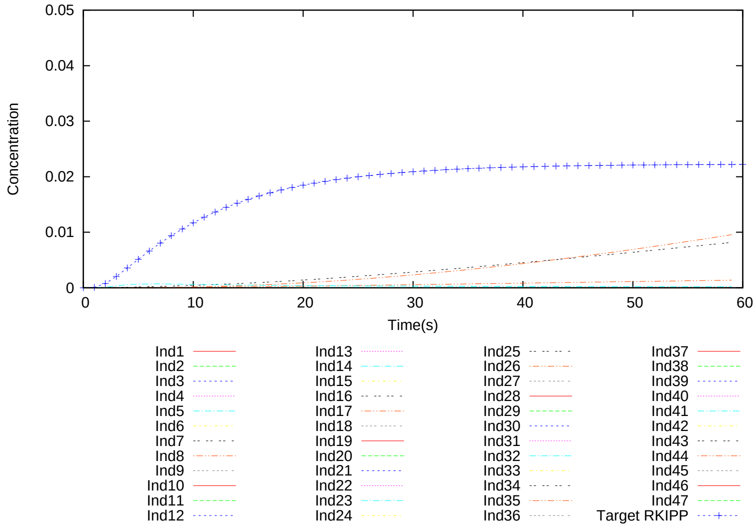


Figure 6.9: Comparison of species *RKIPP* behaviour.

Regarding species *RP* and *RKIPP* involved in two binding and unbinding reactions, there should be similar species behaviour of *RP* and *RKIPP* exhibited in generated models. Figure 6.8 shows that species *RP* behaviour among most of returned best models are similar to the target one in *RKIP* pathway.

But the species *RKIPP* behaviour in Figure 6.9 indicate that just about four composed models exhibiting species *RKIPP* behaviour, and other composed models can not generate similar species *RKIPP* behaviour because concentrations of species in these models are zero during the whole simulation time as shown in the Figure 6.9.

The reason of resulting missed similar species behaviour could be some extra interactions existing in the composed models. These extra interactions are not the ones in target *RKIP* pathway, which may have influence on the association and/or disassociation of species *RKIPP* during the simulation *in silico*. That is why generated models exhibit different *RKIPP* behaviour, event though the binding and unbinding reactions has been generated.

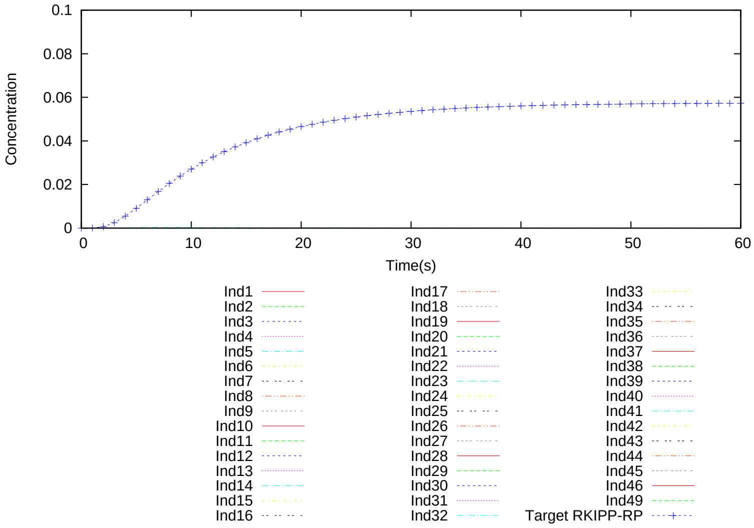


Figure 6.10: Comparison of species *RKIPP*|*RP* behaviour.

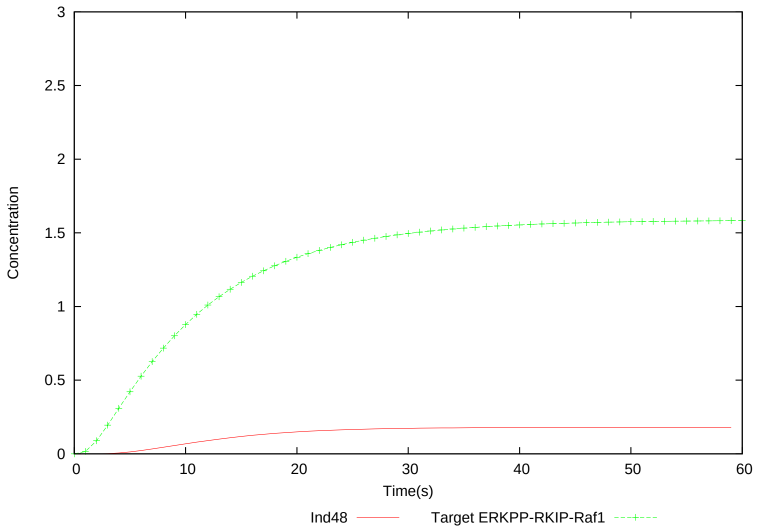


Figure 6.11: Comparison of species *ERKPP*|*RKIP*|*Raf1* behaviour.

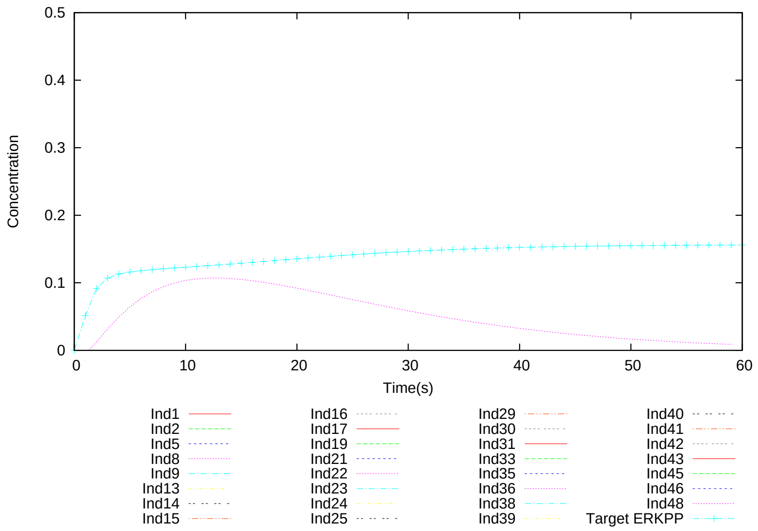


Figure 6.12: Comparison of species *ERKPP* behaviour.

Furthermore, the existing extra interactions in the composed models may have an effect on the generation of some target complex, because the target complex cannot be produced if its forming materials (proteins and other complex) are inhibited or not produced by the extra interactions. For instance, the generation of complex *RKIPP*|*RP* relies on biochemical binding and unbinding reactions $RKIPP + RP \rightarrow RKIPP|RP$ and $RKIPP + RP \leftarrow RKIPP|RP$. If the species *RKIPP* is not obtained correctly in the composed models (for instance, *RKIPP* behaviour is missed in Figure 6.9), the generation of complex *RKIPP*|*RP* is affected and behaviour of *RKIPP*|*RP* is not exhibited in the composed models, as shown in Figure 6.10.

The same problems of missed similar species behaviour happen to *ERKPP*|*RKIP*|*Raf1* and *ERKPP*, due to extra biochemical reactions or missed interactions among the species and complex. As shown in Figure 6.11 and Figure 6.12, only one synthetic model exhibits species *ERKPP*|*RKIP*|*Raf1* and *ERKPP* behaviour respectively. The behaviour of species *ERKPP*|*RKIP*|*Raf1* and *ERKPP* are still far away from the target ones.

After comparing species behaviour in the composed models with corresponding ones in target biochemical system, it is feasible to generate models presenting similar species behaviour in time series data format. But regarding lack of similar species behaviour in the synthetic models, these obtained best models should be studied by comparison with the target biochemical system in terms of topology, in order to validate or improve the quality of synthetic models generated by the 2D hybrid piecewise modelling approach.

6.2.2 Convergence of composed model fitness

The piecewise construction of models can be driven to approach the target RKIP pathway by improving the fitness. Composed models can be evaluated for returning estimated fitness value for each model, and the returned fitness value should converge with increasing number of running generations during the modelling process.

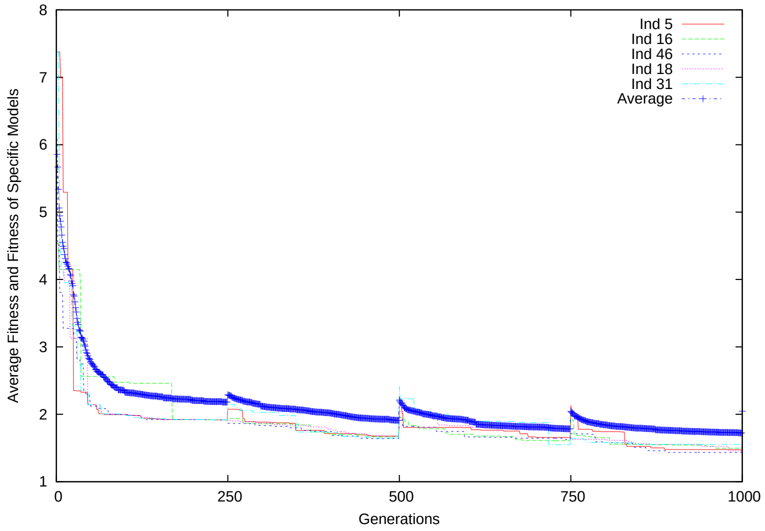


Figure 6.13: Average and five best fitness values of synthetic *RKIP* models.

As shown in Figure 6.13, there is an average fitness value for 50 synthetic models, converging to a minimum value with the increased number of generations in the simulation. In our current implementation, the hybrid piecewise modelling process is set to call the SA layer to optimize the kinetic rates of each model at every 250 generations within total pre-defined 1000 running generations for the simulation. Due to the probabilistic mechanism of accepting a worse solution by SA, there is a jump of average fitness convergence for the models at each end of run of calling SA layer to optimize the kinetic rates. The average fitness value converges again after move back to ES layer, following a traditional evolutionary process, until reaching the end of simulation.

Moreover, in order to investigate the fitness convergence for each developed model, we choose to analyze the fitness convergence among five synthetic models from 50 composed models. In Figure 6.13, fitness values of the five best models converge as the average one with increased number of generations, and jump at each run of calling SA layer. Thus a group of returned best models from the modelling framework is close to the target biochemical pathway in terms of behaviour measurement based on Euclidean distance function.

6.2.3 Quantitative analysis of composed topologies

6.2.3.1 Compression

Figure 6.14 illustrates the compression scores from comparison between the 50 synthetic models and target RKIP pathway in terms of topology. These composed 50 models are from one run based on the same simulation setting of the hybrid piecewise modelling framework. Here we attempt to compare the generated models with target biochemical pathway in terms of matched arcs (interconnections among species or complex), for illustrating quantitative analysis on the topologies of composed model.

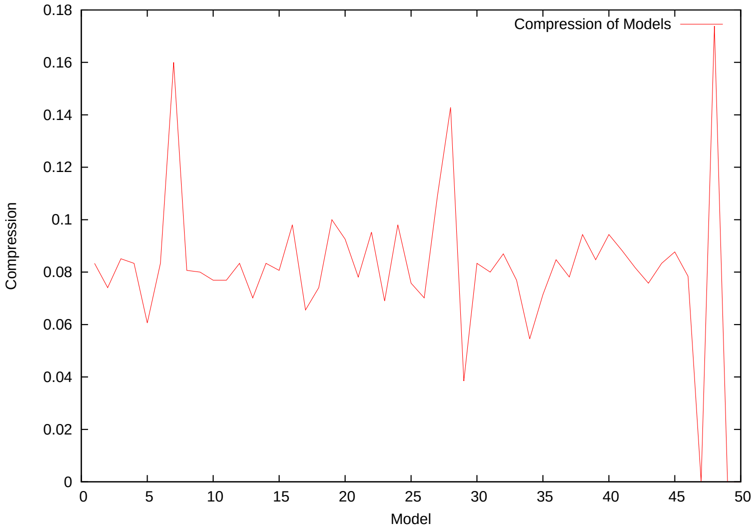


Figure 6.14: Compression analysis of the synthetic topologies.

As shown in Figure 6.14, compression scores of the synthetic models are very poor, ranging over [0, 0.18]. There are even two composed models which including no matched arcs on the topologies, compared to the target structure of *RKIP* pathway. According to the definition and description of compression in Section 5.4.2, low compression score means less matched topologies in synthetic models, which indicates generation of models with various structures. In wet-lab, biologists might be interested in models with different topologies but exhibiting similar behaviour. Thus, these composed models with low compression scores can be provided to biologists for further experimental investigation.

6.2.3.2 Coverage

Quantitative analysis on generated model in terms of topology can be performed by computing coverage scores of these models, as an complementary measurement to the analysis based on compression.

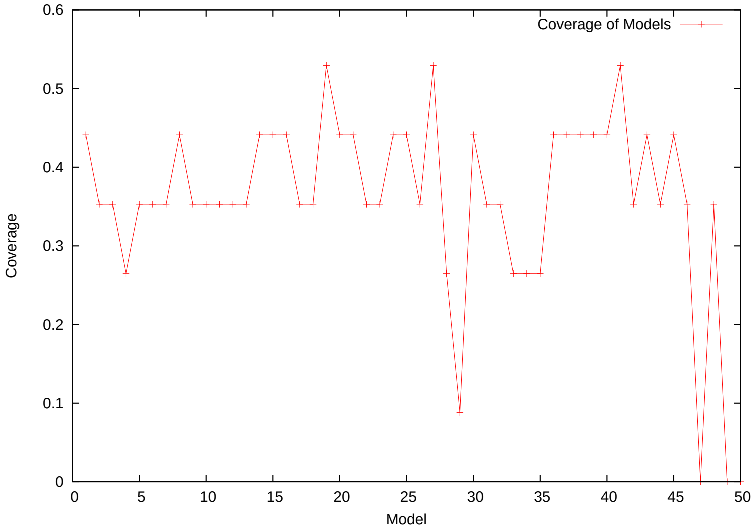


Figure 6.15: Coverage analysis of the synthetic topologies.

Figure 6.15 shows that most of coverage scores of synthetic models for target *RKIP* pathway is in the ranges of [0, 0.53], including two models with zero coverage score as the estimation of compression. Regarding the low compression and coverage scores for these generated models, we can say models obtained from the hybrid modelling framework are very different to the target *RKIP* pathway in terms of topology. Therefore, these different models are obtained and preserved as a good resource for biological research in wet-lab.

6.2.4 Generation of alternative topologies

In order to illustrate generation of different topologies in synthetic models, we compared one of generated models from our simulation with target *RKIP* pathway in terms of reactions. By analyzing how many reactions in target pathway can be generated in the composed model, we can quantitatively measure difference of the alternative topology compared with target one.

Table 6.1: Comparison of one synthetic model with *RKIP* pathway.

Reactions in *RKIP* pathway	Reactions in One Generated Model
*$Raf1 + RKIP \xrightarrow{k1} RKIP\vert Raf1$	$ERK\vert RP \xrightarrow{r1} ERKP + RP$
*$RKIP\vert Raf1 \xrightarrow{k2} Raf1 + RKIP$	$ERKPP\vert MEKPP \xrightarrow{r2} ERKPP + MEKPP$
$RKIP\vert Raf1 + ERKPP \xrightarrow{k3} ERKPP\vert RKIP\vert Raf1$	$ERK\vert RP + ERKPP\vert RKIPP \xrightarrow{r3} ERK\vert ERKPP\vert RKIPP\vert RP$
$ERKPP\vert RKIP\vert Raf1 \xrightarrow{k4} RKIP\vert Raf1 + ERKPP$	$ERK + RKIP\vert Raf1 \xrightarrow{r4} ERK\vert RKIP\vert Raf1$
$ERKPP\vert RKIP\vert Raf1 \xrightarrow{k5} Raf1 + ERK + RKIPP$	*$RKIP + Raf1 \xrightarrow{r5} RKIP\vert Raf1$
*$ERK + MEKPP \xrightarrow{k6} ERK\vert MEKPP$	*$ERK + MEKPP \xrightarrow{r6} ERK\vert MEKPP$
*$ERK\vert MEKPP \xrightarrow{k7} ERK + MEKPP$	$ERKPP\vert MEKPP + MEKPP\vert RKIPP \xrightarrow{r7} ERKPP\vert MEKPP\vert RKIPP$
$ERK\vert MEKPP \xrightarrow{k8} MEKPP + ERKPP$	$RKIP + ERK\vert RP \xrightarrow{r8} ERK\vert RKIP\vert RP$
$RKIPP + RP \xrightarrow{k9} RKIPP\vert RP$	*$RKIP\vert Raf1 \xrightarrow{r9} RKIP + Raf1$
$RKIPP\vert RP \xrightarrow{k10} RKIP + RP$	$ERK\vert MEKPP \xrightarrow{r10} ERKP + MEKPP$
$RKIPP\vert RP \xrightarrow{k11} RKIPP + RP$	$RKIP\vert Raf1 + ERKP \xrightarrow{r11} ERKP\vert RKIP\vert Raf1$
	*$ERK\vert MEKPP \xrightarrow{r12} ERK + MEKPP$

As shown in Table 6.1, four reactions marked with star in target *RKIP* pathway are generated in a synthetic model. The synthetic model consists of 12 reactions that four of them being identical to the ones in *RKIP* pathway. Regarding a low coverage score of the compared synthetic model, we can find that the hybrid modelling framework can obtain alternative topologies of composed models exhibiting similar behaviour to the target ones in biochemical systems.

Alternative topologies in synthetic models illustrate target biochemical system in a different way, providing templates to biologists in wet-lab for further experimental examination at the properties of the biochemical systems.

6.3 Levchenko Pathway

In biochemical systems, in addition to preventing crosstalk among related signaling pathways, scaffold proteins might facilitate signal transduction by preforming multimolecular complexes that can be rapidly activated by incoming signal. In many cases, such as mitogen-activated protein kinase (*MAPK*) cascades, scaffold proteins are necessary for full activation of a signalling pathway [Levc 00].

Levchenko et al. investigated a quantitative computer model of *MAPK* cascade with a generic scaffold protein to suggest a detailed biochemical model of scaffold action. From the analysis of the suggested model, Levchenko et al. show that specificity, efficiency and amplitude of signal propagation can be regulated by using formation of scaffold-kinase complexes.

In this thesis, the model studied by Levchenko et al. [Levc 00] is employed as our second test case; details of the model can be obtained from BioModels database(Model NO. BIOMD0000000011) [Li 10]. We call the utilized model the Levchenko2000 model. Figure 6.16 shows the structure of the Levchenko2000 model.

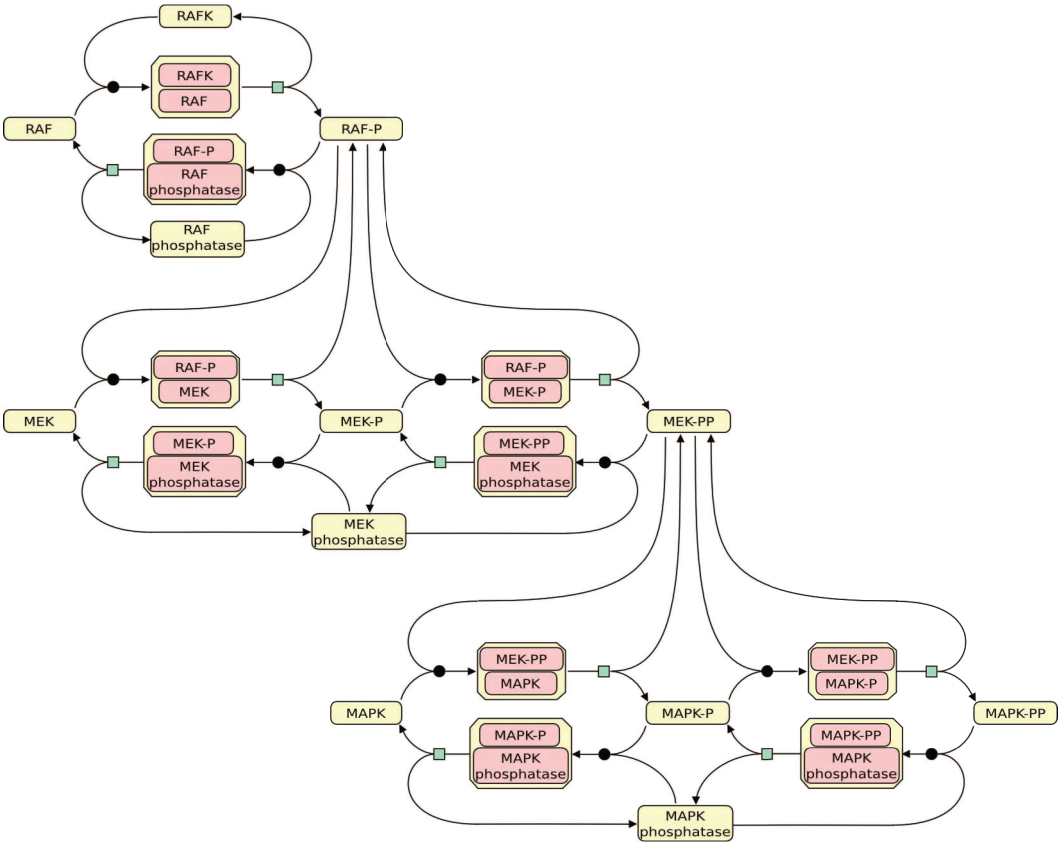


Figure 6.16: Three signalling cascades of Levchenko2000 model, reproduced from [Li 10]. This is a representation of the signalling cascades, not the Petri net.

6.3.1 Generation of similar behaviour

Similar species behaviour in composed models of Levchenko2000 are shown in figures. Figure 6.17 to Figure 6.23 show the generated models with similar behaviour of species *Raf*, *RafP*, *RasGTP*, *Raf*|*RasGTP*, *Phase3*, *MEK* and *MEKP* to the target ones for presenting *MAPK* cascades signalling pathway.

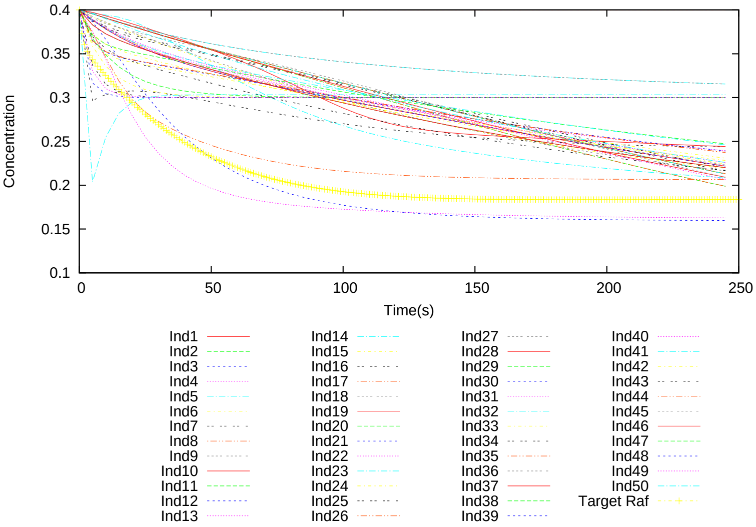


Figure 6.17: Comparison of species *Raf* behaviour.

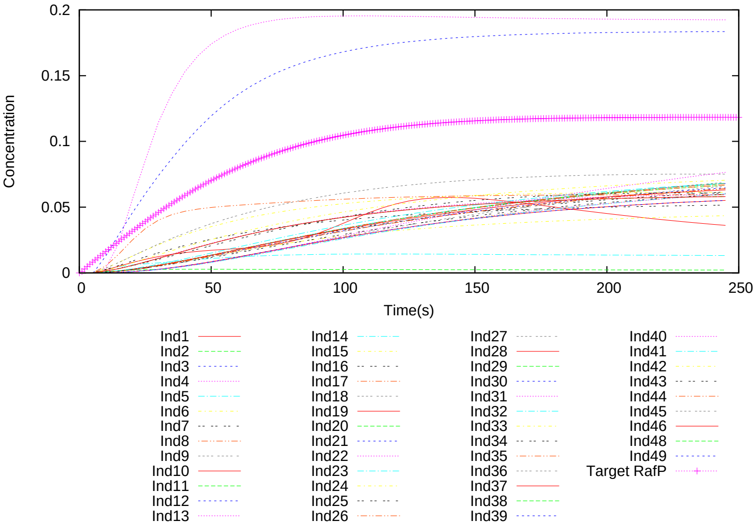


Figure 6.18: Comparison of species *RafP* behaviour.

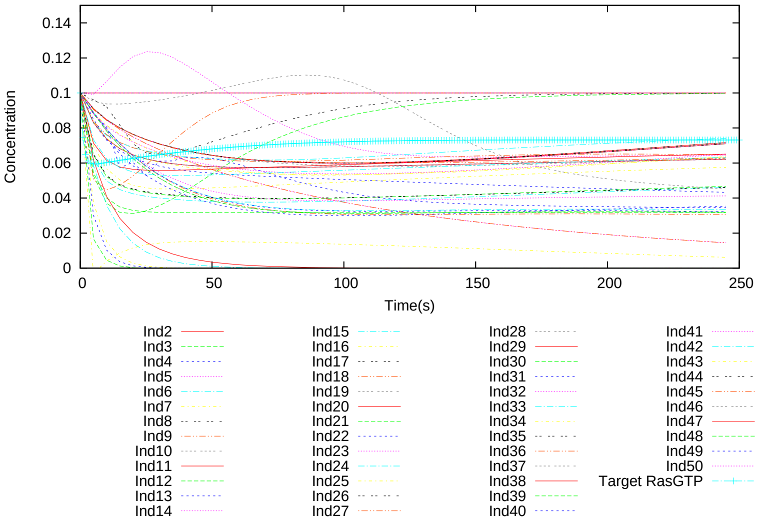


Figure 6.19: Comparison of species *RasGTP* behaviour.

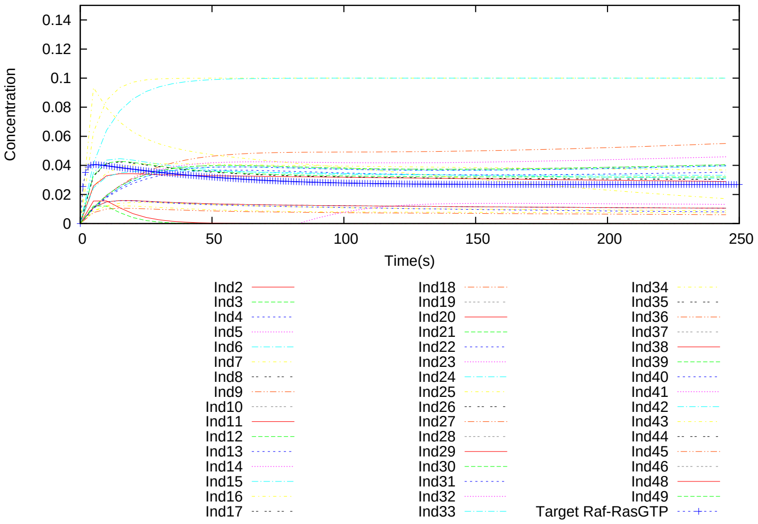


Figure 6.20: Comparison of complex *Raf*|*RasGTP* behaviour.

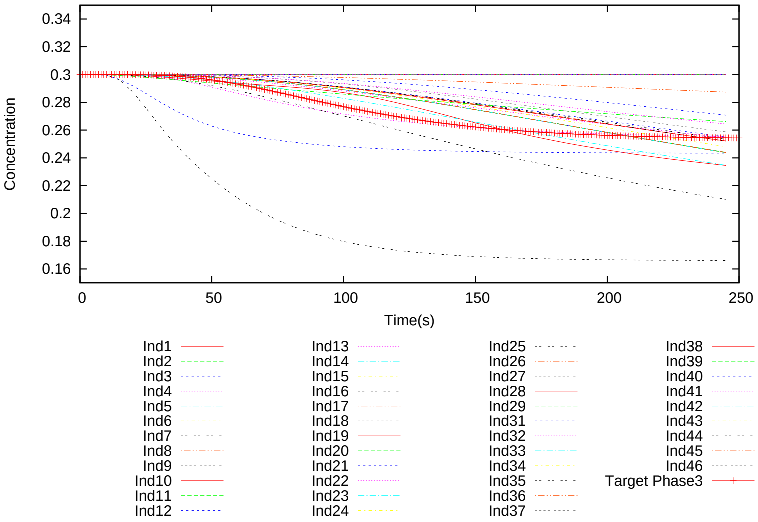


Figure 6.21: Comparison of *Phase3* behaviour.

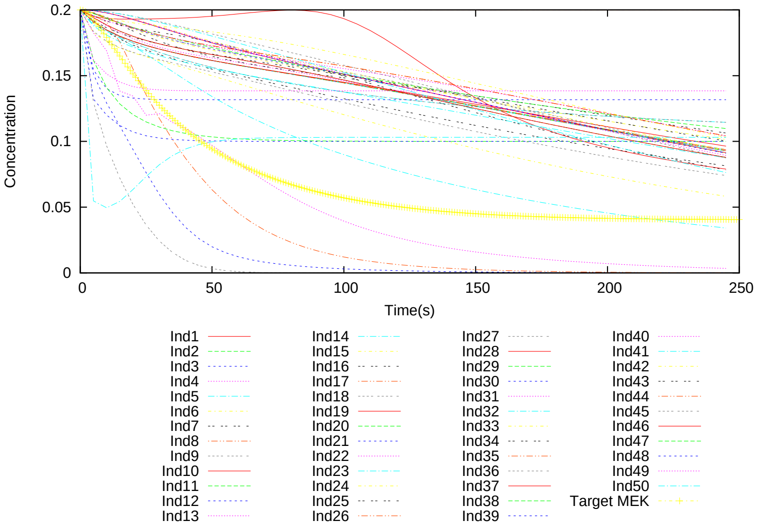


Figure 6.22: Comparison of species *MEK* behaviour.

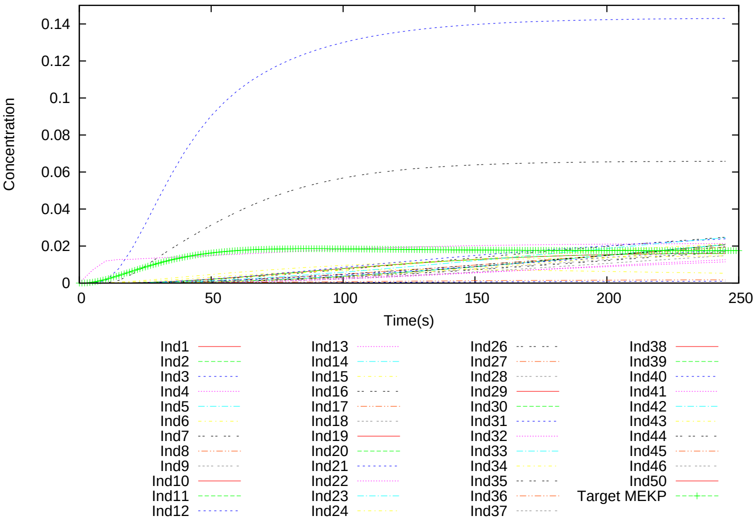


Figure 6.23: Comparison of species *MEKP* behaviour.

6.3.2 Convergence of composed model fitness

While modelling the Levchenko2000, the same parameters of running hybrid modelling are utilized for applying ES layer to evolve model seeds at every generation and calling SA layer at every 250 generations to optimize kinetic rates associated with reactions in these model seeds. Figure 6.24 shows that an average fitness value for 50 synthetic models converge to a minimum value with the increased number of generations in the simulation.

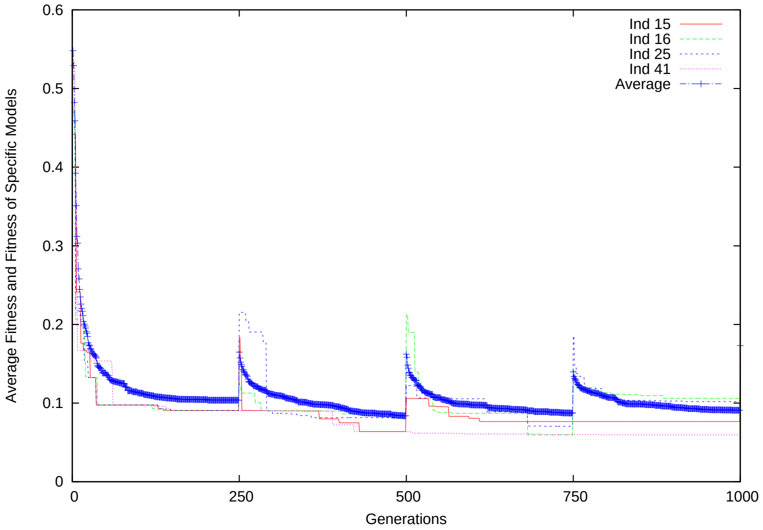


Figure 6.24: Average and four best fitness values of synthetic Levchenko2000 models.

Regarding the probability of models selection during the SA based optimization process, worse models with bad performance in terms of behaviour distance can be accepted. Therefore, there is jump in the fitness values of the models under construction in the figure. We also analyze the fitness values of the four best models obtained from the set of returned models. The fitness values of the four best models converge and jump with the increased generation numbers, as the converged average fitness shown in the same figure. Therefore, it is feasible to apply our proposed 2D hybrid piecewise modelling approach to obtain models with converged fitness (indicating the models to be close to target biochemical system) and similar behaviour (suggesting correct generation of biochemical interactions in the synthetic models).

6.3.3 Quantitative analysis of composed topologies

6.3.3.1 Compression

Constructed models of Levchenko2000 are analyzed on topologies by employing one of the quantitative measurements, *Compression*. Figure 6.25 shows that 50 synthetic models are compared with the target Levchenko2000 and corresponding compression scores are computed.

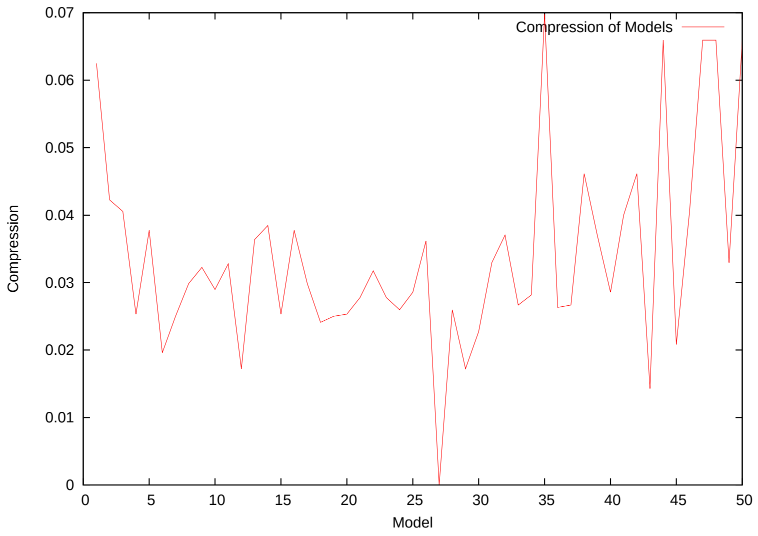


Figure 6.25: Compression analysis of the synthetic topologies.

As shown in Figure 6.25, compression scores of the synthetic models are very poor, which are distributed in a range of [0, 0.07]. There is one composed model not including matched arcs on the structure. The poor compression scores indicate that synthetic models are very different to the target model. In order to investigate the characteristics of various structures among these constructed models, we also utilized another quantitative measurement, *Coverage*, to analyze the composed models. The details of analysis of coverage scores are illustrated in next section.

6.3.3.2 Coverage

Coverage scores of constructed models are calculated and shown in Figure 6.26.

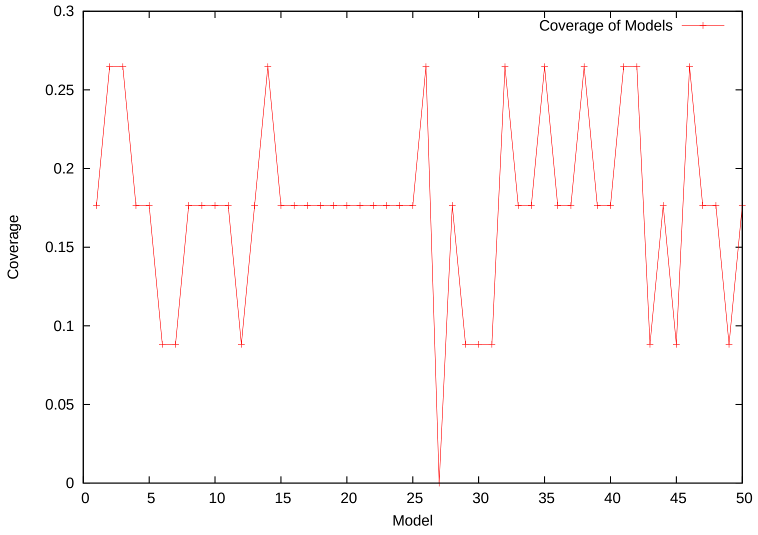


Figure 6.26: Coverage analysis of the synthetic topologies.

A set of very low coverage scores of synthetic models for target Levchenko2000 is obtained, ranging over [0, 0.27]. The same as the illustrated compression scores of these synthetic models, a constructed model having zero coverage score suggests that no matched arcs exist in the model. Therefore, regarding both low compression and coverage scores of these obtained synthetic models, we can conclude that constructed models from the hybrid piecewise modelling are very different to the given target biochemical pathway in terms of topology, but most of the species or complex among these synthetic models exhibit similar behaviour to the target ones in Levchenko2000.

6.3.4 Generation of alternative Topologies

We compare one of composed model with target Levchenko2000 model to present generation of alternative topologies from our approach. There are 30 reactions in Levchenko2000 and 31 reactions are generated in a synthetic model.

Table 6.2: Comparison of one synthetic model with Levchenko2000.

Reactions in Levchenko2000 model	Reactions in One Generated Model
*$Raf + RasGTP \xrightarrow{k1} Raf\vert RasGTP$	*$MEKP\vert RafP \xrightarrow{r1} MEKPP + RafP$
$Raf\vert RasGTP \xrightarrow{k2} RasGTP + Raf$	*$MEK\vert RafP \xrightarrow{r2} MEK + RafP$
$Raf\vert RasGTP \xrightarrow{k3} RafP + RasGTP$	$MEK + RasGTP \xrightarrow{r3} MEK\vert RasGTP$
$Phase1 + RafP \xrightarrow{k4} Phase1\vert RafP$	$ERKPPP\vert Raf \xrightarrow{r4} ERKPPP + Raf$
$Phase1\vert RafP \xrightarrow{k5} Phase1 + RafP$	$MEKPP\vert Raf \xrightarrow{r5} MEKPP + Raf$
$Phase1\vert RafP \xrightarrow{k6} Raf + Phase1$	$ERK\vert RasGTP \xrightarrow{r6} MEK\vert Phase1 + RasGTP$
$RafP + MEK \xrightarrow{k7} MEK\vert RafP$	$ERK\vert Raf + Phase3 \xrightarrow{r7} ERK\vert Phase3\vert Raf$
*$MEK\vert RafP \xrightarrow{k8} RafP + MEK$	$MEKP\vert RasGTP \xrightarrow{r8} MEKPP + RasGTP$
$MEK\vert RafP \xrightarrow{k9} MEKP + RafP$	*$ERK\vert MEKPP \xrightarrow{r9} ERK + MEKPP$
$RafP + MEKP \xrightarrow{k10} MEKP\vert RafP$	$MEK\vert MEKPP\vert Phase1 + ERK\vert MEKP\vert Phase2 \xrightarrow{r10} ERK\vert MEK\vert MEKP\vert MEKPP\vert Phase1\vert Phase2$
$MEKP\vert RafP \xrightarrow{k11} RafP + MEKP$	$ERK\vert Phase3 + ERK\vert MEKPP \xrightarrow{r11} ERK\vert MEKPP\vert Phase3$
*$MEKP\vert RafP \xrightarrow{k12} MEKPP + RafP$	$Raf\vert RafP \xrightarrow{r12} RafP + Raf$
$Phase2 + MEKPP \xrightarrow{k13} MEKPP\vert Phase2$	*$Raf + RasGTP \xrightarrow{r13} Raf\vert RasGTP$
$MEKPP\vert Phase2 \xrightarrow{k14} Phase2 + MEKPP$	$Raf\vert RasGTP \xrightarrow{r14} Raf\vert RafP + RasGTP$
$MEKPP\vert Phase2 \xrightarrow{k15} Phase2 + MEKP$	$ERK + ERK\vert RasGTP \xrightarrow{r15} ERK\vert RasGTP$
*$Phase2 + MEKP \xrightarrow{k16} MEKP\vert Phase2$	$ERK + MEKP\vert Phase2 \xrightarrow{r16} ERK\vert MEKP\vert Phase2$
$MEKP\vert Phase2 \xrightarrow{k17} Phase2 + MEKP$	$MEK\vert RasGTP \xrightarrow{r17} MEKP + RasGTP$
$MEKP\vert Phase2 \xrightarrow{k18} MEK + Phase2$	*$MEKP + Phase2 \xrightarrow{r18} MEKP\vert Phase2$
$MEKPP + ERK \xrightarrow{k19} ERK\vert MEKPP$	$MEKP\vert RasGTP + MEK\vert RasGTP \xrightarrow{r19} MEK\vert MEKP\vert RasGTP$
*$ERK\vert MEKPP \xrightarrow{k20} MEKPP + ERK$	$ERK\vert Phase3\vert Raf + Raf \xrightarrow{r20} ERK\vert Phase3\vert Raf$
$ERK\vert MEKPP \xrightarrow{k21} ERKP + MEKPP$	$ERK\vert Phase3\vert Raf + MEK\vert RasGTP \xrightarrow{r21} ERK\vert MEK\vert Phase3\vert Raf\vert RasGTP$
$MEKPP + ERKP \xrightarrow{k22} ERKP\vert MEKPP$	$ERKPP\vert Phase1 + ERK\vert MEK\vert Phase3\vert Raf\vert RasGTP \xrightarrow{r22} ERK\vert ERKPP\vert MEK\vert Phase1\vert Phase3\vert Raf\vert RasGTP$
$ERKP\vert MEKPP \xrightarrow{k23} MEKPP + ERKP$	$MEK + MEKP\vert RasGTP \xrightarrow{r23} MEK\vert MEKP\vert RasGTP$
$ERKP\vert MEKPP \xrightarrow{k24} ERKPP + MEKPP$	$MEK\vert MEKPP \xrightarrow{r24} MEK + MEKPP$
$Phase3 + ERKPP \xrightarrow{k25} ERKPP\vert Phase3$	$Raf + ERKPPP \xrightarrow{r25} ERKPPP\vert Raf$
$ERKPP\vert Phase3 \xrightarrow{k26} Phase3 + ERKPP$	$MEKP + MEKPP \xrightarrow{r26} MEKP\vert MEKPP$
$ERKPP\vert Phase3 \xrightarrow{k27} Phase3 + ERKP$	$ERK + ERK\vert Phase3 \xrightarrow{r27} ERK\vert Phase3$
$Phase3 + ERKP \xrightarrow{k28} ERKP\vert Phase3$	$MEK\vert RasGTP \xrightarrow{r28} MEK + RasGTP$
$ERKP\vert Phase3 \xrightarrow{k29} Phase3 + ERKP$	$MEK + ERK\vert MEKP\vert Phase2\vert Raf \xrightarrow{r29} ERK\vert MEK\vert MEKP\vert Phase2\vert Raf$
$ERKP\vert Phase3 \xrightarrow{k30} ERK + Phase3$	$MEK\vert RafP + ERK\vert MEKPP \xrightarrow{r30} ERK\vert MEK\vert MEKPP\vert RafP$
	$MEKP\vert Phase2 + Raf \xrightarrow{r31} MEKP\vert Phase2\vert Raf$

As shown in Table 6.2, five reactions in the synthetic model are identical to the ones in original Levchenko2000. The identical reactions are marked with star in the table, indicating that an alternative topology of Levchenko2000 can be obtained with similar behaviour from our hybrid modelling approach.

6.4 Simulations and statistical analysis on modelling variants

In order to quantitatively study various modelling variants, we utilized statistical methodology to analyze the performance of modelling variants by comparing fitness values, compression and coverage scores which are acquired from a set of synthetic models representing target *RKIP* pathway. Firstly, we describe simulation settings for generating synthetic models to compare the modelling variants, then details of statistical analysis and summaries of variants comparison are given.

6.4.1 Simulation settings

There are five pairs of modelling variants compared and investigated by employing the 2D hybrid piecewise modelling approach. For analyzing simulation results and summarizing conclusions about performance of modelling variants from a large group of composed models, each pair of compared modelling variants is utilized to compose models in 10 runs. Details of simulation settings are given in Table 6.3 as follows.

As shown in Table 6.3, there are 10 runs for implementation of each modelling variant on the hybrid modelling platform, ♯Runs=10. The hybrid modelling platform calls the subtraction operator at every two generations, Sub@Ge=2; SA is called to optimize kinetic rates in each model individual at every 25 generations, OptRate@Ge=25; reward ε_1 and penalty ε_2 of models construction are 0.01 and 1000 respectively, ε_1=0.01 and ε_2=1000. ES and SA are employed to compose models of biochemical systems, therefore the standard settings of ES and SA are utilized. The number of generations in one run of ES is 100,

Table 6.3: Simulation settings for running modelling variants.

Modelling Variants	Hybrid Modelling	ES	SA	Gaussian $N(\mu, \sigma)$

Data Driven:
Fixed vs Dynamic
	♯Runs = 10
	Sub@Ge= 2
	OptRate@Ge = 25
	ε_1=0.01
	ε_2=1000
		GeSi = 100
		PopSi = 50
			T_{ini} = 10
			CoRate = 0.8
			T_{min} = 1
			Iter = 10
				μ= 0
				σ= 0.00001

Survival Selection: SES vs PES				
Mutation: Fixed vs Random				
Recombination: Best vs Random				
Fitness Function: ED vs (ED+RP)				

GeSi=100; the number of population (models seeds) in one generation is 50, PopSi=50. Initial SA system temperature is 10, T_{ini}=10; cooling rate of SA system is 0.8, CoRate=0.8; minimum temperature for stopping simulation is 1, T_{min}=1; and iterations at each simulated annealing temperature are 10, Iter=10. The mean μ and standard deviation σ of Gaussian distribution $N(\mu, \sigma)$ are 0 and 0.00001, μ=0 and σ=0.00001. Other properties of the simulation setting during the modelling process are fixed without modification except the two compared modelling variants, which allows a fair comparison between two modelling variants in each pair in terms of performance on generation of synthetic models. Since there are 50 models seeds initiated at each run for models development and 10 runs simulation for examination of each modelling variant, there are 2×500 composed models obtained for comparison and analysis of each pair of modelling variants.

6.4.2 Statistical analysis

Since we compare pairs of modelling variants with specific aims on modelling biochemical systems, it is necessary to statistically investigate the variances of modelling variants in each pair for the generations and evaluations of synthetic models. Two-sample based statistical methods for analysis of two groups of simulation results, for instance compression scores, coverage scores and fitness values of composed models, need to be performed for understanding if the variances of the variants are the same on some specific modelling aims. Two statistical measures in *R* package [R De 09], ‘var.test(X, Y)’ and ‘t.test(X, Y)’, are employed to perform the statistical analysis.

Fitness values, compression and coverage scores of synthetic models are used to calculate p-value in ‘var.test(X, Y)’ and ‘t.test(X, Y)’ for further statistical analysis. Obtained p-value in two statistical measures are compared with a traditional significant level ‘p=0.05’, and the ratios of variances among generated models from different implementation of modelling variants are also compared. Conclusions are summarized from results of statistical analysis, which is shown with comparison of fitness, compression and coverage among these synthetic models. Appendix A gives a short explanation of two samples tests in *R* package for ‘var.test(X, Y)’ and ‘t.test(X, Y)’.

Table 6.4: Statistical analysis of average fitness sets

NO.	X vs Y	var.test(X, Y)		t.test(X, Y)		
		p-value	$r_{Variances}$	p-value	$\bar{X}$	$\bar{Y}$
1.1	Dri_{Fixed} vs Dri_{Dyn}	0.0229	0.6309	$<$ 2.2e-16	4.2474	3.1602
1.2	SES vs PES	0.4574	1.1616	0.837		4.2289
1.3	M_{Fixed} vs M_{Ran}	0.6821	0.9208	0.0262		4.035
1.4	$\otimes_{Ran}$ vs $\otimes_{Best}$	1.07e-03	1.9448	0.5737		4.2019
1.5	ED vs (ED+RP)	$<$ 2.2e-16	6.15e-06	$<$ 2.2e-16		348.78

After obtaining sets of generated models from simulation runs based on different modelling variants of 2D hybrid piecewise modelling approach, average of fitness of these models among all runs can be calculated for statistical analysis. Table 6.4 shows statistical analysis results on the synthetic models from simulations based on each pair of compared modelling variants: p-value and ratio of variances from var.test() measure; and p-value and means of fitness of models constructed by employing modelling variants.

The compression and coverage scores of these composed models can be measured for further statistical analysis. Table 6.5 and Table 6.6 show that compression and coverage scores of synthetic models from different simulation runs based on different modelling variants are analyzed to obtain p-value, ratio of variances, and means of these scores in var.test() and t.test() statistical measurements. By comparing the statistical analysis results in each pair of modelling variants, advantage and disadvantage of the variants for modelling biochemical systems can be illustrated quantitatively.

Table 6.5: Statistical analysis of average compression.

NO.	X vs Y	var.test(X, Y)		t.test(X, Y)		
		p-value	$r_{Variances}$	p-value	$\bar{X}$	$\bar{Y}$
1.1	Dri_{Fixed} vs Dri_{Dyn}	0.0096	0.4713	$<$ 2.2e-16	0.0526	0.025
1.2	SES vs PES	0.0461	1.7802	6.78e-16		0.0361
1.3	M_{Fixed} vs M_{Ran}	0.75	1.0958	0.0296		0.0567
1.4	$\otimes_{Ran}$ vs $\otimes_{Best}$	1.60e-06	0.2387	$<$ 2.2e-16		0.1033
1.5	ED vs (ED+RP)	1.25e-05	3.6546	0.0004		0.0469

Table 6.6: Statistical analysis of average coverage.

NO.	X vs Y	var.test(X, Y)		t.test(X, Y)		
		p-value	$r_{Variances}$	p-value	$\bar{X}$	$\bar{Y}$
1.1	Dri_{Fixed} vs Dri_{Dyn}	6.74e-12	8.4369	$<$ 2.2e-16	0.2322	0.0731
1.2	SES vs PES	0.4961	1.2161	0.0261		0.2065
1.3	M_{Fixed} vs M_{Ran}	0.062	1.7147	6.63e-05		0.2765
1.4	$\otimes_{Ran}$ vs $\otimes_{Best}$	0.3373	1.3178	0.1888		0.2174
1.5	ED vs (ED+RP)	9.39e-05	0.3163	1.05e-14		0.3967

Details of advantage and disadvantage of applying different modelling variants to construct models are described in next section with the quantitative comparison of modelling variants. Moreover, since the synthetic models in a generation are independent during the construction process, the corresponding compression and coverage scores of the models can be analyzed in a cumulative ascending order, as a complementary analysis of the statistical analysis results.

6.4.3 Comparison of modelling variants

6.4.3.1 Fixed vs Dynamic - Data driven

Here is a brief summary of comparing data driven modelling variants which are in fixed or dynamic manner:

- For generating desired behaviour: dynamic variant is better than fixed one;
- For generating similar topologies: fixed variant is better than dynamic one;

- For generating alternative topologies: dynamic variant is better than fixed one.

Figure 6.27 shows that the dynamic version converges more quickly in terms of fitness function than the fixed one. In Table 6.4 (1.1) The two p-value of var.test() and t.test() are both smaller than the significance level 0.05 which means that the variances of fixed variant is smaller than the dynamic one and the mean fitness of the fixed one is greater than that of the dynamic one.

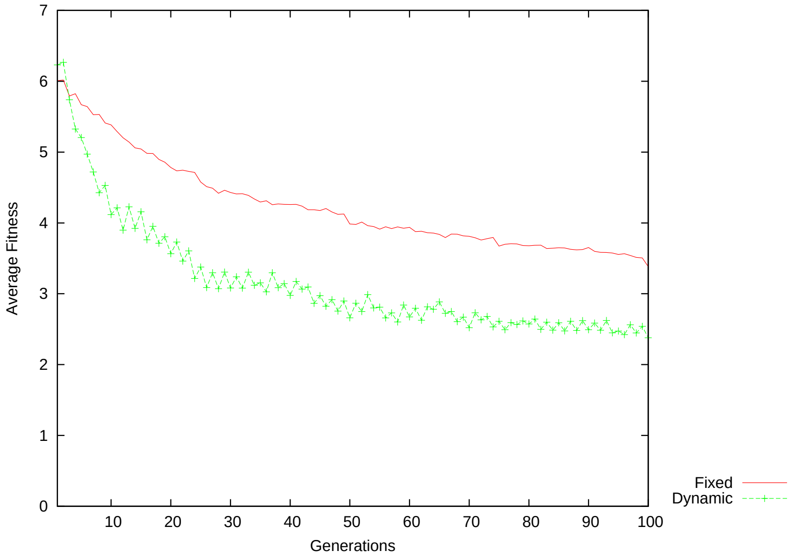
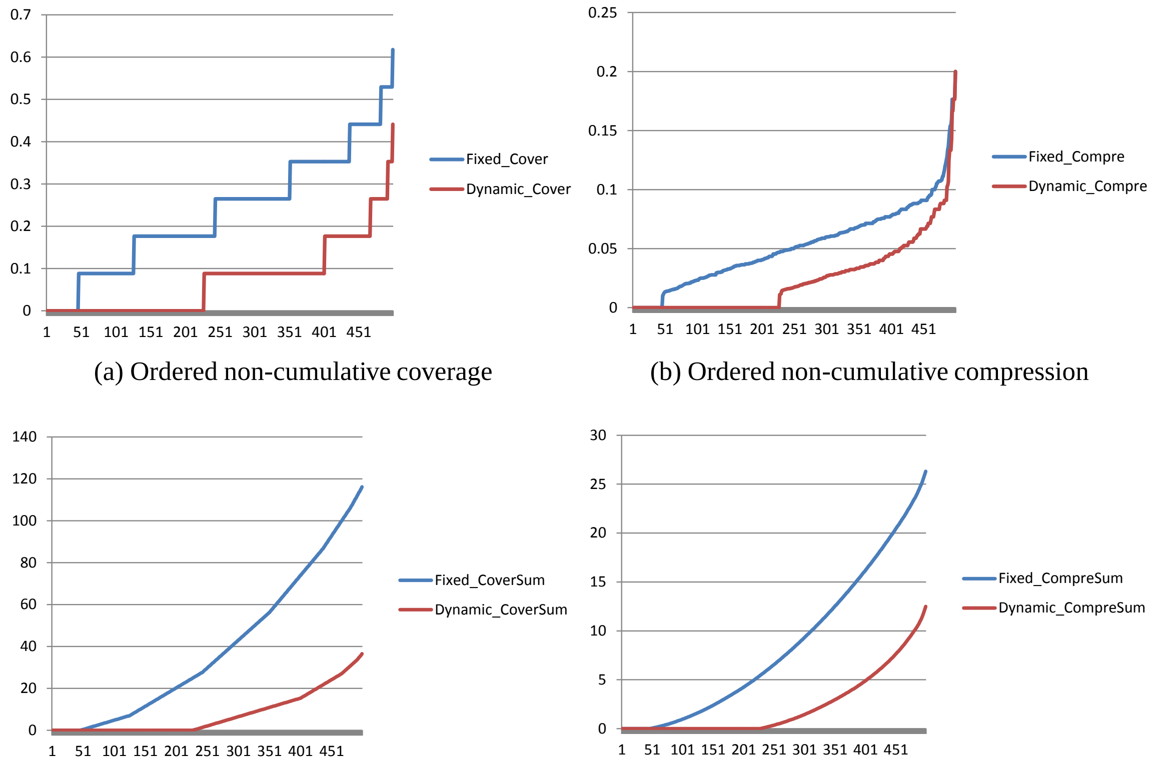


Figure 6.27: Data Driven: Fixed vs Dynamic, Comparison of average fitness of models.

Regarding the exploration of alternative topologies, the compression values of the models generated by dynamic variant is significantly different from the one generated by the fixed variant, see Table 6.5 (1.1) where both p-value are smaller than 0.05. The variance of the dynamic variant is greater than the variance of fixed variant, indicating there is a significant variance in the topologies generated.

In terms of similarity to the target topology, the coverage value of generated models by the fixed variant is greater than that of the dynamic one, as shown in Figure 6.28a and



(a) Ordered non-cumulative coverage

(b) Ordered non-cumulative compression

(c) Ordered cumulative coverage

(d) Ordered cumulative compression

Figure 6.28: Data Driven: Fixed vs Dynamic. (a)-(b) ordered and non-cumulative coverage and compression; (c)-(d) ordered and cumulative coverage and compression. Horizontal axes in the subfigures are cumulative number of generated models. Vertical axes in the subfigures are cumulative/non-cumulative scores of coverage or compression.

Figure 6.28c. As evident from Table 6.6 (1.1), the p-values are smaller than 0.05 which indicates a significant difference between the two variants. Moreover, the variances and means of the fixed variant are greater than the corresponding values of the dynamic one, indicating a higher coverage of structure by the fixed variant. The compression values shown in Figure 6.28b and Figure 6.28d also support this conclusion.

6.4.3.2 SES vs PES - Survival selection

A summary of comparison of implementing survival selection based on SES and PES variants is given as following:

- For generating desired behaviour: the experiments do not show a difference between the implementation of SES and PES;
- For generating similar topologies: SES is better than PES;
- For generating alternative topologies: SES is better than PES.

Figure 6.29 shows that SES and PES have a similar performance regarding the convergence of fitness values. As evident from Tables 6.4 (1.2) and 6.6 (1.2), the p-values are larger than the significant level 0.05 which means the variances and mean fitness values are the same for the two variants.

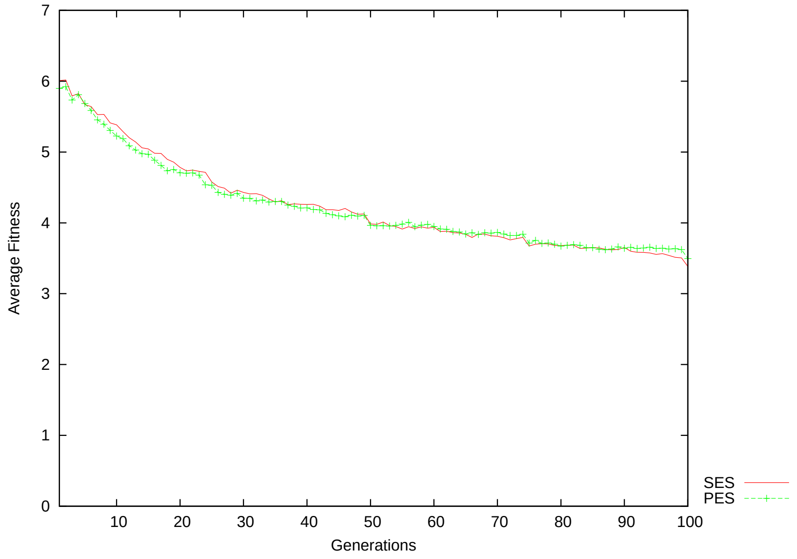
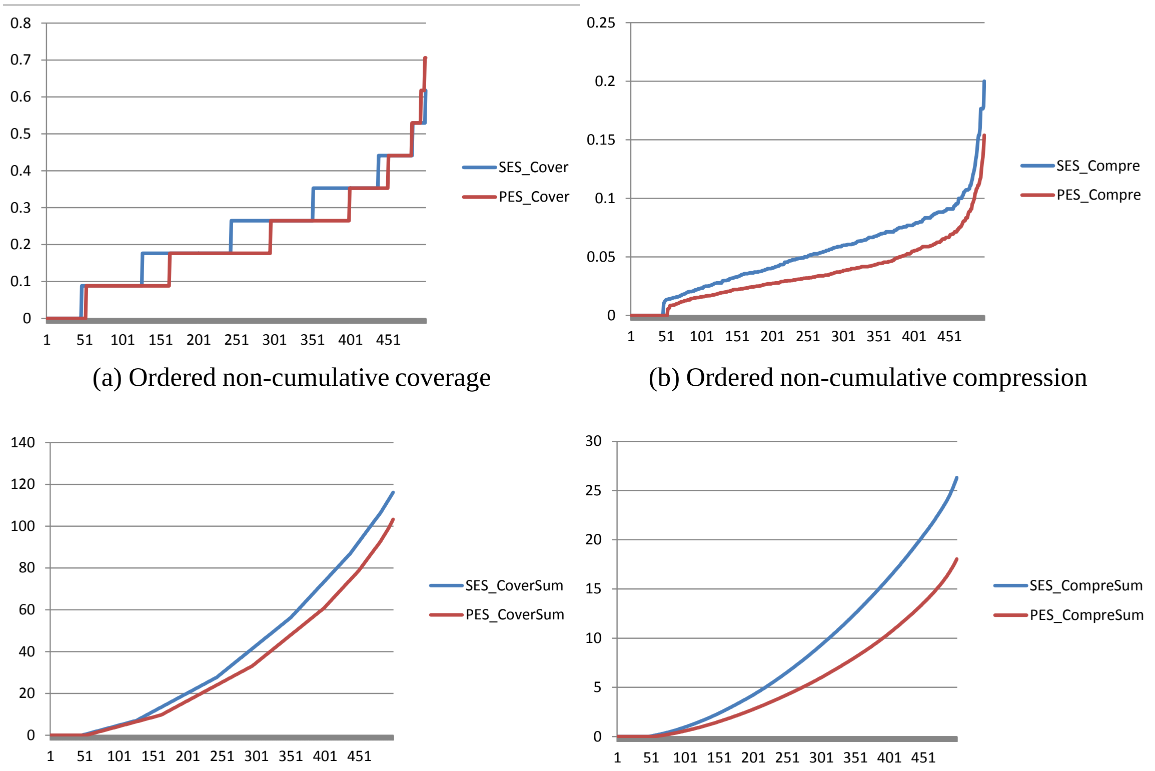


Figure 6.29: Survival Selection: SES vs PES, Comparison of average fitness of models

For exploring alternative topologies, the compression values of the models generated by SES are slightly different from the ones generated by PES (Table 6.5 (1.2), p-value of var.test() is 0.04608 around the significant level 0.05). The ratio of variances between SES and PES is larger than 1, which suggests that SES is better than PES for exploring alternative topologies to the target biochemical system.



(a) Ordered non-cumulative coverage

(b) Ordered non-cumulative compression

(c) Ordered cumulative coverage

(d) Ordered cumulative compression

Figure 6.30: Survival Selection: SES vs PES. (a)-(b) ordered and non-cumulative coverage and compression; (c)-(d) ordered and cumulative coverage and compression. Horizontal axes in the subfigures are cumulative number of generated models. Vertical axes in the subfigures are cumulative/non-cumulative scores of coverage or compression.

Figure 6.30a and Figure 6.30c show that a larger range of coverage values can be generated by SES. Furthermore, in Table 6.6 (1.2), the p-value of t.test() is smaller than 0.05, which means the coverage of models by SES is larger than the one provided by PES. The compression values shown in Figures 6.30b and 6.30d also support this finding.

6.4.3.3 Fixed vs Random - Mutation operator

There are conclusions from the comparison of implementing mutation operator based on fixed and random modelling variants:

- For generating desired behaviour and similar topologies: random variant is better than fixed one;
- alternative topologies: random variant is the same as fixed one.

Figure 6.31 shows the convergence of the fitness values of models generated by fixed and random variants. In Table 6.4 (1.3) and Table 6.6 (1.3), the two p-value of t.test() are both smaller than the significance level 0.05, indicating the mean fitness of fixed variant is significantly different from the random one. It suggests that the mean fitness value of random variant is smaller (more close to the desired behaviour) than the fixed one; and the mean of coverage of random variant is larger (more coverage of the target structure) than the fixed one.

For exploring alternative topologies, the random variant is the same as the fixed one, supported by Figure 6.32a and Figure 6.32c for similar coverage scores, and Figure 6.32b and Figure 6.32d for similar compression scores. In Table 6.5 (1.3), the variances of fixed and random variants are not different (p-value of var.test() is great larger than 0.05), which indicates that the fixed and random variants have the same ability of exploring alternative structures.

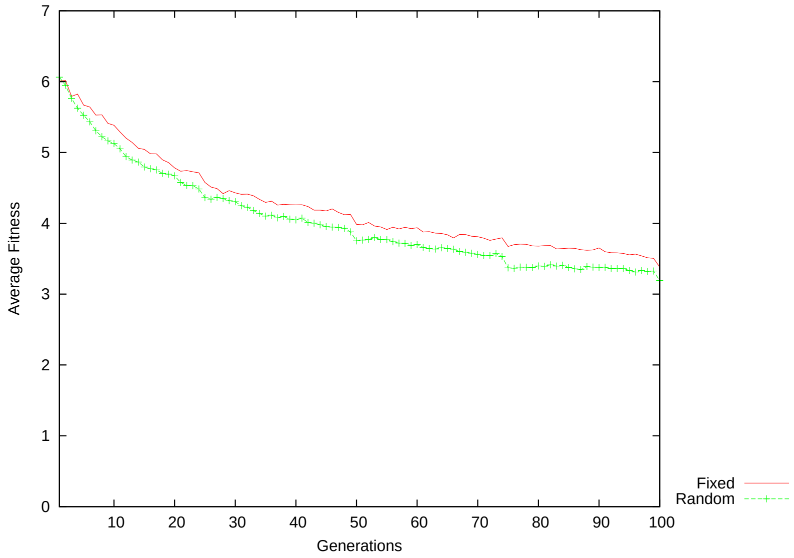


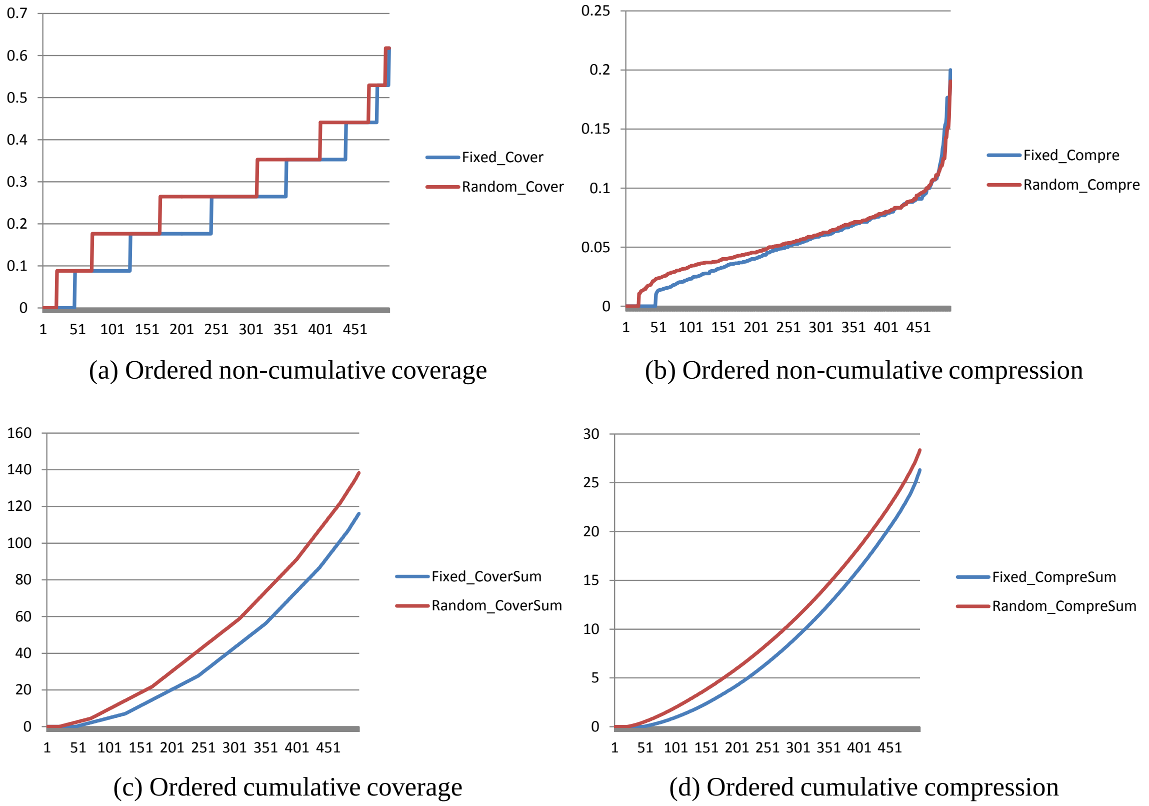
Figure 6.31: Mutation: Fixed vs Random, Comparison of average fitness of models.

6.4.3.4 Best vs Random - Crossover operator

Following conclusions are from the comparison of implementation of crossover operator based on best and random modelling variants:

- For generating desired behaviour and similar topologies: a random selection of mate for recombination works the same as the selection of the best individual;
- For generating alternative topologies: selection of best individual for recombination is better than the random selection.

Figure 6.33 shows the convergence of the fitness values. In Table 6.4 (1.4) and Table 6.6 (1.4), the two p-values of t.test() are both larger than the significant level 0.05, indicating that the mean fitness and coverage values of the random variant are the same as the ones of the best variant. It suggests that the best and random mechanisms of selecting individual for crossover have the same performance in terms of approaching desired behaviour and generating similar topology.



(a) Ordered non-cumulative coverage

(b) Ordered non-cumulative compression

(c) Ordered cumulative coverage

(d) Ordered cumulative compression

Figure 6.32: Mutation: Fixed vs Random. (a)-(b) ordered and non-cumulative coverage and compression; (c)-(d) ordered and cumulative coverage and compression. Horizontal axes in the subfigures are cumulative number of generated models. Vertical axes in the subfigures are cumulative/non-cumulative scores of coverage or compression.

In Table 6.5 (1.4), the variances of random and best strategies are significantly different (p-value of var.test() is smaller than 0.05), and the ratio of variances is smaller than 1, supporting the conclusion that the best variant is better than the random one exploring various structures of the target biochemical pathway. This conclusion is also supported by comparing the coverage values in Figure 6.34a and Figure 6.34c, and the compression values in Figure 6.34b and Figure 6.34d.

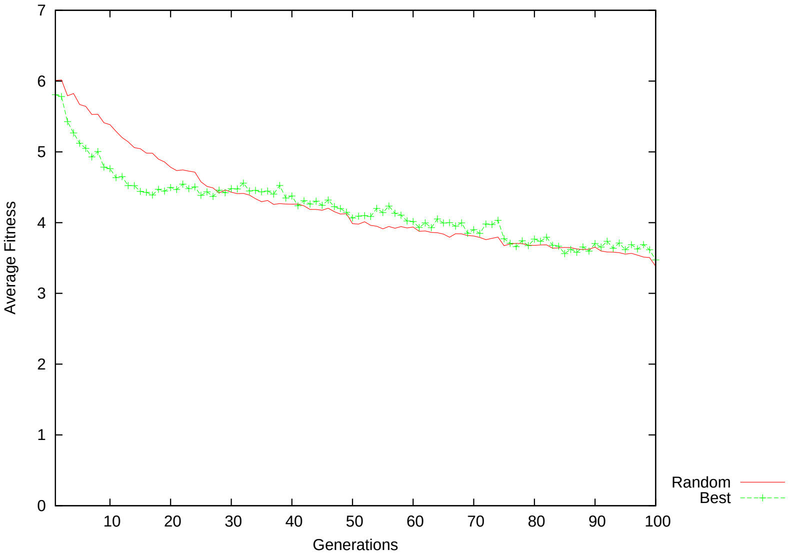


Figure 6.33: Recombination: Best vs Random, Comparison of average fitness of models.

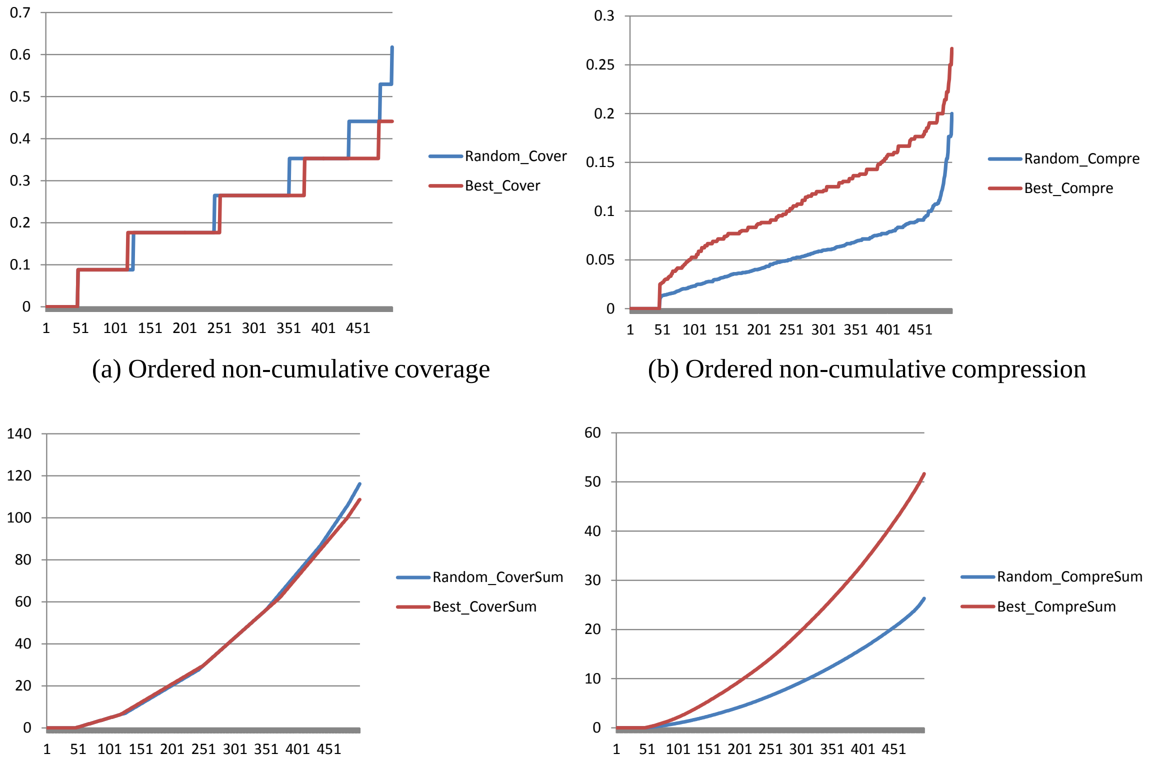
6.4.3.5 ED vs ED+RP - Objective function

A summary of comparison of implementing ED and ED+RP based distance estimation in objective function is given as follows:

- For generating similar topologies: ED+RP variant is better than ED one;
- For generating alternative topologies: ED variant is better than ED+RP one.

In Table 6.4 (1.5), the p-value is much smaller than 0.05, indicating a significant difference between ED and ED+RP variants. Figure 6.35 describes the average fitness values from the objective functions involving a measurement of pure ED, or a mechanism of reward and penalty in the distance estimation function.

As shown in Figure 6.36a and Figure 6.36c, the average coverage values are significantly different between ED and ED+RP, illustrated in Table 6.6 (1.5). Moreover, the average coverage value is larger for the models estimated by ED+RP which suggests that



(a) Ordered non-cumulative coverage

(b) Ordered non-cumulative compression

(c) Ordered cumulative coverage

(d) Ordered cumulative compression

Figure 6.34: Recombination: Best vs Random. (a)-(b) ordered and non-cumulative coverage and compression; (c)-(d) ordered and cumulative coverage and compression. Horizontal axes in the subfigures are cumulative number of generated models. Vertical axes in the subfigures are cumulative/non-cumulative scores of coverage or compression.

the ED+RP variant can be better than the ED variant in terms of generating similar topologies. But the p-value of var.test() in Table 6.5 (1.5) is smaller than 0.05 and the ratio of variances is larger than 1.

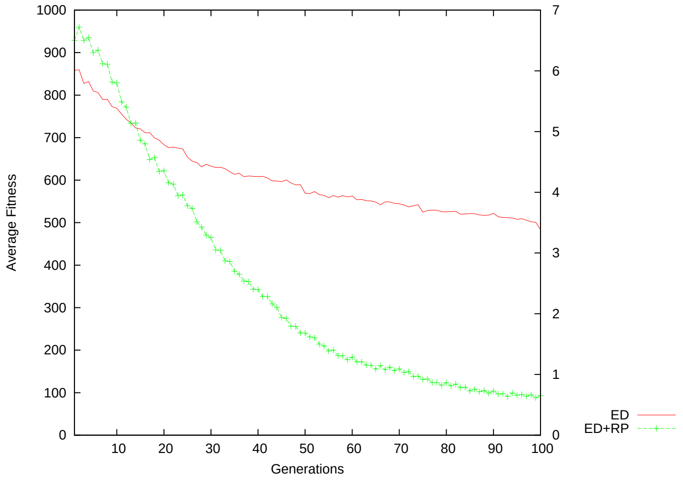
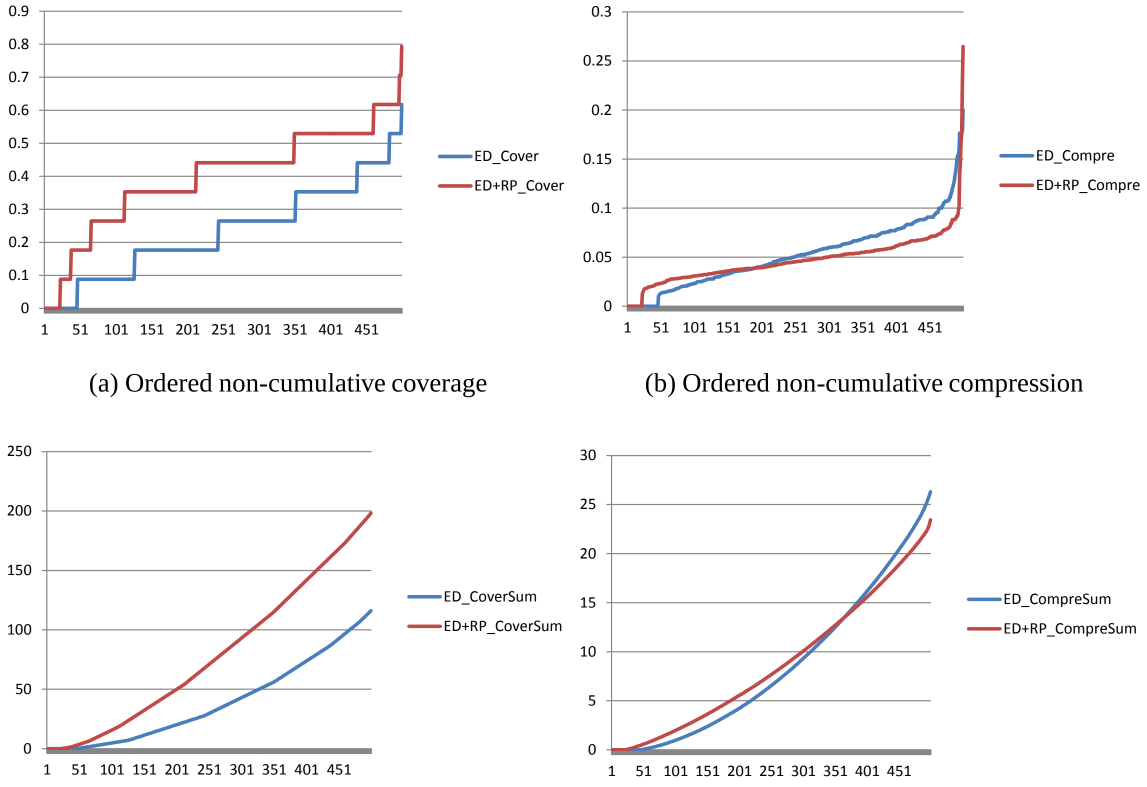


Figure 6.35: Objective Function: ED vs ED+RP, Comparison of average models fitness.



(a) Ordered non-cumulative coverage

(b) Ordered non-cumulative compression

(c) Ordered cumulative coverage

(d) Ordered cumulative compression

Figure 6.36: Fitness Function: ED vs ED+RP. (a)-(b) ordered and non-cumulative coverage and compression; (c)-(d) ordered and cumulative coverage and compression. Horizontal axes in the subfigures are cumulative number of generated models. Vertical axes in the subfigures are cumulative/non-cumulative scores of coverage or compression.

6.4.4 A summary of findings

After performing simulations on modelling variants, results are generated for representing advantage and disadvantage of these variants regarding specific functions of modelling biochemical systems. In addition, statistical analysis of the composed models from different implementation of modelling variants are carried out. A summary of findings about the performance of these variants focusing on specific modelling aspects is obtained. Details of the summary is shown in Table 6.7 which describes whether a modelling variant performs better, worse or the same as another one it is directly compared with.

Table 6.7: A summary of performance between compared modelling variants.

Modelling Variants	Desired Behaviours	Similar Topologies	Alternative Topologies
Data Driven: Fixed vs Dynamic	Dynamic	Fixed	Dynamic
Survival Selection: SES vs PES	=	SES	SES
Mutation: Fixed vs Random	Random	×	=
Recombination: Best vs Random	=	×	Best
Fitness Function: ED vs (ED+RP)	×	ED+RP	ED

Notes: × means not comparable; '=' means the same.

Note that some of the modelling variants are not directly comparable, because the statistical values are not in the same measurement scale. For instance, the modelling variants ED and ED+RP are not comparable in terms of fitness values, since the mechanism of reward and penalty generates a different fitness scale. Details of comparison and summaries among pairs of modelling variants are given in Section 6.4.3. The conclusions about the performance of modelling variants are based on the statistical analysis of average fitness, compression and coverage scores of the composed models in Section 6.4.2.

6.5 Summary

This chapter focuses on the implementation of our 2D hybrid piecewise modelling approach on concrete signalling pathways, comparing performance of different modelling variants. Alternative topologies of synthetic models obtained *in silico* can be taken as general guides for biologists to examine and understand biochemical systems by experimental techniques in wet-lab. Moreover, these composed models with alternative structures can be used as templates for researchers in synthetic biology to develop specific functions of biochemical systems. Summaries about the performance of applying different modelling variants to develop models are useful for further models construction with respect to specific aims of modelling biochemical systems.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

The research in this thesis presents a hybrid piecewise modelling framework based on evolutionary algorithms and graph theory to model biochemical systems in terms of topology and kinetic rates, driven by target species behaviour. We have applied the modelling framework in which both topology and kinetic rates are manipulated. Furthermore, variants of the proposed modelling framework are investigated for understanding which features are important for modelling various aspects of biochemical systems *in silico*.

Regarding dynamic continuous behaviour which is of interest and exists in signalling pathways, we focus on modelling of signalling pathways by our proposed hybrid modelling approach. Metabolic pathways and gene regulatory networks are not in the scope of this research, since there is steady state in metabolic pathways and only stochastic behaviour exists in gene regulatory networks. Investigation of modelling metabolic pathways can be found in recent literatures, for example Lodhi and Gilbert [Lodh] studied parameters estimation by use of bootstrapping for time series data characterized by noise. In gene regulatory networks, the inputs are proteins (for instance transcription factors produced from signal transduction or metabolic activity) which can influence the expression of genes. In addition, enzymatic activity plays no direct role in the gene regulatory networks, but the products of gene regulatory networks can have an influence in the transcription of other proteins, or can act as enzymes in signalling or metabolic pathways[Brei 08]. Therefore, we only apply our hybrid piecewise modelling framework to study dynamic continuous behaviour in the signalling pathways.

We have introduced background of modelling biochemical systems, with brief descriptions of how to present and simulate biochemical systems in 'dry-lab' based on current different formal mathematical tools, especially examining the issues of modelling biochemical systems in terms of topology and kinetic rates, see Chapter 2.

To achieve the aims of hybrid modelling biochemical systems, in Chapter 3 we have defined basic components and synthetic models in formal syntax and semantics to represent given biochemical systems in our study. Mass-action 1 kinetic law has been employed to define atomic components which can be reused during the process of piecewise models composition. In order to preserve defined atomic components and composed models for reuse while modelling biochemical systems, two libraries have been designed and implemented to support the piecewise development of models. Then, we presented three genetic composition operators and a set of composition rules for implementation. Because components and models are described in Petri nets format, composition operators are proposed to evolve Petri nets for manipulation of synthetic models under construction and generation of similar species behaviour to the target ones. In addition, we have discussed issues of fine tuning Petri nets models by the composition operators and rules. Note that Petri net is chosen for graphical representations of biochemical pathways, because of following reasons: firstly, computational ODEs can be directly mapped from the Petri nets for estimations of synthetic models; secondly, graph operations of addition, subtraction and crossover can be easily applied to Petri nets for composition of models. Although there are other possible graph representations for biochemical pathways, such as compound graph, reaction graph and hypergraph, Petri nets are a natural and established notation for describing biochemical reaction networks both share the bipartite property without any ambiguity [Hein 11, Hein 12].

We applied two types of hybrid modelling approaches to construct models of biochemical systems in Chapter 4: a models generator based on the 1D hybrid piecewise modelling which focuses on construction of models by manipulating topology and optimizing kinetic rates separately; a 2D hybrid piecewise modelling approach which composes biochemical models by employing two evolutionary and heuristic algorithms to set up a two-layer hybrid modelling environment. The 2D hybrid piecewise modelling approach addresses the challenges of constructing models of biochemical systems with respect to involving topology and kinetic rates.

Our proposed hybrid modelling framework is developed by introduction of a grid technique to parallelize modelling process, and comparison of variants of the hybrid modelling approach, see Chapter 5. The GridGain technique has been employed to parallelize the topology construction and kinetic rates optimization respectively. By using GridGain based hybrid piecewise modelling approach, models from the same generation in the evolutionary modelling process can be composed and optimized independently. Simulation process can be speeded up, because the parallel execution of models performs addition, subtraction, crossover operators and rates optimization to models under construction. Regarding specific modelling aims, modelling variants have been explored to investigate the advantages or disadvantages of functions in these variants for piecewise modelling, with an emphasis on the effect of mutation operators and evaluation criteria of the overall hybrid methods. Moreover, measurements of composed models have been studied, by including pure Euclidean distance function and a reward and penalty function in an objective function to estimate behaviour difference between generated and target model. We have presented how to evaluate composed models in terms of topology by introducing quantitative and qualitative methods.

We have applied the 2D hybrid piecewise modelling approach with its variants to concrete signalling pathways for constructing models exhibiting similar behaviour with alternative topologies, see Chapter 6. Simulation results show that it is feasible to compose models from scratch and develop models topologies piece by piece, along with optimization of kinetic rates associated with the biochemical reactions in these models. Examination of modelling variants with analyzed simulation results suggest a set of conclusions can be obtained for indicating advantage and disadvantage of modelling variants on specific modelling aspects.

In summary, this thesis presents a hybrid modelling framework based on quantitative Petri nets to piecewise model and optimize biological systems in terms of topology and kinetic rates. Performance of modelling variants in a hybrid two-layer framework is also investigated. Simulation results are statistically analyzed, providing conclusions about implementation of modelling variants in the hybrid modelling environment.

Our simulation results do not clearly show that one modelling variant clearly outperforms the others, but it provides an indication regarding which features are important to be considered for various aspects of the modelling problem. These conclusions about the variants performance in a hybrid modelling environment can be employed to improve further modelling issues. Moreover, our study in this thesis addresses the evolution of quantitative Petri nets and could thus be applied to stochastic and hybrid Petri nets as well as the continuous Petri nets, which can benefit mathematical modelling.

7.2 Future Work

Theoretical and practical study has been investigated in this thesis for modelling of biochemical systems. A list of potential research directions is proposed as follows.

1. To develop patterns for instantiation of atomic components by using MA2, MA3 and MM kinetic laws for piecewise modelling;

   Different kinetic laws guide biochemical reactions in biological systems. It could confuse many experimentalists in wet-lab, if only applying MA1 kinetics to model biochemical systems. Moreover, an active enzymatic reaction is measured by the MM kinetics, and it is difficult to obtain rates for the atomic reactions. Therefore, a sophisticated modelling strategy including different patterns developed for different kinetic laws could enhance the piecewise modelling.

2. To apply more biological constraints to define and implement composition rules;

   Components are instantiated by following a set of given biological principles. Models are constructed by applying composition rules predefined by users according to given biological constraints. Regarding complex working mechanisms among substrates in biochemical systems, it is necessary to involve more precise and concrete biological knowledge which can guide the instantiation of atomic components and generation of biochemical interactions, for approaching more biological relevant synthetic models.

3. To use concrete biological values, including kinetic rates constants and initial concentrations, while fitting parameters of biochemical systems;

   Random choices of kinetic constants and initial concentrations are feasible while modelling biochemical systems in silico, but these random operations are strange for experimentalists in wet-lab. Therefore, it is better to apply concrete kinetic values from literature or biochemical databases to the variables associated with biochemical reactions while modelling.

4. To optimize kinetic rates by employing Multiobjective Optimization;

   Modification of kinetic rates associated with reactions would result in composed models exhibiting different behaviour. Fine tuning one of reaction rates in a model may affect other reactions, therefore multiple objective optimization methodologies can be employed to analyze the effects of rates modification.

5. To account matched structures and topology sizes of composed models in the objective function for the overall estimation of generated models;

   While fitness of composed models converge to optimal values with increased generations, the topology sizes of composed models could grow without control, even thought subtraction and crossover operations are applied to manipulate the models. The objective function can account for a weighted estimation of matched interactions between generated and target model, for approaching synthetic models with ‘optimal’ fitness and ‘minimal’ topologies.

6. To take improvement of synthetic models across generations into account for modelling stop criteria;

It is important to apply criterion to stop the modelling process automatically, for avoiding anomalies of generating meaningless interactions among substrates in the models. For instance, if there is no improvement after pre-defined number of generations, the modelling process can stop and return current optimal results.

7. To study hybrid implementation of SA and ES at different modelling stages, for instance in a rough manner at initial generations and in a precise manner at final generations;

   Modelling in a rough manner means aspects of generated models are not strictly treated, then criteria of estimating synthetic models are not rigorous. Whereas modelling in a precise manner means criteria can be tough for satisfying modelling requirements. Combination and implementation of rough and precise modelling stages allow models to be developed without rejection even though serious problems existing, and later these models can be checked by strict criteria for more meaningful synthetic models.

8. To independently construct submodels with driving information from different experimental stages in wet-lab, and then to compose these submodels into an integrated model representing target biochemical system.

   A biochemical system is difficult to be observed and measured on concentrations in wet-lab, because of the natural complex of biochemical interactions. It is common to only consider specific experimental stages from which information of the biochemical system can be obtained. Parts of a model (submodules) within different experimental time slots can be generated independently, then these submodules can be composed together into an integrated model.

In summary, we take the hybrid piecewise modelling framework to be only a first trial towards automatically modelling of biochemical systems from scratch by employing meta-heuristics and reusing definable atomic components, driven by target species behaviour information. Regarding availability of generating models from scratch with basic building blocks and biochemical knowledge, we argue that it is a great opportunity for computational biology research to construct alternative and comprehensible models which can be useful for biologists discovering hidden biochemical knowledge and heuristically building biochemical systems. We would like to share our opinions of potential research directions and encourage other software engineers and biological modelers to contribute their efforts to this developing interdisciplinary area.

Bibliography

[Albe 05] M.-A. Albert, J. R. Haanstra, V. Hannaert, J. Van Roy, F. R. Opperdoes, B. M. Bakker, and P. A. Michels. “Experimental and in silico analyses of glycolytic flux control in bloodstream form Trypanosoma brucei”. *Journal of Biological Chemistry*, Vol. 280, No. 31, pp. 28306–28315, Aug. 2005.

[Andr 06] E. Andrianantoandro, S. Basu, D. K. Karig, and R. Weiss. “Synthetic biology: new engineering rules for an emerging discipline.”. *Molecular systems biology*, Vol. 2, No. 1, pp. msb4100073–E1–msb4100073–E14, May 2006.

[Anil 87] S. Anily and A. Federgruen. “Simulated Annealing Methods with General Acceptance Probabilities”. *Journal of Applied Probability*, Vol. 24, No. 3, pp. 657–667, 1987.

[Arde 03] I. I. Ardelean and M. Cavaliere. “Modelling biological processes by using a probabilistic P system software”. Vol. 2, No. 2, pp. 173–197, July 2003.

[Back 94] T. Bäck and F. Hoffmeister. “Basic aspects of evolution strategies”. *Statistics and Computing*, Vol. 4, pp. 51–63, 1994. 10.1007/BF00175353.

[Bald 10] P. Baldan, N. Cocco, A. Marin, and M. Simeoni. “Petri nets for modelling metabolic pathways: a survey”. *Natural Computing*, Vol. 9, pp. 955–989, 2010.

[Ball 10] P. Ballarini and M. L. Guerriero. “Query-based verification of qualitative trends and oscillations in biochemical systems”. *Theoretical Computer Science*, Vol. 411, No. 20, pp. 2019–2036, Apr. 2010.

[Benn 05] S. A. Benner and A. M. Sismour. “Synthetic biology”. *Nature Reviews Genetics*, Vol. 6, No. 7, pp. 533–543, July 2005.

[Berg 02] J. M. Berg, J. L. Tymoczko, and L. Stryer. *Biochemistry*. W. H. Freeman and Co., New York, fifth Ed., 2002.

[Bern 74] C. Bernard. *Lectures on the phenomena of life common to animals and plants*. Charles C. Thomas, Springfield, Illinois, 1974.

[Bert 68] L. von Bertalanffy. *General system theory: foundations, development, applications*. George Braziller, New York, 1968.

[Bett 06] K. Bettenbrock, S. Fischer, A. Kremling, K. Jahreis, T. Sauter, and E.-D. D. Gilles. “A quantitative approach to catabolite repression in Escherichia coli.”. *Journal of biological chemistry*, Vol. 281, No. 5, pp. 2578–2584, Feb. 2006.

[Beye 02] H.-G. Beyer and H.-P. Schwefel. “Evolution strategies - A comprehensive introduction”. *Natural Computing*, Vol. 1, No. 1, pp. 3–52, May 2002.

[Beye 06] A. Beyer, C. Workman, J. Hollunder, D. Radke, U. Möller, T. Wilhelm, and T. Ideker. “Integrated assessment and prediction of transcription factor binding”. *PLoS Computational Biology*, Vol. 2, No. 6, p. e70, jun 2006.

[Blak 11] J. Blakes, J. Twycross, F. J. RomeroCampero, and N. Krasnogor. “The Infobiotics workbench: an integrated in silico modelling platform for systems and synthetic biology”. *Bioinformatics*, Vol. 27, No. 23, pp. 3323–3324, Dec. 2011.

[Blat] M. Blätke, M. Heiner, and W. Marwan. “Tutorial - petri nets in systems biology”. Tech. Rep.

[Blow 02] P. Blower, M. Fligner, J. Verducci, and J. Bjoraker. “On combining recursive partitioning and simulated annealing to detect groups of biologically active compounds”. *Journal of Chemical Information and Computer Sciences*, Vol. 42, No. 2, pp. 393–404, 2002.

[Brau 05] D. Braun, S. Basu, and R. Weiss. “Parameter estimation for two synthetic gene networks: a case study”. In: *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP ’05). IEEE International Conference on*, march 2005.

[Braz 98] A. Br¯azma, I. Jonassen, J. Vilo, and E. Ukkonen. “Pattern discovery in biosequences”. In: *ICGI’98 Proceedings*, pp. 257–270, Springer, July 1998.

[Brei 08] R. Breitling, D. Gilbert, M. Heiner, and R. Orton. “A structured approach for the engineering of biochemical network models, illustrated for signalling pathways”. *Briefings in Bioinformatics*, Vol. 9, No. 5, pp. 404–421, September 2008.

[Brei 10] R. Breitling, R. A. Donaldson, D. R. Gilbert, and M. Heiner. “Biomodel engineering – from structure to behavior”. In: *Trans. Computational Systems Biology XII*, pp. 1–12, Springer, 2010.

[Brug 07] F. J. Bruggeman and H. V. Westerhoff. “The nature of systems biology”. *Trends in microbiology*, Vol. 15, No. 1, pp. 45–50, Jan. 2007.

[Cald 04] M. Calder, S. Gilmore, and J. Hillston. “Modelling the influence of RKIP on the ERK signalling pathway using the stochastic process algebra PEPA”. In: *Trans. Computational Systems Biology*, pp. 1–23, Springer, 2004.

[Cann 32] W. B. Cannon. *The wisdom of the body*. Norton, New York, 1932.

[Cann 35] W. B. Cannon. “Stresses and strains of homeostasis”. *American Journal of the Medical Science*, Vol. 189, No. 1, pp. 13–14, January 1935.

[Cao 10] H. Cao, F. Romero-Campero, S. Heeb, M. Cmara, and N. Krasnogor. “Evolving cell models for systems and synthetic biology”. *Systems and Synthetic Biology*, Vol. 4, pp. 55–84, 2010. 10.1007/s11693-009-9050-7.

[Chao 04] C. Chaouiya, E. Remy, P. Ruet, and D. Thieffry. “Qualitative modelling of genetic networks: From logical regulatory graphs to standard Petri nets”. In: *LNCS*, pp. 137–156, Springer-Verlag, 2004.

[Chao 07] C. Chaouiya. “Petri net modelling of biological networks”. *Briefings in Bioinformatics*, Vol. 8, No. 4, pp. 210–219, 2007.

[Chao 08] C. Chaouiya, E. Remy, and D. Thieffry. “Petri net modelling of biological regulatory networks”. *Journal of Discrete Algorithms*, Vol. 6, No. 2, pp. 165–177, June 2008.

[Chen 07] L. Chen, G. Qi-Wei, M. Nakata, H. Matsuno, and S. Miyano. “Modelling and simulation of signal transductions in an apoptosis pathway by using timed Petri nets”. *Journal of Biosciences*, Vol. 32, No. 1, pp. 113–127, Jan. 2007.

[Chen 09] W. W. Chen, B. Schoeberl, P. J. Jasper, M. Niepel, U. B. Nielsen, D. A. Lauffenburger, and P. K. Sorger. “Input-output behavior of ErbB signaling pathways as revealed by a mass action model trained against dynamic data”. *Molecular Systems Biology*, Vol. 5, Jan. 2009.

[Cho 03] K. H. Cho, S. Y. Shin, H. W. Kim, O. Wolkenhauer, B. Mcferran, and W. Kolch. “Mathematical modeling of the influence of RKIP on the ERK signaling pathway”. In: *C. Priami, editor, Computational Methods in Systems Biology (CSMB’03)*, pp. 127–141, Springer, 2003.

[Chou 09] I.-C. C. Chou and E. O. Voit. “Recent developments in parameter estimation and structure identification of biochemical and genomic systems”. *Mathematical biosciences*, Vol. 219, No. 2, pp. 57–83, June 2009.

[Ciri 10] M. Cirit, C.-C. Wang, and J. M. Haugh. “Systematic quantification of negative feedback mechanisms in the extracellular signal-regulated kinase (ERK) signaling network”. *Journal of Biological Chemistry*, Vol. 285, No. 47, pp. 36736–36744, 2010.

[Cool 10] M. T. Cooling, V. Rouilly, G. Misirli, J. Lawson, T. Yu, J. Hallinan, and A. Wipat. “Standard virtual biological parts: a repository of modular modeling components for synthetic biology”. *Bioinformatics*, Vol. 26, No. 7, pp. 925–931, 2010.

[Czei 11] E. Czeizler, V. Rogojin, and I. Petre. “The phosphorylation of the heat shock factor as a modulator for the heat shock response”. In: *Proceedings of the 9th International Conference on Computational Methods in Systems Biology*, pp. 9–23, ACM, New York, NY, USA, 2011.

[DeLi 88] C. DeLisi. “Computers in molecular biology: current applications and emerging trends”. *Science*, Vol. 240, No. 4848, pp. 47–52, 1988.

[Dunl 07] M. Dunlop, E. Franco, and R. Murray. “A multi-model approach to identification of biosynthetic pathways”. In: *Proceeding of the 2007 American Control Conference (ACC)*, pp. 1600–1605, 2007.

[Durz 08] M. Durzinsky, A. Wagler, R. Weismantel, and W. Marwan. “Automatic reconstruction of molecular and genetic networks from discrete time series data”. *Biosystems*, Vol. 93, No. 3, pp. 181 – 190, 2008.

[Elli 02] W. Elliot and D. Elliot. *Biochemistry and molecular biology*. Oxford University Press, 2nd edition Ed., 2002.

[Feng 04] X. J. Feng, S. Hooshangi, D. Chen, G. Li, R. Weiss, and H. Rabitz. “Optimizing genetic circuits by global sensitivity analysis”. *Biophysical Journal*, Vol. 87, No. 4, pp. 2195–2202, 2004.

[Ferr 09] J. E. Ferrell. “Q&A: systems biology”. *Journal of biology*, Vol. 8, No. 1, p. 2+, January 2009.

[Foge 94] D. B. Fogel. “An introduction to simulated evolutionary optimization”. *Neural Networks, IEEE Transactions on*, Vol. 5, No. 1, pp. 3–14, 1994.

[Fome 07] Y. Fomekong-Nanfack, J. A. Kaandorp, and J. Blom. “Efficient parameter estimation for spatio-temporal models of pattern formation: case study of Drosophila melanogaster”. *Bioinformatics*, Vol. 23, No. 24, pp. 3356–3363, Dec. 2007.

[Fran 04] P. Francois and V. Hakim. “Design of genetic networks with specified functions by evolution in silico”. *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 101, No. 2, pp. 580–585, January 2004.

[Funa 03] A. Funahashi, M. Morohashi, H. Kitano, and N. Tanimura. “CellDesigner: a process diagram editor for gene-regulatory and biochemical networks”. *Biosilico*, Vol. 1, No. 5, pp. 159–162, Nov. 2003.

[Funa 08] A. Funahashi, Y. Matsuoka, A. Jouraku, M. Morohashi, N. Kikuchi, and H. Kitano. “CellDesigner 3.5: a versatile modeling tool for biochemical networks”. *Proceedings of the IEEE*, Vol. 96, No. 8, pp. 1254–1265, Aug. 2008.

[Gheo 08] M. Gheorghe, N. Krasnogor, and M. Camara. “P systems applications to systems biology”. *Biosystems*, Vol. 91, pp. 435–437, 2008.

[Gilb 03] D. Gilbert, D. Westhead, and J. Viksna. “Techniques for comparison, pattern matching and pattern discovery: from sequences to protein topology”. In:

P. Frasconi and R. Shamir, Eds., *Artificial Intelligence and Heuristic Methods in Bioinformatics*, pp. 128–147, IOS Press, 2003.

[Gilb 06] D. Gilbert and M. Heiner. "From Petri Nets to differential equations - an integrative approach for biochemical network analysis". In: S. Donatelli and P. S. Thiagarajan, Eds., *27th International Conference, ATPN 2006, Turku, Finland, June 26-30, 2006*, pp. 181–200, Springer, 2006.

[Gilb 09] D. Gilbert, R. Breitling, M. Heiner, and R. Donaldson. *Membrane Computing*. Springer-Verlag, Berlin, Heidelberg, 2009.

[Gome 05] L. Gomes and J. P. Barros. "Structuring and composability issues in Petri nets modeling". *Industrial Informatics, IEEE Transactions on*, Vol. 1, No. 2, pp. 112 – 123, may 2005.

[Gonz 07] O. R. Gonzalez, C. Küper, K. Jung, P. C. Naval, and E. Mendoza. "Parameter estimation using simulated annealing for S-system models of biochemical networks". *Bioinformatics*, Vol. 23, No. 4, pp. 480–486, Feb. 2007.

[Grid] "GridGain 3.0". `http://www.gridgain.com`. Accessed: 30/09/2012.

[Guim 05] R. Guimerà and L. A. N. Amaral. "Functional cartography of complex metabolic networks". *Nature*, Vol. 433, No. 7028, pp. 895–900, Feb. 2005.

[Guim 07] R. Guimerà, M. Sales-Pardo, and L. Amaral. "A network-based method for target selection in metabolic networks". *Bioinformatics*, Vol. 23, No. 13, pp. 1616–1622, July 2007.

[Hard 08] S. Hardy and P. N. Robillard. "Petri net-based method for the analysis of the dynamics of signal propagation in signaling pathways". *Bioinformatics*, Vol. 24, No. 2, pp. 209–217, Jan. 2008.

[Hein 06] M. Heinemann and S. Panke. “Synthetic biology-putting engineering into biology”. *Bioinformatics*, Vol. 22, No. 22, pp. 2790–2799, Nov. 2006.

[Hein 11] M. Heiner and D. Gilbert. “How might petri nets enhance your systems biology toolkit”. In: *Proceedings of the 32nd international conference on Applications and theory of Petri Nets*, pp. 17–37, Springer-Verlag, Berlin, Heidelberg, 2011.

[Hein 12] M. Heiner and D. Gilbert. “BioModel engineering for multiscale Systems Biology”. *Progress in Biophysics and Molecular Biology*, Vol. , No. 0, pp. 1 – 10, 2012.

[Hoop 06] S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer. “COPASI—a complex pathway simulator”. *Bioinformatics*, Vol. 22, No. 24, pp. 3067–3074, Dec. 2006.

[Huck 10] M. Hucka, F. Bergmann, S. Hoops, S. Keating, S. Sahle, J. Schaff, L. Smith, and D. Wilkinson. “The systems biology markup language (SBML): language specification for level 3 version 1 core”. 6 October 2010.

[Ihme 04] J. H. Ihmels and S. Bergmann. “Challenges and prospects in the analysis of large-scale gene expression data”. *Briefings in Bioinformatics*, Vol. 5, No. 4, pp. 313–327, Jan. 2004.

[Ji 06] X. Ji and Y. Xu. “libSRES: a C library for stochastic ranking evolution strategy for parameter estimation”. *Bioinformatics*, Vol. 22, No. 1, pp. 124–126, Jan. 2006.

[Joyn 11] M. J. Joyner and B. K. Pedersen. “Ten questions about systems biology”. *The Journal of Physiology*, Vol. 589, No. 5, pp. 1017–1030, March 2011.

[Kacs 73] H. Kacser and J. A. Burns. “The control of flux”. *Symposia of the Society for Experimental Biology*, Vol. 27, pp. 65–104, 1973.

[Kffn 00] R. Kffner, R. Zimmer, and T. Lengauer. “Pathway analysis in metabolic databases via differential metabolic display (DMD)”. *Bioinformatics*, Vol. 16, No. 9, pp. 825–836, 2000.

[Khal 10] A. S. Khalil and J. J. Collins. “Synthetic biology: applications come of age”. *Nature Reviews Genetics*, Vol. 11, No. 5, pp. 367–379, May 2010.

[Khol 02] B. N. Kholodenko, A. Kiyatkin, F. J. Bruggeman, E. Sontag, H. V. Westerhoff, and J. B. Hoek. “Untangling the wires: A strategy to trace functional interactions in signaling and gene networks”. *Proceedings of the National Academy of Sciences*, Vol. 99, No. 20, pp. 12841–12846, Oct. 2002.

[Khol 99] B. N. Kholodenko, O. V. Demin, G. Moehren, and J. B. Hoek. “Quantification of short term signaling by the epidermal growth factor receptor”. *Journal of Biological Chemistry*, Vol. 274, No. 42, pp. 30169–30181, Oct. 1999.

[Kirk 83] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. “Optimization by simulated annealing”. *Science, Number 4598, 13 May 1983*, Vol. 220, 4598, pp. 671–680, 1983.

[Kita 02a] H. Kitano. “Systems biology: a brief overview.”. *Science*, Vol. 295, No. 5560, pp. 1662–1664, March 2002.

[Kita 02b] H. Kitano. “Computational systems biology”. *Nature*, Vol. 420, No. 6912, pp. 206–210, November 2002.

[Kita 02c] H. Kitano. “Looking beyond the details: a rise in system-oriented approaches in genetics and molecular biology”. *Current Genetics*, 2002.

[Kita 05] H. Kitano, A. Funahashi, Y. Matsuoka, and K. Oda. “Using process diagrams for the graphical representation of biological networks.”. *Nature biotechnology*, Vol. 23, No. 8, pp. 961–966, August 2005.

[Kiya 06] A. Kiyatkin, E. Aksamitiene, N. I. Markevich, N. M. Borisov, J. B. Hoek, and B. N. Kholodenko. "Scaffolding protein Grb2-associated binder 1 sustains epidermal growth factor-induced mitogenic and survival signaling by multiple positive feedback loops.". *Journal of biological chemistry*, Vol. 281, No. 29, pp. 19925–19938, July 2006.

[Klip 05] E. Klipp, R. Herwig, A. Kowald, C. Wierling, and H. Lehrach. *Systems Biology in Practice: Concepts, Implementation and Application*. Wiley-VCH, 2005.

[Koch 05] I. Koch, B. H. Junker, and M. Heiner. "Application of petri net theory for modelling and validation of the sucrose breakdown pathway in the potato tuber". *Bioinformatics*, Vol. 21, No. 7, pp. 1219–1226, Apr. 2005.

[Kohn 83] M. C. Kohn and W. J. Letzkus. "A graph-theoretical analysis of metabolic regulation". *Journal of Theoretical Biology*, Vol. 100, No. 2, pp. 293 – 304, 1983.

[Krem 01] A. Kremling, K. Bettenbrock, B. Laube, K. Jahreis, J. W. Lengeler, and E. D. Gilles. "The organization of metabolic reaction networks: III. application for diauxic growth on glucose and lactose". *Metabolic Engineering*, Vol. 3, No. 4, pp. 362–379, 2001.

[Krem 04] A. Kremling, S. Fischer, K. Gadkar, F. J. Doyle, T. Sauter, E. Bullinger, F. Allgöwer, and E. D. Gilles. "A benchmark for methods in reverse engineering and model discrimination: Problem formulation and solutions". *Genome Research*, Vol. 14, No. 9, pp. 1773–1785, Sep. 2004.

[Lang 89] C. G. Langton. *Artificial Life: proceedings of an interdisciplinary workshop on the synthesis and simulation of living systems*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1989.

[Levc 00] A. Levchenko, J. Bruck, and P. W. Sternberg. “Scaffold proteins may biphasically affect the levels of mitogen-activated protein kinase signaling and reduce its threshold properties”. *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 97, No. 11, pp. 5818–5823, May 2000.

[Li 05] W. Li and H. Kurata. “A grid layout algorithm for automatic drawing of biochemical networks”. *Bioinformatics*, Vol. 21, No. 9, pp. 2036–2042, May 2005.

[Li 10] C. Li, M. Donizelli, N. Rodriguez, H. Dharuri, L. Endler, V. Chelliah, L. Li, E. He, A. Henry, M. I. Stefan, J. L. Snoep, M. Hucka, N. Le Novère, and C. Laibe. “BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models.”. *BMC Systems Biology*, Vol. 4, p. 92, June 2010.

[Liu 08] X. Liu, J. Jiang, O. Ajayi, X. Gu, and D. Gilbert. “BioNessie(G)- A Grid Enabled Biochemical Networks Simulation Environment”. *Studies in Health Technology and Informatics*, Vol. 138, pp. 147–157, 2008.

[Liu 12] F. Liu, M. Heiner, and C. Rohr. “Manual for colored petri nets in Snoopy”. Tech. Rep. 02-12, Brandenburg University of Technology Cottbus, Department of Computer Science, March 2012.

[Lodh] H. Lodhi and D. Gilbert. “Bootstrapping parameter estimation in dynamic systems”. In: T. Elomaa, J. Hollmn, and H. Mannila, Eds., *Discovery Science*, pp. 194–208, Springer Berlin / Heidelberg.

[Loza 10] M. Lozano and C. García-Martínez. “Hybrid metaheuristics with evolutionary algorithms specializing in intensification and diversification: Overview and progress report”. *Computers and operations research*, Vol. 37, No. 3, pp. 481–497, March 2010.

[Mach 11] D. Machado, R. Costa, M. Rocha, E. Ferreira, B. Tidor, and I. Rocha. “Modeling formalisms in systems biology”. *AMB Express*, Vol. 1, No. 1, pp. 1–14, Dec. 2011.

[Manc 11] V. Manca and L. Marchetti. “Log-Gain stoichiometric stepwise regression for MP systems”. *International Journal of Foundations of Computer Science*, pp. 97–106, 2011.

[Mari 04] G. Maria. “A review of algorithms and trends in kinetic model identification for chemical and biochemical systems”. *Chemical and Biochemical Engineering Quarterly*, Vol. 18, No. 3, pp. 195–222, 2004.

[Marw 08] W. Marwan, A. Wagler, and R. Weismantel. “A mathematical approach to solve the network reconstruction problem”. *Mathematical Methods of Operations Research*, Vol. 67, pp. 117–132, 2008. 10.1007/s00186-007-0178-5.

[Marw 11] W. Marwan, A. Wagler, and R. Weismantel. “Petri nets as a framework for the reconstruction and analysis of signal transduction pathways and regulatory networks”. Vol. 10, No. 2, pp. 639–654, June 2011.

[Marw 12] W. Marwan, C. Rohr, and M. Heiner. *Petri nets in Snoopy: a unifying framework for the graphical display, computational modelling, and simulation of bacterial regulatory networks*, Chap. 21, pp. 409–437. Vol. 804 of *Methods in Molecular Biology*, Humana Press, 2012.

[Mats 06] H. Matsuno, C. Li, and S. Miyano. “Petri net based descriptions for systematic understanding of biological pathways”. *IEICE - Transactions on Fundamentals of Electronics, Communications and Computer Sciences website*, Vol. E89-A, pp. 3166–3174, November 2006.

[Mauc 03] H. Mauch. “Evolving petri nets with a genetic algorithm”. In: *Proceedings of the 2003 international conference on Genetic and evolutionary computation: PartII*, pp. 1810–1811, Springer-Verlag, Berlin, Heidelberg, 2003.

[Mayo 05] M. Mayo. “Learning petri net models of non-linear gene interactions”. *Biosystems*, Vol. 82, No. 1, pp. 74 – 82, 2005.

[Mayo 11] M. Mayo and L. Beretta. “Modelling epistasis in genetic disease using petri nets, evolutionary computation and frequent itemset mining”. *Expert Systems with Applications*, Vol. 38, No. 4, pp. 4006 – 4013, 2011.

[Mend 09a] P. Mendes, S. Hoops, S. Sahle, R. Gauges, J. Dada, and U. Kummer. “Computational modeling of biochemical networks using COPASI”. *Methods in molecular biology (Clifton, N.J.)*, Vol. 500, pp. 17–59, 2009.

[Mend 09b] P. Mendes, H. Messiha, N. Malys, and S. Hoops. “Enzyme kinetics and computational modeling for systems biology”. *Methods in enzymology*, Vol. 467, pp. 583–599, 2009.

[Meng 12] Y. Meng and H. Guo. “Evolving network motifs based morphogenetic approach for self-organizing robotic swarms”. In: *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference*, pp. 137–144, ACM, New York, NY, USA, 2012.

[Mole 03] C. G. Moles, P. Mendes, and J. R. Banga. “Parameter estimation in biochemical pathways: A comparison of global optimization methods”. *Genome Research*, Vol. 13, No. 11, pp. 2467–2474, Nov. 2003.

[Moor 03] J. H. Moore and L. W. Hahn. “Petri net modeling of high-order genetic systems using grammatical evolution”. *BioSystems*, Vol. 72, No. 1-2, pp. 177–186, Nov. 2003.

[Mora 98] M. Morange. *A history of molecular biology*. Cambridge, MA: Harvard University Press, 1998.

[Mukh 09] S. Mukherji and A. van Oudenaarden. “Synthetic biology: understanding biological design from synthetic circuits”. *Nat Rev Genet*, Vol. 10, No. 12, pp. 859–871, Dec. 2009.

[Mura 89] T. Murata. “Petri nets: Properties, analysis and applications”. *Proceedings of the IEEE*, Vol. 77, No. 4, pp. 541–580, April 1989.

[Nobl 60] D. Noble. “Cardiac action and pacemaker potentials based on the Hodgkin-Huxley equations”. *Nature*, Vol. 188, 1960.

[Numm 05] J. Nummela and B. A. Julstrom. “Evolving petri nets to represent metabolic pathways”. In: *Proceedings of the 2005 conference on Genetic and evolutionary computation*, pp. 2133–2139, ACM, New York, NY, USA, 2005.

[Paul 03] J. ao Paulo Barros and L. Gomes. “Modifying petri net models by means of crosscutting operations”. *Application of Concurrency to System Design, International Conference on*, Vol. 0, p. 177, 2003.

[Paun 00] G. Păun. “Computing with membranes”. *Journal of Computer and System Sciences*, Vol. 61, No. 1, pp. 108–143, 2000.

[Paun 06] G. Păun and M. J. Pérez-Jiménez. “Membrane computing: Brief introduction, recent results and applications”. *Biosystems*, Vol. 85, No. 1, pp. 11 – 22, 2006.

[Paun 98] G. Păun. “Computing with membranes”. Tech. Rep. 208, Turku Center for Computer Science-TUCS, 1998.

[Paun 99] G. Păun. “Computing with membranes. an introduction”. *Bulletin of the EATCS*, No. 67, pp. 139–152, February 1999.

[Pele 05] M. Peleg, D. Rubin, and R. B. Altman. “Using petri net tools to study properties and dynamics of biological systems”. *Journal of the American Medical Informatics Association*, Vol. 12, No. 2, pp. 181–199, 2005.

[Poly 11] A. Polyvyanyy, M. Weidlich, and M. Weske. “Connectivity of workflow nets: the foundations of stepwise verification”. *Acta Informatica*, Vol. 48, pp. 213–242, 2011.

[R De 09] R Development Core Team. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, 2009.

[Rand 08] R. Randhawa. *Model composition and aggregation in macromolecular regulatory networks*. PhD thesis, Faculty of the Virginia Polytechnic Institute and State University, 2008.

[Rech 65] I. Rechenberg. “Cybernetic solution path of an experimental problem”. Tech. Rep., Royal Air Force Establishment, 1965.

[Rech 73] I. Rechenberg. *Evolutions strategie: optimierung technischer systeme nach prinzipien der biologischen evolution*. 1973.

[Redd 93] V. N. Reddy, M. L. Mavrovouniotis, and M. N. Liebman. “Petri net representations in metabolic pathways”. In: *Proceedings of the 1st International Conference on Intelligent Systems for Molecular Biology*, pp. 328–336, AAAI Press, 1993.

[Redd 96] V. N. Reddy, M. N. Liebman, and M. L. Mavrovouniotis. “Qualitative analysis of biochemical reaction systems”. *Computers in Biology and Medicine*, Vol. 26, No. 1, pp. 9 – 24, 1996.

[Rodr 07a] G. Rodrigo, J. Carrera, and A. Jaramillo. “Asmparts: assembly of biological model parts”. *Systems and synthetic biology*, Vol. 1, No. 4, pp. 167–170, Dec. 2007.

[Rodr 07b] G. Rodrigo, J. Carrera, and A. Jaramillo. "Genetdes: automatic design of transcriptional networks.". *Bioinformatics*, Vol. 23, No. 14, pp. 1857–1858, 2007.

[Rohr 10] C. Rohr, W. Marwan, and M. Heiner. "Snoopy - a unifying Petri net framework to investigate biomolecular networks". *Bioinformatics*, Vol. 26, No. 7, pp. 974–975, 2010.

[Rome 08] F. Romero-Campero, H. Cao, M. Camara, and N. Krasnogor. "Structure and parameter estimation for cell systems biology models". In: M. K. et.al, Ed., *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2008)*, pp. 331–338, ACM Publisher, 2008.

[Rome 09] F. J. Romero-Campero, J. Twycross, M. Camara, M. Bennett, M. Gheorghe, and N. Krasnogor. "Modular assembly of cell systems biology models using P systems". *International Journal of Foundations of Computer Science*, Vol. 20, No. 3, pp. 427–442, 2009.

[Ruz 10] G. A. Ruz and E. Goles. "Learning gene regulatory networks with predefined attractors for sequential updating schemes using simulated annealing". In: *Proceedings of the 2010 Ninth International Conference on Machine Learning and Applications*, pp. 889–894, IEEE Computer Society, Washington, DC, USA, 2010.

[Sack 06] A. Sackmann, M. Heiner, and I. Koch. "Application of petri net based analysis techniques to signal transduction pathways.". *BMC Bioinformatics*, Vol. 7, No. 1, pp. 482–498, Nov. 2006.

[Sahl 06] S. Sahle, R. Gauges, J. Pahle, N. Simus, U. Kummer, S. Hoops, C. Lee, M. Singhal, L. Xu, and P. Mendes. "Simulation of biochemical networks using COPASI: a complex pathway simulator". In: *WSC '06: Proceedings of* *the 37th conference on Winter simulation*, pp. 1698–1706, Winter Simulation Conference, 2006.

[Sava 69a] M. A. Savageau. “Biochemical systems analysis. I. Some mathematical properties of the rate law for the component enzymatic reactions.”. *Journal of Theoretical Biology*, Vol. 25, No. 3, pp. 365–369, Dec. 1969.

[Sava 69b] M. A. Savageau. “Biochemical systems analysis: II. The steady-state solutions for an n-pool system using a power-law approximation”. *Journal of Theoretical Biology*, Vol. 25, No. 3, pp. 370 – 379, 1969.

[Sava 70] M. A. Savageau. “Biochemical systems analysis: III. Dynamic solutions using a power-law approximation”. *Journal of Theoretical Biology*, Vol. 26, No. 2, pp. 215 – 226, 1970.

[Sava 76] M. A. Savageau. *Biochemical systems analysis : a study of function and design in molecular biology*. Addison-Wesley Pub. Co., Advanced Book Program, Reading, Mass., 1976.

[SBML 12] SBML-Team. “The systems biology markup language (SBML)”. `http://sbml.org/Main_Page`, September 2012.

[Schm 04] J. Schmid, K. Mauch, M. Reuss, E. Gilles, and A. Kremling. “Metabolic design based on a coupled gene expression-metabolic network model of tryptophan production in Escherichia coli.”. *Metabolic Engineering*, Vol. 6, No. 4, pp. 364–77, 2004.

[Schw 65] H.-P. Schwefel. *Cybernetic evolution as strategy for experimental research in fluid mechanics (in German)*. PhD thesis, Hermann-Fttinger Institute for Fluid Mechanics, Technical University Berlin, March 1965.

[Schw 75] H. P. Schwefel. “Evolutionsstrategie und numerische optimierung”. *PhD Thesis*, 1975.

[Spie 04] C. Spieth, F. Streichert, N. Speer, and A. Zell. "Optimizing topology and parameters of gene regulatory network models from time-series experiments". In: *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2004)*, pp. 461–470, 2004.

[Step 12] G. Stephanopoulos. "Synthetic biology and metabolic engineering". *ACS Synth. Biol.*, Nov. 2012.

[Stre 04] F. Streichert, H. Planatscher, C. Spieth, H. Ulmer, and A. Zell. "Comparing genetic programming and evolution strategies on inferring gene regulatory networks". In: K. Deb, R. Poli, W. Banzhaf, H.-G. Beyer, E. K. Burke, P. J. Darwen, D. Dasgupta, D. Floreano, J. A. Foster, M. Harman, O. Holland, P. L. Lanzi, L. Spector, A. Tettamanzi, D. Thierens, and A. M. Tyrrell, Eds., *Genetic and Evolutionary Computation Conference - GECCO 2004*, pp. 471–480, Springer Verlag, Seattle, Washington, USA, June 26-30 2004.

[Suen 04] A. Suenaga, A. B. Kiyatkin, M. Hatakeyama, N. Futatsugi, N. Okimoto, Y. Hirano, T. Narumi, A. Kawai, R. Susukita, T. Koishi, H. Furusawa, K. Yasuoka, N. Takada, Y. Ohno, M. Taiji, T. Ebisuzaki, J. B. Hoek, A. Konagaya, and B. N. Kholodenko. "Tyr-317 phosphorylation increases Shc structural rigidity and reduces coupling of domain motions remote from the phosphorylation site as revealed by molecular dynamics simulations". *Journal of Biological Chemistry*, Vol. 279, No. 6, pp. 4657–62, Feb. 2004.

[Sun 12] J. Sun, J. M. Garibaldi, and C. Hodgman. "Parameter estimation using metaheuristics in systems biology: A comprehensive review". *IEEE Transactions on Computational Biology and Bioinformatics*, Vol. 9, No. 1, pp. 185–202, Jan. 2012.

[Suzu 83] I. Suzuki and T. Murata. “A method for stepwise refinement and abstraction of petri nets”. *Journal of Computer and System Sciences*, Vol. 27, pp. 51–76, 1983.

[Talb 09] E.-G. Talbi. *Metaheuristics: From design to implementation*. Wiley Publishing, 2009.

[Tana 04] A. Tanay, R. Sharan, M. Kupiec, and R. Shamir. “Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data”. *Proceedings of the National Academy of Sciences of the United States of America*, Vol. 101, No. 9, pp. 2981–2986, March 2004.

[Tayl 03] C. F. Taylor, N. W. Paton, K. L. Garwood, P. D. Kirby, D. A. Stead, Z. Yin, E. W. Deutsch, L. Selway, J. Walker, I. Riba-Garcia, S. Mohammed, M. J. Deery, J. A. Howard, T. Dunkley, R. Aebersold, D. B. Kell, K. S. Lilley, P. Roepstorff, J. R. Yates, A. Brass, A. J. Brown, P. Cash, S. J. Gaskell, S. J. Hubbard, and S. G. Oliver. “A systematic approach to modeling, capturing, and disseminating proteomics experimental data”. *Nature Biotechnology*, Vol. 21, No. 3, pp. 247–254, March 2003.

[Thom 12] S. Thomas and Y. Jin. “Combining genetic oscillators and switches using evolutionary algorithms”. In: *Computational intelligence in bioinformatics and computational biology (CIBCB), 2012 IEEE Symposium on*, may 2012.

[Thor 15] W. M. Thorburn. “Occam’s Razor”. *Mind*, Vol. 24, No. 2, pp. 287–288, 1915.

[Toms 06] J. Tomshine and Y. Kaznessis. “Optimization of a stochastically-integrated gene network model via simulated annealing”. *Biophysical Journal*, Vol. 91, pp. 3196–3205, Aug. 2006.

[Vale 79] R. Valette. “Analysis of petri nets by stepwise refinements”. *Journal of Computer and System Sciences*, Vol. 18, pp. 35–46, 1979.

[Vall 10] R. R. Vallabhajosyula and A. Raval. “Computational modeling in systems biology”. In: J. M. Walker and Q. Yan, Eds., *Systems Biology in Drug Discovery and Development*, Chap. 5, pp. 97–120, Humana Press, Totowa, NJ, 2010.

[Vija 09] W. M. H. D. S. B. Vijaysekhar Chellaboina, Sanjay P. Bhat. “Modeling and analysis of mass-action kinetics”. *IEEE Control Systems Magazine*, Vol. 29, No. 4, pp. 60–78, August 2009.

[Vlad 04] M. O. Vlad, A. Arkin, and J. Ross. “Response experiments for nonlinear systems with application to reaction kinetics and genetics.”. *Proceedings of the National Academy of Sciences*, Vol. 101, No. 19, pp. 7223–7228, May 2004.

[Voet 06] D. Voet and J. G. Voet. *Biochemistry*. John Wiley & Sons, 2006.

[Voig 12] C. A. Voigt. “Synthetic biology”. *ACS Synth. Biol.*, Vol. 1, No. 1, pp. 1–2, Jan. 2012.

[Vysh 08] V. Vyshemirsky and M. Girolami. “Bayesian ranking of biochemical system models”. *Bioinformatics*, Vol. 24, No. 6, pp. 833–839, 2008.

[Wagl 11] A. K. Wagler and R. Weismantel. “The combinatorics of modeling and analyzing biological systems”. Vol. 10, No. 2, pp. 655–681, June 2011.

[Wang 04] T. Wang, J. W. Touchman, and G. Xue. “Applying two-Level simulated annealing on bayesian structure learning to infer genetic networks”. In: *Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference*, pp. 647–648, IEEE Computer Society, Washington, DC, USA, 2004.

[West 04] H. V. Westerhoff and B. O. Palsson. “The evolution of molecular biology into systems biology.”. *Nature Biotechnology*, Vol. 22, No. 10, pp. 1249–1252, Oct. 2004.

[Wien 65] N. Wiener. *Cybernetics or the control and communication in the animal and the machine*. The MIT Press,Cambridge, 1965.

[Wolp 97] D. H. Wolpert and W. G. Macready. “No free lunch theorems for optimization”. *IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, pp. 67–82, Apr. 1997.

[Wu 10] Z. Wu, Q. Gao, and D. Gilbert. “Target driven biochemical network reconstruction based on petri nets and simulated annealing”. In: *Proceedings of the 8th International Conference on Computational Methods in Systems Biology*, pp. 33–42, ACM, New York, NY, USA, 2010.

[Wu 12] Z. Wu, S. Yang, and D. Gilbert. “A hybrid approach to piecewise modelling of biochemical systems”. In: C. Coello, V. Cutello, K. Deb, S. Forrest, G. Nicosia, and M. Pavone, Eds., *Parallel Problem Solving from Nature - PPSN XII*, pp. 519–528, Springer Berlin Heidelberg, 2012.

[Yeun 00] K. Yeung, P. Janosch, B. McFerran, D. W. Rose, H. Mischak, J. M. Sedivy, and W. Kolch. “Mechanism of suppression of the Raf/MEK/Extracellular signal-regulated kinase pathway by the Raf kinase inhibitor protein”. *Molecular and Cellular Biology*, Vol. 20, No. 9, pp. 3079–3085, 2000.

[Yeun 99] K. Yeung, T. Seitz, S. Li, P. Janosch, B. McFerran, C. Kaiser, F. Fee, K. D. Katsanakis, D. W. Rose, H. Mischak, J. M. Sedivy, and W. Kolch. “Suppression of Raf-1 kinase activity and MAP kinase signaling by RKIP”. *Nature*, Vol. 401, pp. 173–177, 1999.

[Zeve 03] I. Zevedei-Oancea and S. Schuster. “Topological analysis of metabolic networks based on Petri net theory.”. *In Silico Biology*, Vol. 3, No. 3, pp. 323–345, 2003.

[Zhou 92] M. Zhou, F. DiCesare, and A. A. Desrochers. “A hybrid methodology for synthesis of petri net models for manufacturing systems”. *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 3, pp. 350–361, June 1992.

[Zi 06] Z. Zi and E. Klipp. “SBML-PET: a systems biology markup language-based parameter estimation tool”. *Bioinformatics (Oxford, England)*, Vol. 22, No. 21, pp. 2704–2705, Nov. 2006.

[Zube 99] W. M. Zuberek. “Stepwise refinements of net models and their place invariants”. In: *Proceedings of the The 8th International Workshop on Petri Nets and Performance Models (PNPM’99), 8-10 October 1999, Zaragoza, Spain*, pp. 92–101, IEEE Computer Society, Washington, DC, USA, 1999.

Appendix A

Statistical Analysis of Two-sample Tests in R

Common operation of comparing aspects of two samples in R is implementation of two-sample tests. An example is given to illustrate how to obtain information of two given samples. Consider the following sets of data on the latent heat of the fusion of ice [R De 09].

Method A:	79.98 80.04 80.02 80.04 80.03 80.03 80.04 79.97
	80.05 80.03 80.02 80.00 80.02
Method B:	80.02 79.94 79.98 79.97 79.97 80.03 79.95 79.97

To test for the equality of the means of the two examples, we can use an unpaired t-test by ‘Welch Two Sample t-test’ as follows.

```
> t.test(A, B)
data: A and B
t = 3.2499, df = 12.027, p-value = 0.00694
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval: 0.01385526 0.07018320
sample estimates:
mean of x = 80.02077
mean of y = 79.97875
```

which does indicate a significant difference, assuming normality. By default the R function does not assume equality of variances in the two samples. We can use the F test to test for equality in the variances, provided that the two samples are from normal populations. Details of the test are given as follows.

```
> var.test(A, B)
data: A and B
F = 0.5837, num df = 12, denom df = 7, p-value = 0.3938
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval: 0.1251097 2.1052687
sample estimates: ratio of variances = 0.5837405
```

The about analysis result shows no evidence of a significant difference.

Appendix B

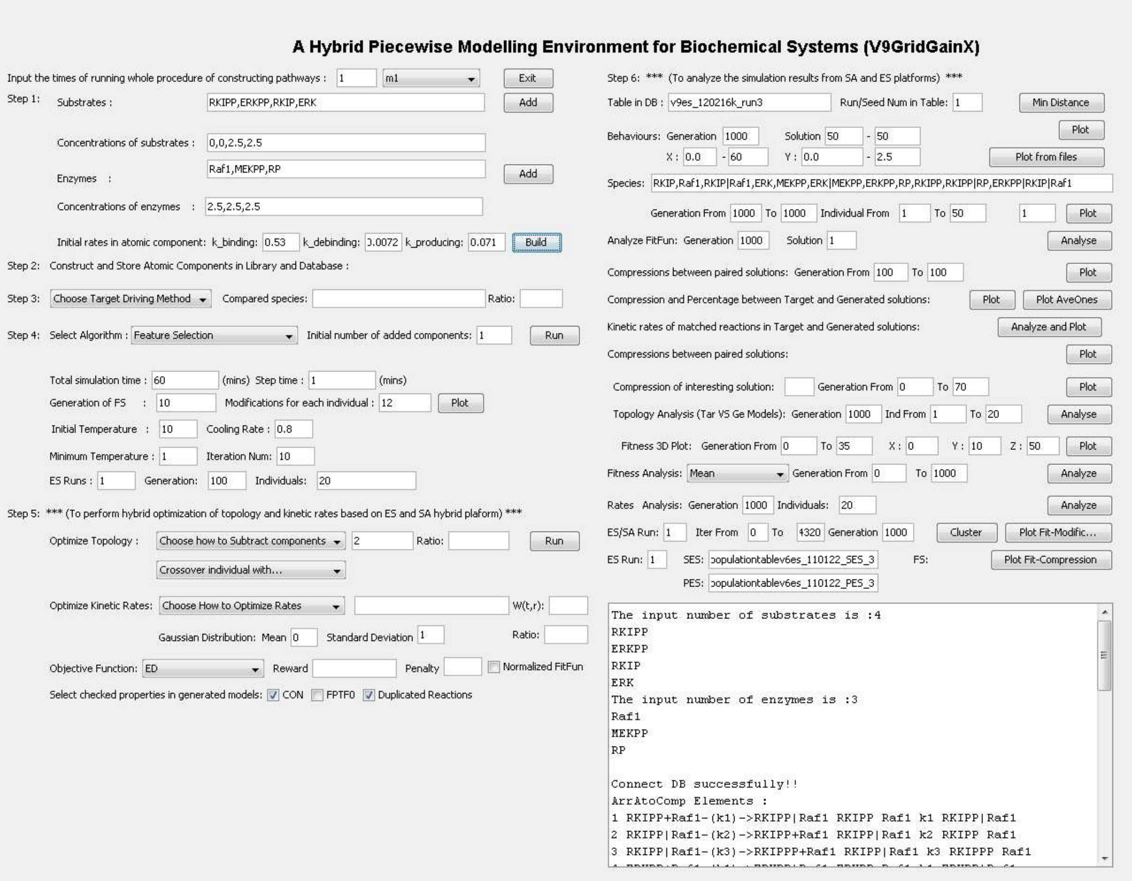
A Hybrid Piecewise Modelling Environment

In order to run piecewise modelling of biochemical systems in a Java based modelling environment, see Figure B.1, we need to input elements as substrates and enzymes for generation of components by a combination mechanism. For instance, ‘*RKIPP*, *ERKPP*, *RKIP* and *ERK*’ are input as substrates, and ‘*Raf1*, *MEKPP* and *RP*’ are input as enzymes.

After instantiating components by pre-defined two binding and unbinding patterns, there is a library for preserving these instantiated components. As shown in Figure B.2, instantiated component with details of reactants, products and kinetic rates are preserved in the library.

While composing models in a piecewise manner, models can be composed and preserved in a models library. As shown in Figure B.3, final optimized synthetic models are kept for investigating details of the composed models. The models library provides information about the optimization results, for instance models obtained at which generation (RandomNum) and model candidates (IterationNum) in population pool, what are the fitness values of these models (DeltaDistance) compared with the target biochemical system, what are the topologies (GenerateODEs) and what are the simulation results for exhibiting species behaviour (SimulationResult).

Details of composed topology and corresponding simulation result of a synthetic model



A Hybrid Piecewise Modelling Environment for Biochemical Systems (V9GridGainX)

Input the times of running whole procedure of constructing pathways : 1 m1

Step 1: Substrates : RKIPP,ERKPP,RKIP,ERK

Concentrations of substrates : 0,0,2.5,2.5

Enzymes : Raf1,MEKPP,RP

Concentrations of enzymes : 2.5,2.5,2.5

Initial rates in atomic component: k_binding: 0.53 k_debinding:).0072 k_producing: 0.071

Step 2: Construct and Store Atomic Components in Library and Database :

Step 3: Choose Target Driving Method Compared species: Ratio:

Step 4: Select Algorithm : Feature Selection Initial number of added components: 1

Total simulation time : 60 (mins) Step time : 1 (mins)

Generation of FS : 10 Modifications for each individual : 12

Initial Temperature : 10 Cooling Rate : 0.8

Minimum Temperature : 1 Iteration Num: 10

ES Runs : 1 Generation: 100 Individuals: 20

Step 5: *** (To perform hybrid optimization of topology and kinetic rates based on ES and SA hybrid plaform) ***

Optimize Topology : Choose how to Subtract components 2 Ratio:

Crossover individual with...

Optimize Kinetic Rates: Choose How to Optimize Rates W(t,r):

Gaussian Distribution: Mean 0 Standard Deviation 1 Ratio:

Objective Function: ED Reward Penalty □ Normalized FitFun

Select checked properties in generated models: ☑ CON □ FPTF0 ☑ Duplicated Reactions

Step 6: *** (To analyze the simulation results from SA and ES platforms) ***

Table in DB : v9es_120216k_run3 Run/Seed Num in Table: 1

Behaviours: Generation 1000 Solution 50 - 50

X : 0.0 - 60 Y : 0.0 - 2.5

Species: RKIP,Raf1,RKIP|Raf1,ERK,MEKPP,ERK|MEKPP,ERKPP,RP,RKIPP,RKIPP|RP,ERKPP|RKIP|Raf1

Generation From 1000 To 1000 Individual From 1 To 50 1

Analyze FitFun: Generation 1000 Solution 1

Compressions between paired solutions: Generation From 100 To 100

Compression and Percentage between Target and Generated solutions:

Kinetic rates of matched reactions in Target and Generated solutions:

Compressions between paired solutions:

Compression of interesting solution: Generation From 0 To 70

Topology Analysis (Tar VS Ge Models): Generation 1000 Ind From 1 To 20

Fitness 3D Plot: Generation From 0 To 35 X : 0 Y : 10 Z : 50

Fitness Analysis: Mean Generation From 0 To 1000

Rates Analysis: Generation 1000 Individuals: 20

ES/SA Run: 1 Iter From 0 To 4320 Generation 1000

ES Run: 1 SES: ɔopulationtablev6es_110122_SES_3 FS:

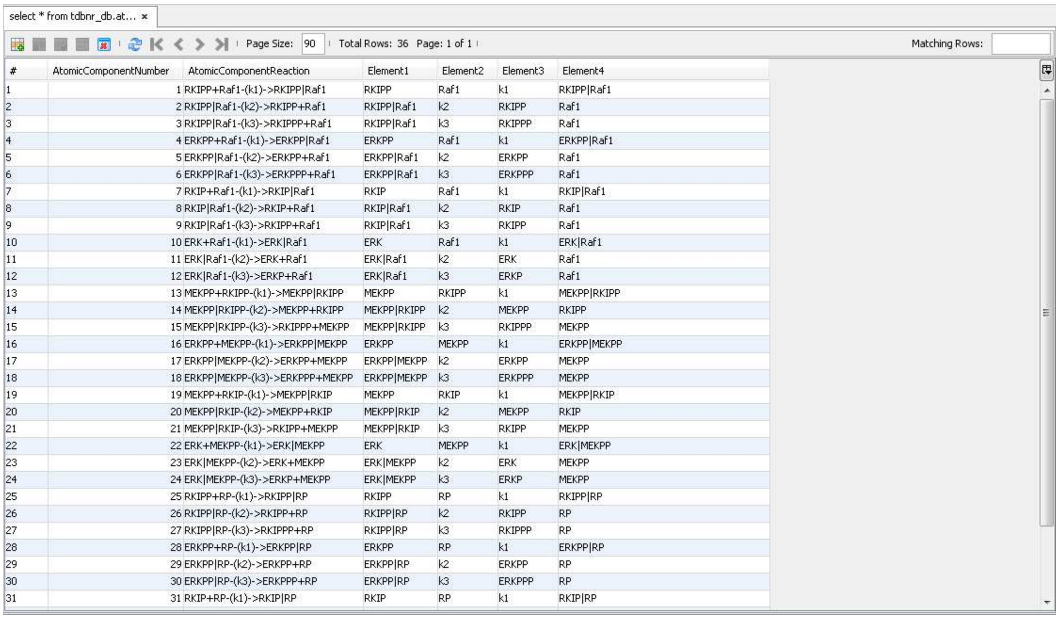
PES: ɔopulationtablev6es_110122_PES_3

```
The input number of substrates is :4
RKIPP
ERKPP
RKIP
ERK
The input number of enzymes is :3
Raf1
MEKPP
RP

Connect DB successfully!!
ArrAtoComp Elements :
1 RKIPP+Raf1-(k1)->RKIPP|Raf1 RKIPP Raf1 k1 RKIPP|Raf1
2 RKIPP|Raf1-(k2)->RKIPP+Raf1 RKIPP|Raf1 k2 RKIPP Raf1
3 RKIPP|Raf1-(k3)->RKIPPP+Raf1 RKIPP|Raf1 k3 RKIPPP Raf1
```

Figure B.1: A hybrid piecewise modelling environment for biochemical systems.

can be found in the cells indexed with ‘GenerateODEs’ and ‘SimulationResult’ in the models library, respectively. Thus, we can get a set of ODEs which is mathematical description of a synthetic model and mapped from its topology presented in a component style in ‘GenerateODEs’ column, as shown in Figure B.4. Moreover, simulation of composed models can examine generated species behaviour of these models, which behaviour are presented in the time series data format, as shown in Figure B.5.

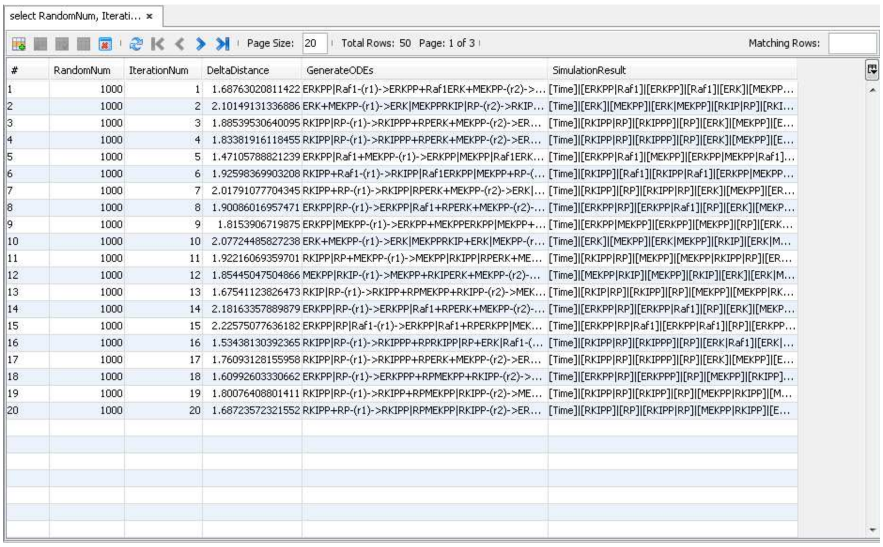


select * from tdbnr_db.at...

Page Size: 90 Total Rows: 36 Page: 1 of 1 Matching Rows:

<table>
<tr><th>#</th><th>AtomicComponentNumber</th><th>AtomicComponentReaction</th><th>Element1</th><th>Element2</th><th>Element3</th><th>Element4</th></tr>
<tr><td>1</td><td>1</td><td>RKIPP+Raf1-(k1)->RKIPP|Raf1</td><td>RKIPP</td><td>Raf1</td><td>k1</td><td>RKIPP|Raf1</td></tr>
<tr><td>2</td><td>2</td><td>RKIPP|Raf1-(k2)->RKIPP+Raf1</td><td>RKIPP|Raf1</td><td>k2</td><td>RKIPP</td><td>Raf1</td></tr>
<tr><td>3</td><td>3</td><td>RKIPP|Raf1-(k3)->RKIPPP+Raf1</td><td>RKIPP|Raf1</td><td>k3</td><td>RKIPPP</td><td>Raf1</td></tr>
<tr><td>4</td><td>4</td><td>ERKPP+Raf1-(k1)->ERKPP|Raf1</td><td>ERKPP</td><td>Raf1</td><td>k1</td><td>ERKPP|Raf1</td></tr>
<tr><td>5</td><td>5</td><td>ERKPP|Raf1-(k2)->ERKPP+Raf1</td><td>ERKPP|Raf1</td><td>k2</td><td>ERKPP</td><td>Raf1</td></tr>
<tr><td>6</td><td>6</td><td>ERKPP|Raf1-(k3)->ERKPPP+Raf1</td><td>ERKPP|Raf1</td><td>k3</td><td>ERKPPP</td><td>Raf1</td></tr>
<tr><td>7</td><td>7</td><td>RKIP+Raf1-(k1)->RKIP|Raf1</td><td>RKIP</td><td>Raf1</td><td>k1</td><td>RKIP|Raf1</td></tr>
<tr><td>8</td><td>8</td><td>RKIP|Raf1-(k2)->RKIP+Raf1</td><td>RKIP|Raf1</td><td>k2</td><td>RKIP</td><td>Raf1</td></tr>
<tr><td>9</td><td>9</td><td>RKIP|Raf1-(k3)->RKIPP+Raf1</td><td>RKIP|Raf1</td><td>k3</td><td>RKIPP</td><td>Raf1</td></tr>
<tr><td>10</td><td>10</td><td>ERK+Raf1-(k1)->ERK|Raf1</td><td>ERK</td><td>Raf1</td><td>k1</td><td>ERK|Raf1</td></tr>
<tr><td>11</td><td>11</td><td>ERK|Raf1-(k2)->ERK+Raf1</td><td>ERK|Raf1</td><td>k2</td><td>ERK</td><td>Raf1</td></tr>
<tr><td>12</td><td>12</td><td>ERK|Raf1-(k3)->ERKP+Raf1</td><td>ERK|Raf1</td><td>k3</td><td>ERKP</td><td>Raf1</td></tr>
<tr><td>13</td><td>13</td><td>MEKPP+RKIPP-(k1)->MEKPP|RKIPP</td><td>MEKPP</td><td>RKIPP</td><td>k1</td><td>MEKPP|RKIPP</td></tr>
<tr><td>14</td><td>14</td><td>MEKPP|RKIPP-(k2)->MEKPP+RKIPP</td><td>MEKPP|RKIPP</td><td>k2</td><td>MEKPP</td><td>RKIPP</td></tr>
<tr><td>15</td><td>15</td><td>MEKPP|RKIPP-(k3)->RKIPPP+MEKPP</td><td>MEKPP|RKIPP</td><td>k3</td><td>RKIPPP</td><td>MEKPP</td></tr>
<tr><td>16</td><td>16</td><td>ERKPP+MEKPP-(k1)->ERKPP|MEKPP</td><td>ERKPP</td><td>MEKPP</td><td>k1</td><td>ERKPP|MEKPP</td></tr>
<tr><td>17</td><td>17</td><td>ERKPP|MEKPP-(k2)->ERKPP+MEKPP</td><td>ERKPP|MEKPP</td><td>k2</td><td>ERKPP</td><td>MEKPP</td></tr>
<tr><td>18</td><td>18</td><td>ERKPP|MEKPP-(k3)->ERKPPP+MEKPP</td><td>ERKPP|MEKPP</td><td>k3</td><td>ERKPPP</td><td>MEKPP</td></tr>
<tr><td>19</td><td>19</td><td>MEKPP+RKIP-(k1)->MEKPP|RKIP</td><td>MEKPP</td><td>RKIP</td><td>k1</td><td>MEKPP|RKIP</td></tr>
<tr><td>20</td><td>20</td><td>MEKPP|RKIP-(k2)->MEKPP+RKIP</td><td>MEKPP|RKIP</td><td>k2</td><td>MEKPP</td><td>RKIP</td></tr>
<tr><td>21</td><td>21</td><td>MEKPP|RKIP-(k3)->RKIPP+MEKPP</td><td>MEKPP|RKIP</td><td>k3</td><td>RKIPP</td><td>MEKPP</td></tr>
<tr><td>22</td><td>22</td><td>ERK+MEKPP-(k1)->ERK|MEKPP</td><td>ERK</td><td>MEKPP</td><td>k1</td><td>ERK|MEKPP</td></tr>
<tr><td>23</td><td>23</td><td>ERK|MEKPP-(k2)->ERK+MEKPP</td><td>ERK|MEKPP</td><td>k2</td><td>ERK</td><td>MEKPP</td></tr>
<tr><td>24</td><td>24</td><td>ERK|MEKPP-(k3)->ERKP+MEKPP</td><td>ERK|MEKPP</td><td>k3</td><td>ERKP</td><td>MEKPP</td></tr>
<tr><td>25</td><td>25</td><td>RKIPP+RP-(k1)->RKIPP|RP</td><td>RKIPP</td><td>RP</td><td>k1</td><td>RKIPP|RP</td></tr>
<tr><td>26</td><td>26</td><td>RKIPP|RP-(k2)->RKIPP+RP</td><td>RKIPP|RP</td><td>k2</td><td>RKIPP</td><td>RP</td></tr>
<tr><td>27</td><td>27</td><td>RKIPP|RP-(k3)->RKIPPP+RP</td><td>RKIPP|RP</td><td>k3</td><td>RKIPPP</td><td>RP</td></tr>
<tr><td>28</td><td>28</td><td>ERKPP+RP-(k1)->ERKPP|RP</td><td>ERKPP</td><td>RP</td><td>k1</td><td>ERKPP|RP</td></tr>
<tr><td>29</td><td>29</td><td>ERKPP|RP-(k2)->ERKPP+RP</td><td>ERKPP|RP</td><td>k2</td><td>ERKPP</td><td>RP</td></tr>
<tr><td>30</td><td>30</td><td>ERKPP|RP-(k3)->ERKPPP+RP</td><td>ERKPP|RP</td><td>k3</td><td>ERKPPP</td><td>RP</td></tr>
<tr><td>31</td><td>31</td><td>RKIP+RP-(k1)->RKIP|RP</td><td>RKIP</td><td>RP</td><td>k1</td><td>RKIP|RP</td></tr>
</table>

Figure B.2: A library for preserving instantiated components for composition.

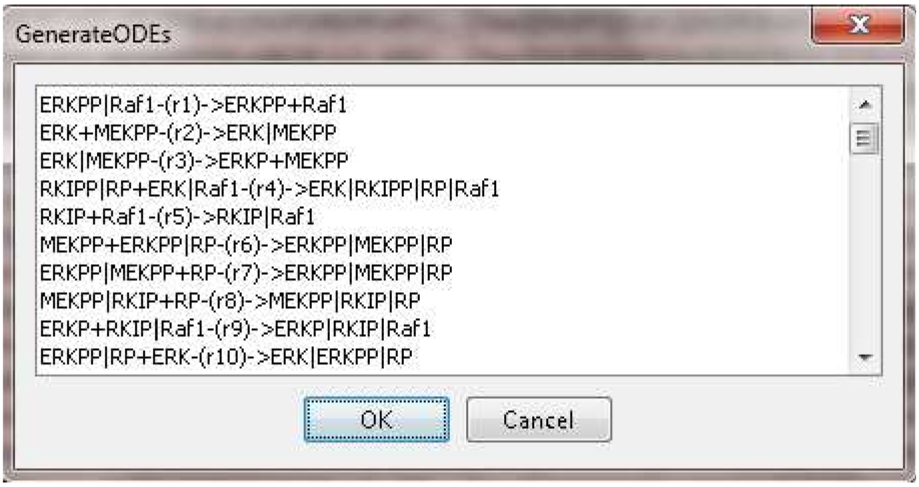


select RandomNum, Iterati...

Page Size: 20 Total Rows: 50 Page: 1 of 3 Matching Rows:

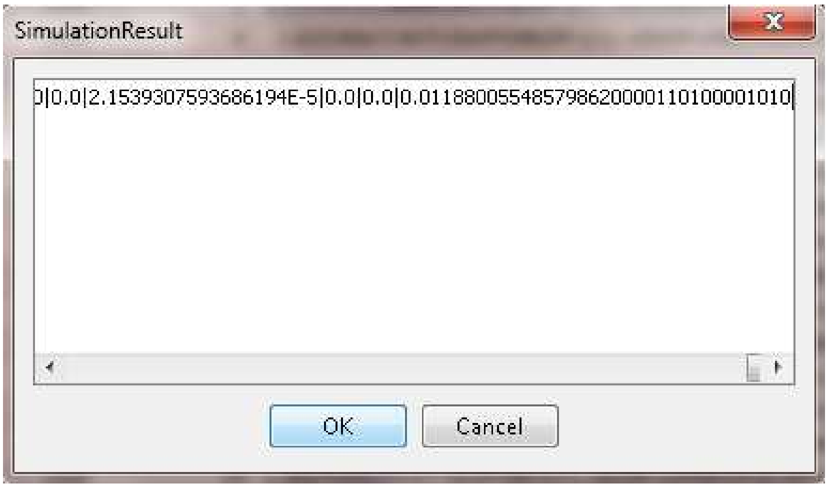
<table>
<tr><th>#</th><th>RandomNum</th><th>IterationNum</th><th>DeltaDistance</th><th>GenerateODEs</th><th>SimulationResult</th></tr>
<tr><td>1</td><td>1000</td><td>1</td><td>1.68763020811422</td><td>ERKPP|Raf1-(r1)->ERKPP+Raf1ERK+MEKPP-(r2)->...</td><td>[Time]|[ERKPP|Raf1]|[ERKPP]|[Raf1]|[ERK]|[MEKPP...</td></tr>
<tr><td>2</td><td>1000</td><td>2</td><td>2.10149131336886</td><td>ERK+MEKPP-(r1)->ERK|MEKPPRKIP|RP-(r2)->RKIP...</td><td>[Time]|[ERK]|[MEKPP]|[ERK|MEKPP]|[RKIP|RP]|[RKI...</td></tr>
<tr><td>3</td><td>1000</td><td>3</td><td>1.88539530640095</td><td>RKIPP|RP-(r1)->RKIPPP+RPERK+MEKPP-(r2)->ER...</td><td>[Time]|[RKIPP|RP]|[RKIPPP]|[RP]|[ERK]|[MEKPP]|[E...</td></tr>
<tr><td>4</td><td>1000</td><td>4</td><td>1.83381916118455</td><td>RKIPP|RP-(r1)->RKIPPP+RPERK+MEKPP-(r2)->ER...</td><td>[Time]|[RKIPP|RP]|[RKIPPP]|[RP]|[ERK]|[MEKPP]|[E...</td></tr>
<tr><td>5</td><td>1000</td><td>5</td><td>1.47105788821239</td><td>ERKPP|Raf1+MEKPP-(r1)->ERKPP|MEKPP|Raf1ERK...</td><td>[Time]|[ERKPP|Raf1]|[MEKPP]|[ERKPP|MEKPP|Raf1]...</td></tr>
<tr><td>6</td><td>1000</td><td>6</td><td>1.92598369903208</td><td>RKIPP+Raf1-(r1)->RKIPP|Raf1ERKPP|MEKPP+RP-(...</td><td>[Time]|[RKIPP]|[Raf1]|[RKIPP|Raf1]|[ERKPP|MEKPP...</td></tr>
<tr><td>7</td><td>1000</td><td>7</td><td>2.01791077704345</td><td>RKIPP+RP-(r1)->RKIPP|RPERK+MEKPP-(r2)->ERK|...</td><td>[Time]|[RKIPP]|[RP]|[RKIPP|RP]|[ERK]|[MEKPP]|[ER...</td></tr>
<tr><td>8</td><td>1000</td><td>8</td><td>1.90086016957471</td><td>ERKPP|RP-(r1)->ERKPP|Raf1+RPERK+MEKPP-(r2)-...</td><td>[Time]|[ERKPP|RP]|[ERKPP|Raf1]|[RP]|[ERK]|[MEKP...</td></tr>
<tr><td>9</td><td>1000</td><td>9</td><td>1.8153906719875</td><td>ERKPP|MEKPP-(r1)->ERKPP+MEKPPERKPP|MEKPP+...</td><td>[Time]|[ERKPP|MEKPP]|[ERKPP]|[MEKPP]|[RP]|[ERK...</td></tr>
<tr><td>10</td><td>1000</td><td>10</td><td>2.07724485827238</td><td>ERK+MEKPP-(r1)->ERK|MEKPPRKIP+ERK|MEKPP-(r...</td><td>[Time]|[ERK]|[MEKPP]|[ERK|MEKPP]|[RKIP]|[ERK|M...</td></tr>
<tr><td>11</td><td>1000</td><td>11</td><td>1.92216069359701</td><td>RKIPP|RP+MEKPP-(r1)->MEKPP|RKIPP|RPERK+ME...</td><td>[Time]|[RKIPP|RP]|[MEKPP]|[MEKPP|RKIPP|RP]|[ER...</td></tr>
<tr><td>12</td><td>1000</td><td>12</td><td>1.85445047504866</td><td>MEKPP|RKIP-(r1)->MEKPP+RKIPERK+MEKPP-(r2)-...</td><td>[Time]|[MEKPP|RKIP]|[MEKPP]|[RKIP]|[ERK]|[ERK|M...</td></tr>
<tr><td>13</td><td>1000</td><td>13</td><td>1.67541123826473</td><td>RKIP|RP-(r1)->RKIPP+RPMEKPP+RKIPP-(r2)->MEK...</td><td>[Time]|[RKIP|RP]|[RKIPP]|[RP]|[MEKPP]|[MEKPP|RK...</td></tr>
<tr><td>14</td><td>1000</td><td>14</td><td>2.18163357889879</td><td>ERKPP|RP-(r1)->ERKPP|Raf1+RPERK+MEKPP-(r2)-...</td><td>[Time]|[ERKPP|RP]|[ERKPP|Raf1]|[RP]|[ERK]|[MEKP...</td></tr>
<tr><td>15</td><td>1000</td><td>15</td><td>2.22575077636182</td><td>ERKPP|RP|Raf1-(r1)->ERKPP|Raf1+RPERKPP|MEK...</td><td>[Time]|[ERKPP|RP|Raf1]|[ERKPP|Raf1]|[RP]|[ERKPP...</td></tr>
<tr><td>16</td><td>1000</td><td>16</td><td>1.53438130392365</td><td>RKIPP|RP-(r1)->RKIPPP+RPRKIPP|RP+ERK|Raf1-(...</td><td>[Time]|[RKIPP|RP]|[RKIPPP]|[RP]|[ERK|Raf1]|[ERK|...</td></tr>
<tr><td>17</td><td>1000</td><td>17</td><td>1.76093128155958</td><td>RKIPP|RP-(r1)->RKIPPP+RPERK+MEKPP-(r2)->ER...</td><td>[Time]|[RKIPP|RP]|[RKIPPP]|[RP]|[ERK]|[MEKPP]|[E...</td></tr>
<tr><td>18</td><td>1000</td><td>18</td><td>1.60992603330662</td><td>ERKPP|RP-(r1)->ERKPPP+RPMEKPP+RKIPP-(r2)->...</td><td>[Time]|[ERKPP|RP]|[ERKPPP]|[RP]|[MEKPP]|[RKIPP]...</td></tr>
<tr><td>19</td><td>1000</td><td>19</td><td>1.80076408801411</td><td>RKIPP|RP-(r1)->RKIPP+RPMEKPP|RKIPP-(r2)->ME...</td><td>[Time]|[RKIPP|RP]|[RKIPP]|[RP]|[MEKPP|RKIPP]|[M...</td></tr>
<tr><td>20</td><td>1000</td><td>20</td><td>1.68723572321552</td><td>RKIPP+RP-(r1)->RKIPP|RPMEKPP|RKIPP-(r2)->ER...</td><td>[Time]|[RKIPP]|[RP]|[RKIPP|RP]|[MEKPP|RKIPP]|[E...</td></tr>
</table>

Figure B.3: A library for preserving composed models.



```
ERKPP|Raf1-(r1)->ERKPP+Raf1
ERK+MEKPP-(r2)->ERK|MEKPP
ERK|MEKPP-(r3)->ERKP+MEKPP
RKIPP|RP+ERK|Raf1-(r4)->ERK|RKIPP|RP|Raf1
RKIP+Raf1-(r5)->RKIP|Raf1
MEKPP+ERKPP|RP-(r6)->ERKPP|MEKPP|RP
ERKPP|MEKPP+RP-(r7)->ERKPP|MEKPP|RP
MEKPP|RKIP+RP-(r8)->MEKPP|RKIP|RP
ERKP+RKIP|Raf1-(r9)->ERKP|RKIP|Raf1
ERKPP|RP+ERK-(r10)->ERK|ERKPP|RP
```

Figure B.4: An example of generated ODEs illustrating a composed model.



```
0|0.0|2.1539307593686194E-5|0.0|0.0|0.011880055485798620000110100001010|
```

Figure B.5: Results of simulating a composed model.