12-2012

Modeling of Yeast Pheromone Pathway using Petri Nets

Abhishek Majumdar
University of Nebraska-Lincoln, majumdar@cse.unl.edu

Modeling of Yeast Pheromone Pathway using Petri Nets

by

Abhishek Majumdar

Presented to the Faculty of

The Graduate College at the University of Nebraska

In Partial Fulfillment of Requirements

For the Degree of Master of Science

Major: Computer Science

Under the Supervision of Professor Stephen D. Scott and

Professor Jitender Singh Deogun

Lincoln, Nebraska

December, 2012

Modeling of Yeast Pheromone Pathway using Petri Nets

Abhishek Majumdar, M.S.

University of Nebraska, 2012

Advisors: Stephen D. Scott and Jitender Singh Deogun

Yeast (*Saccharomyces cerevisiae*) is one of the most widely studied single celled organisms. Mating of yeast cells occur between cells of opposite mating types **a** and α. Pheromone secretion by a cell alerts the corresponding opposite type cell about its presence and eventually facilitates the process of mating between them. The details of how pheromones affect cells can be studied from the pheromone response pathway in a yeast cell. A response pathway typically depicts the chain of interactions that happens between the different proteins in the cells in response to the pheromone. In this thesis we model the yeast pheromone response pathway using Petri nets and simulate various conditions under which a cell will respond positively to the secreted pheromone. We also take a look at how different proteins in the cell might be able to facilitate the correct functionality of the pathway. The objective of this thesis is to serve as a guideline for performing lab experiments to further explore the yeast pheromone pathway.

ACKNOWLEDGEMENTS

My heart felt gratitude goes to my advisors, Prof. Stephen D. Scott and Prof. Jitender S. Deogun, for their continuous support, guidance and encouragement during the thesis. Without their patience, motivation, guidance, immense knowledge and encouragement it would not have been possible for me to complete the thesis.

I would like to thank Prof. Steven Harris, for his support during the thesis. His expertise has been a guiding star in the development of this thesis.

I would like to thank Dr. Monica Heiner of Brandenburg University of Technology, Cottbus, Germany for her advise on setting up the experiments.

Contents

Chapter 1

Introduction

Our environment is composed of tightly interlinked complex systems at different levels of magnitude. The smallest biological level of detail is the molecular level of DNA, RNA, proteins and metabolites. All these molecules together form a cell. Cells form tissues. Different tissues constitute the organs of an organism. Different organisms together form the ecosystem. So the relationships between different biological elements at any level of detail can be represented as a network called a biological network [6]. To date, five different types of biological networks have been characterized: transcription factor binding, protein-protein interactions, protein phosphorylation, metabolic interactions and genetic interaction networks. Transcription is the process of copying DNA to mRNA. A transcription factor is a protein which performs this function acting alone or with other agents in a complex either as an activator or a repressor. Protein phosphorylation is a post-translational modification of proteins by addition of phosphate groups. Metabolic interaction networks focus on the mass flow in basic chemical pathways that generate essential components such as amino acids, sugars, lipids etc and the energy required by the biochemical reactions. Gene interaction or protein interaction networks, as the names indicate, reflect the interaction between

the different genes/proteins [16]. Modeling of biological networks helps to come up with an overall view about the state of biological knowledge of the network and also to reveal important properties that might remain hidden without any proper models. Also, these models can be used to predict behaviors about the network which can then be verified with experimental results.

Yeasts are single celled microorganisms in the Fungi kingdom. *Saccharomyces cerevisiae* a particular species of yeast, has been widely studied in genetics and cell biology. It is a simple eukaryote cell which serves as a model for all eukaryotes for the study of fundamental cellular processes like cell cyle, cell division, metabolism, DNA replication, etc. [5] *S. cerevisiae* has both asexual and sexual reproduction. Sexual reproduction takes place between two haploid cells of opposite types **a** and α. The process of mating is initiated by secretion of pheromone by one of the cells. Receptors on the opposite cell detect the presence of pheromone and initiates a series of protein-protein interactions within the cell which ultimate might facilitate mating. This series of protein-protein interactions in the cell is known as the yeast pheromone pathway. This pathway is a well studied problem where the basic structure and different proteins participating in the pathway are now known. So this is essentially a protein-protein interaction network.

In this thesis we use Petri nets [14] to model this protein interaction network. To this end we propose a model of the pheromone pathway (adopted from Sackmann et al. [15]), modify it according to our needs, and augment it with some additional proteins which we believe might have some accessory role in the process [9]. In absence of consistent real world data of the different kd values of the different protein interactions we generate those randomly from the range $\{1, 2, \ldots, 100\}$ [9]. We simulate the pathway using our model to identify the conditions under which yeast cells will mate. The contributions of the thesis include putting together the model, identifying a set

of rules that determines, whether a cell would successfully mate or not and finally identifying a possible process used by yeast cell to overcome detrimental conditions and successfully mate.

Chapter 2

Background

2.1 Pheromone Response Process in Yeast

In this chapter we describe the process of pheromone binding to its receptor on the cell surface and the subsequent effects of that phenomenon on the cell functionality. The chapter is based on the textbook by H. Madhani [12]. The mating process is initiated, when a yeast cell detects the presence of pheromone secreted by a cell of opposite sex. Mating consists of sequence of processes; activation of gene expression, arrest of cell cycle, polarized growth of *shmoo* (cellular projection in yeast cell) towards the mating partner, cell-cell fusion and nuclear fusion. There are two cell types in yeast called **a** and α. They are analogous to egg and sperm cells of animals. These **a** and α cells can mate to produce an **a**/α cell. This in turn undergoes meiosis to produce the haploid gametes (child cells) **a** and α cells. An **a** cell produces a pheromone known as *a-factor* whereas an α cell produces a pheromone known as α-*factor*. An **a** cell contains the α-*factor* receptor Ste2 whereas an α cell contains the *a-factor* receptor Ste3. So **a** cells can mate with α cells only and vice-versa.

For both Ste2 and Ste3, binding to one of the mating pheromone affects its abil-

ity to interact with an intracellular protein complex known as the heterotrimeric G protein. The G protein is made up of subunits called Gpa1, Ste4 and Ste18. In the remainder of this thesis these subunits will be referred to as G_α, G_β, and G_γ, respectively. In the G-protein complex, the G_β and G_γ units form a complex $G_{\beta\gamma}$, which remain bound to G_α when it is bound to GDP. When pheromone binds to the receptor (Ste2 or Ste3), the receptor interacts with G_α, causing it to replace its GDP with GTP. G_α without its GDP cannot keep the $G_{\beta\gamma}$ complex bound to itself. As a result the $G_{\beta\gamma}$ complex is liberated and goes on to interact with other proteins. Gradually GTP bound to G_α is hydrolyzed to GDP by G_α. It then binds back and inhibits the $G_{\beta\gamma}$ complex.

The $G_{\beta\gamma}$ complex when liberated activates a pathway in which four protein kinases are linked in a series to form a cascade kinase. The $G_{\beta\gamma}$ recruits the protein Ste5 which acts as a scaffold to hold three other proteins Ste11, Ste7 and Fus3. These three proteins activate each other in series by phosphorylating them. So an activated Ste11 phosphorylates Ste7, which becomes active and in turn phosphorylates Fus3. The activated Fus3 then leaves the scaffold and enters the nucleus. This final activation triggers the arrest of the cell cycle as well as the transcription of genes involved in mating. The Ste11 at the top of the kinase is activated by a protein Ste20. The protein Ste20 itself becomes activated when it is in the plasma membrane where it is phosphorylated by a membrane associated monomeric GTPase called Cdc42.

Activated Fus3 plays an important role in both cell cyle arresting as well as the transcription of genes. Activated Fus3 phosphorylates protein Far1 which blocks the cell cycle in G_1 phase, to prepare for mating. Fus3 in the nucleus activates the transciption factor Ste12. Ste12 promotes the expression of **a**-specific and α-specific genes. Normally, in absence of pheromone signal, Ste12 is inhibited by proteins Dig1 and Dig2, which bind to different parts of Ste12 (*Dig1 to the DNA-binding domain of*

Ste12 and Dig2 to the carboxyl terminus required for DNA-bound Ste12 to activate transcription). Due to pheromone signalling, activated Fus3 phosphorylatytes Dig1 and Dig2 which in turn release Ste12. The Ste12 is then free to bind and promote the transcription of **a**-specific genes (***a****-sgs*) and α-specific genes (α-*sgs*).

A prominent feature of mating is the formation of shmoo by the cells towards each other. The pheromone receptors most activated on the cell surface are the ones facing the highest concentration of pheromone. This region contains the highest concentration of $G_\beta\gamma$. The $G_\beta\gamma$ complex recruits proteins to promote the formation of the shmoo. Far1 binds to the $G_\beta\gamma$ complex and recruits three proteins Cdc42, Cdc24 and Bem1 which promote polarized cell growth. Cdc24 activates Cdc42 by inducing transformation of GDP to GTP. The Cdc42 bound to GTP together with Bem1, recruit proteins which promote cell membrane growth such as Bni1 and others. Thus a series of protein-protein interactions beginning with active $G_\beta\gamma$ result in the growth of cell membrane towards the highest concentration of pheromone.

To re-enter cell cycle after successful mating or a failed mating attempt, yeast cells have some kind of built-in mechanism which can counter the effects of pheromone exposure on the different proteins. This is typically achieved through various proteins acting in negative feedback loops. Implementation of such loops is beyond the scope of this thesis and hence not discussed here.

2.2 Petri nets

2.2.1 Introduction

Petri nets were first proposed by Carl Adam Petri in 1962. Petri nets are a graphical and mathematical tool applicable to many systems. They can be used for describing and modeling systems that can be characterized as concurrent, asynchronous,

distributed, parallel, non-deterministic, and/or stochastic. As a graphical tool they provide the functionality of flowcharts or block diagrams. Petri nets allow the simulation and visualization of dynamic activities of systems by the use of tokens in its modeling. As a mathematical tool they can be used to represent state equations and other mathematical models governing the behaviour of systems. The following discussion is based mostly on the paper by T. Murata [14].

A Petri net is a kind of a directed graph with an initial marking M_0. The underlying graph of a Petri net is a directed weighted bipartite graph with two kinds of nodes, *places* and *transitions*. Places are represented as circles and transitions are represented as boxes. The arcs are either from places to transitions or vice-versa. The arcs or edges are labeled with weights (positive integers). A unit weight edge is not labeled. A marking M is an assignment of non-negative integers to the places. If a marking assigns integer x to place p, then p is said to be marked with x tokens. Pictorially tokens are denoted by dots. Transitions cannot be assigned any integers. If a Petri net has m places, then a marking is a vector of size m. The p^{th} element of M, denoted $M(p)$ is the number of tokens in place p.

2.2.2 Definitions

In modelling, places represent conditions and transitions represent events. The formal definition of a Petri net is given below.

A Petri net can be defined as a 5-tuple, $PN = (P, T, E, W, M_0)$, where:

$P = \{p_1, p_2, \ldots, p_m\}$ is a finite set of places,

$T = \{t_1, t_2, \ldots, t_n\}$ is a finite set of transitions,

$E \subseteq (P \times T) \cup (T \times P)$ is a set of edges,

$W: E \rightarrow 1, 2, 3, ...$ is a weight function,

$M_0: P \rightarrow 0, 1, 2, ...$ is the initial marking.

$P \cap T = \emptyset$ and $P \cup T \neq \emptyset$.

A Petri net without any initial marking is denoted as N while that with a given initial marking is denoted by (N, M_0). Below we define some terminologies related to Petri nets. A *preplace* of a transition t, denoted $\mathbf{pre(}\boldsymbol{t}\mathbf{)}$ is defined as a place from which there is an incoming edge to the t. Mathematically,

$$\mathbf{pre(t)} = \{p \mid (p, t) \in E\}.$$

Similarly a *postplace* of a transition t, denoted $\mathbf{post(}\boldsymbol{t}\mathbf{)}$ is defined as a place to which there is an edge from t. Mathematically,

$$\mathbf{post(t)} = \{p \mid (t, p) \in E\}.$$

Similar notations are available for places as well. We define the *pre-transition* of a place p as

$$\mathbf{pre(p)} = \{t \mid (t, p) \in E\}$$

and *post-transition* of a place p as

$$\mathbf{post(p)} = \{t \mid (p, t) \in E\}.$$

In order to simulate the behaviour of a system, the Petri net is operated using the following firing rules:

1. A transition t is said to be enabled if each preplace p of t has at least $w(p, t)$ tokens where $w(p, t)$ is the weight of the edge in between p and t.

2. An enabled transition may or may not fire (depending on whether the event actually occurs or not)

3. Once transition t has fired, each of its preplace p lose $w(p,t)$ tokens and each of its postplaces p' gains $w(t,p')$ tokens.

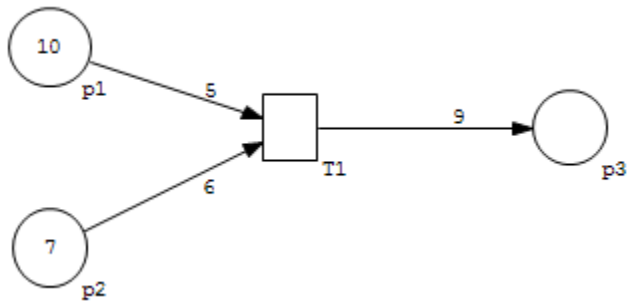

Figure 2.1: A simple Petri net

In Figure 2.1 we show an example of a Petri net with 3 places p_1, p_2 and p_3 and a transition T_1. p_1 has 10 tokens and p_2 has 7 tokens, $w(p_1,T_1)=5$, $w(p_2,T_1)=6$ and $w(T_1,p_3)=9$. By the above mentioned rule 1, T_1 is enabled. Thus when T_1 fires, the Petri net will change as shown in Figure 2.2. After T_1 has fired, p_1 has 5 tokens, p_2 has 1 token and p_3 has 9 tokens (rule 3).

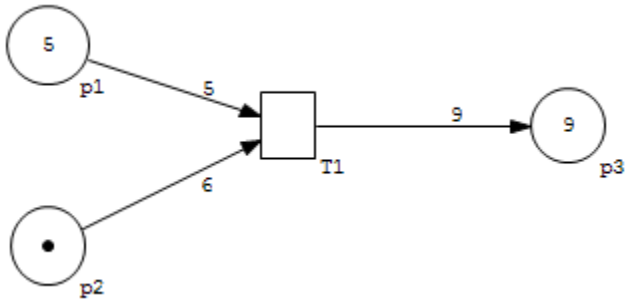

Figure 2.2: The Petri net of Figure 2.1 after T1 fires

2.2.3 Properties

Two types of properties can be studied with a Petri net: *Behavioral properties*, which depend on the initial markings and *Structural properties*, which depend on the topo-

logical structure of the Petri net. We will first discuss some of the behavioral properties and then the structural.

Behavioral Properties

The following properties have been adopted from the paper by T. Murata [14].

1. **Reachability**: A marking M_n in a Petri net is said to be reachable from another marking M_0 if there exists a sequence of transition firings that results in transforming the marking M_0 to M_n. The set of all markings reachable from M_0 is denoted by $R(M_0)$.

2. **Boundedness**: A Petri net (N, M_0) is said to be *k-bounded* if the number of tokens in any place does not exceed some positive integer k for any marking reachable from M_0.

3. **Liveness**: A Petri net (N, M_0) is said to be *live* if irrespective of the current marking of the net, any transition can eventually be fired by progressing through some firing sequence.

4. **Reversibility**: A Petri net is said to be *reversible* if any initial marking M_0 can be reached from all markings reachable from M_0.

5. **Coverability**: A marking M in a Petri net is said to be *coverable* if there exists another marking $M' \in R(M_0)$ such that $M'(p) > M_0(p)$ for each place p in the net.

6. **Persistence**: A Petri net (N, M_0) is said to be persistent if, for any two enabled transitions, the firing of one transition will not disable the other.

Structural Properties

Structural properties of a Petri net are the properties which are independent of the initial marking M_0. These are dependent on the topology of the net. Before describing the properties, we introduce the concept of *incidence matrix* in a Petri net. For a Petri net N with n transitions and m places, the incidence matrix $A = [a_{ij}]$ is an $n \times m$ matrix of integers where each matrix entry is given by

$$a_{ij} = a_{ij}^{+} - a_{ij}^{-},$$

where $a_{ij}^{+} = w(i,j)$ is the weight of the edge from transition i to place j and $a_{ij}^{-} = w(j,i)$ is the weight of the edge from place j to transition i. Some papers use A^T as the incidence matrix instead of A. The structural properties of a Petri net can be represented in terms of its incidence matrix A and its associated homogeneous equations or inequalities. For the following properties, the net is assumed to be free of self loops. For any vector z, the i^{th} element of z is denoted by $z(i)$.

1. **Structural Liveness**: A Petri net N is said to be *structurally live* if there exists a live initial marking for N.

2. **Controllability**: A Petri net N is said to be *completely controllable* if any marking is reachable from any other marking.

3. **Structural Boundedness**: A Petri net is said to be *structurally bounded* if it is bounded for any finite initial marking M_0.

4. **Conservativeness**: A Petri net is said to be (partially) *conservative* if there exists a positive integer $y(p)$ for (some) every place p such that the weighted sum of the tokens $M^T y = M_0^T y$ is a constant for every $M \in R(M_0)$ and for any fixed initial marking M_0.

5. **Repetitiveness**: A Petri net is said to (partially) *repetitive* if there exists an initial marking M_0 and a firing sequence ϖ such that (some) every transition occurs often in ϖ.

6. **Consistency**: A Petri net is said to be (partially) *consistent* if there exists a marking M_0 and a firing sequence ϖ from M_0 to M_0 such that (some) every transition occurs atleast once in ϖ.

Invariants

- P-invariant: An m-vector y of integers is called a *P-invariant* if $A\mathbf{y} = \mathbf{0}$.
- T-invariant: An n-vector x of integers is called a *T-invariant* if $A^T\mathbf{x} = \mathbf{0}$.

The set of places corresponding to a non-zero entry in a P-invariant is called the support of an invariant. A support is said to be *minimal* if no other support is contained in it. An invariant $\mathbf{y}$ is said to be a *minimal invariant* if there exists no other invariant y_1 such that $y_1(p) \leq y(p)$ for all p. For a given minimal support of an invariant there exists a unique invariant corresponding to the minimal support. This is known as *minimal support invariant*. Linear combinations of minimal-support invariants yield new invariants.

Chapter 3

Problem Description

The yeast pheromone response pathway is one of the most well-studied biological pathways. Over the years, ample experiments have been performed which have allowed biologists to piece together the different components of the pathway. As a result we now have a working knowledge of how the pathway functions, the different proteins that take part in this pathway and their respective roles. However, several questions still remain unanswered. One particular question is: how does the cell dynamically adapt the pathway to continue mating under severe environmental changes or under mutation (which might result in the loss of functionality of some proteins known to participate in the pheromone pathway). That is, it is not entirely known how the cell compensates, uses other proteins to contribute towards functioning of the pathway if the original pathway is compromised.

This thesis attempts to answer the above mentioned question. To achieve this, we first propose a model to simulate the pheromone pathway. Then we look at the following two questions.

1. Given the model of the pheromone response pathway, under what conditions does the cell respond positively, i.e., mate?

2. What kind of perturbations in the cell would result in changing a negative response to a positive one?

In our model, the "conditions" mentioned in question 1 typically refers to the different edges weights between the different components of the pathway. Different combinations of the values of the edge weights represent different environmental conditions faced by the cell. The best way to answer question 1 would be to come up with a set of rules that can be used to predict whether a cell will be able to successfully mate or not. "Perturbations" mentioned in question 2 refers to possible methods employed by the cell so that it can mate. We hypothesise that one such method might be to make use of accessory proteins who otherwise are not so prominent in the pheormone pathway. Using appropriate amounts of proteins other than the core pathway component proteins can be a possible compensation method used by the cell to facilitate mating. To answer question 2 we try to come up with a list of accessor proteins and figure out their functionality. The combined answers of questions 1 and 2 are expected to provide a clearer understanding of this process.

Chapter 4

Related Work

In this chapter we discuss related work. We survey some of the papers in which a Petri net approach has been used to model biological networks.

Sachmann et al. [15] provide a systemic modeling method of signal transduction pathways in terms of Petri net components. The authors present a process of representing the following three different cases of a signal transduction model.

Case 1: A substance A does not lose its activity by interacting with a second substance B.

Case 2: A substance C triggers several reactions that are independent of each other.

Case 3: A substance changes state from being phosphorylated to being unphosphorylated and vice versa.

Case1 indicates phosphorylation reactions between different proteins in a network. Case2 describes participation of a protein in multiple independent reactions. Both cases are implemented by using read arcs (bidirectional edges between places and transitions) in their Petri net representations. Case3 indicates the different states of a protein, which is implemented in form of a sub-network. Having described these, the authors propose the following simple steps for representing a signal pathway. First,

translate the biological components into logical strucures like conjunction, disjunction, exclusive disjunction and implication. Second, translate the logical structures in corresponding Petri net forms. Finally, assimilate the Petri net components to form a whole network. Our work uses the modeling approach used by this paper and forms the basic structure of our model on the model provided in this paper.

Chaouiya [7] provides an overview of the different types of Petri net models available and their uses in modeling different types of biological networks. A **Coloured Petri Net** (CPN) uses different colored tokens to represent different data types. They also allow assigning expressions to the edges. For firing of a transition, the corresponding expression on the incoming edges to the transition are to be satisfied. Similarly, to determine the output of a transition, the expression on its outgoing edge needs to be calculated. CPNs have been used to simulate enzymatic reaction chains. A **Stochastic Petri Net** (SPN) allows time delay functions to be associated with the firing of transitions. This allows SPNs to incorporate uncertainty and noise into its modeling. SPNs have been used to model and simulate stochastic molecular interactions. **Hybrid Petri Nets** (HPNs) allow modeling of both discrtete and continuous processes. For discrete events they have discrete places with tokens and transitions while for continuos processes they have continuous places with real variables and continuous transitions which can fire continuously at a given rate. To increase their expressiveness, they also have read arcs and inhibitor arcs (a transition will fire if the concentration(s) of its pre-place(s) are zero). **Hybrid Function Petri Nets** (HFPNs) have some added functionality in addition to those of HPNs. Continuous transition firing rates can be made dependent on the value of its pre-places and the weights of the arcs can be defined as a function of the marking of the connected places. HPNs have been used for quantitative modeling and simulation of gene regulatory networks. Several regulated metabolic pathways have been modelled using HFPNs.

The paper concludes with discussion that the use of Petri nets for modeling metabolic reactions is relatively intuitive. CPNs are capable of representing gene interaction or signal transduction networks, while HPNs support representation of a wide range of molecular mechanism.

Hardy and Robillard [8] also discuss the different types of Petri nets extensions used for analysis, modeling and simulation of molecular biology networks. They identify two categories of goals of Petri net biological modeling: **qualitative** and **quantitative** analysis. Qualitative analysis is the analysis of the different biological properties while quantitative analysis is the simulation of system dynamics. For qualitative analysis, a suitable Petri net extension should be chosen that allows its various different properties to be determined. For quantitative analysis, a Petri net representation with sufficient modeling power should be chosen. The authors provide a summary of different modeling goals for the different Petri net extensions. CPNs are good for analysis of biological system properties while SPNs and HFPNs are good for simulation purposes. For quantitative analysis of a biological system, kinetic parameters like reaction rates and stoichiometric quantities of reactants are necessary. In this thesis since no such data was available, we used the basic Petri net structure for our quantitative analysis. In the future, pending availability of data, we plan to upgrade our model to a HFPN or something similar.

Monica et al. [10] demonstrate a generalized approach towards modeling and analysis of biological pathways using Petri nets. For model validation they utilize one of the basic behavioral properties of Petri nets: T-Invariants. To demonstrate the effectiveness of their approach they model the apoptosis (cell death) process. For modeling, they follow a few basic rules. Biochemical substances are represented by places and any relation between biochemical substances are represented by transitions with corresponding arcs. Signal tranduction events and enzyme catalytic reactions are

modelled using test arcs. For model validation, the authors use T-Invariants, which in the context of biological networks are sets of reactions reproducing a given system state. To facilitate validation, the paper changes the model into an empty Petri net where all input output nodes are transitions. For such a net, the T-Invariants are the sets of transitions reproducing the empty marking. Using T-Ivariant analysis they are successfully able to account for all the known basic behaviors of the process hence validating their model.

Chapter 5

Modeling method

5.1 Model

We use Petri nets to model the pheromone response pathway. We adopt and modify the approach mentioned in the papers by Heiner et al. [10] and Sachmann [15]. We represent each protein as a place in the Petri net and each interaction as a transition. Hence a reaction R between proteins p_A and p_B to produce protein p_C can be represented as the following net structure. In Figure 5.1, the places A and B represent the

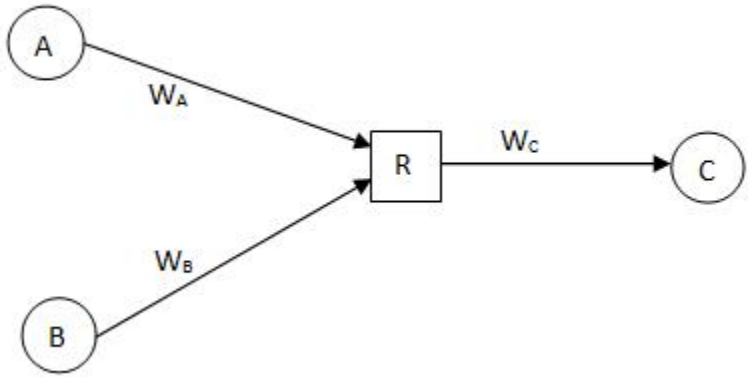

Figure 5.1: Petrinet representation of a reaction process

proteins p_A and p_B and transition R represents the reaction between A and B that forms protein p_C (place C). The tokens in each place represent the initial concentration of each protein. Also, the edge weights between R and the pre-places represent concentration thresholds to be exceeded by individual protein. The firing rules for transitions in Figure 5.1 are given by:

1. Number of tokens from A $\geq W_A$ **AND**

2. Number of tokens from B $\geq W_B$

Using this representation as the basic unit of construction, all reactions in the pathway can be modeled and the full pathway can be obtained by combining these individual reaction representations. Such a model is already available in the paper by Sachmann et al [15]. We base our model on this avaiable network structure [15]. We make several changes to this model to suit our solution approach. In the following section we describe those changes and modifications and the rationale behind them.

5.1.1 Modifications

In a Petri net, a transition can fire if it is *enabled*. By that definition in Figure 5.1, for reaction R to fire and form compound C, both A and B must have at least W_A and W_B tokens each, respectively. So essentially this is a conjunction rule where all pre-places must be present and have some pre-requisite number of tokens for a reaction to occur. We propose a method to convert all such conjunction rules into corresponding disjunction rules.

We know that the reaction between two or more proteins takes place if the corresponding kd value exceeds a certain threshold value. As an example, say proteins A and B interact to form C only when the strength of their interaction (kd value)

is more than, say, W_{AB}. The conventional Petri net representation as shown in Figure 5.1 does not allow for this kd value concept to be implemented properly. For this reason we have come up with the disjunctive representation of reactions in Petri nets. The same reaction involving A, B and C can be represented in Figure 5.2. To Figure 5.1, we have added two dummy post-transitions D_A and D_B to A and B

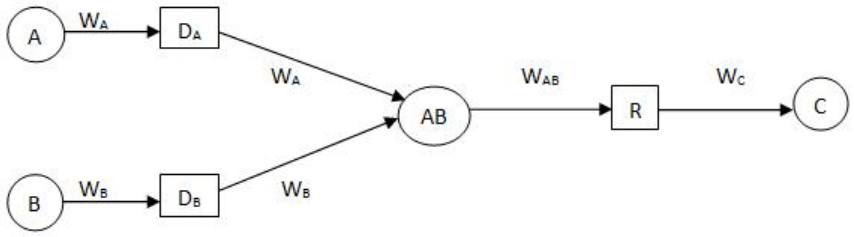

Figure 5.2: Disjunctive Representation

respectively and another preplace AB to transition R. The weight of the edge from place AB to transition R is W_{AB} which makes sure that R will fire only if the number of tokens coming from its pre-place exceeds the threshold value. The firing rules of the different transitions in Figure 5.2 are as follows:

1. D_A: Number of tokens from A $\geq W_A$
2. D_B: Number of tokens from B $\geq W_B$
3. R: Number of tokens from AB $\geq W_{AB} \Rightarrow W_A + W_B \geq W_{AB}$

Comparing Figure 5.1 and Figure 5.2 we see that they essentially represent the same reaction. This notion can be generalized to a reaction with n participants. Assume a reaction R in which n proteins $p_1, p_2, \ldots, p_i, \ldots, p_n$ interact to produce a compound q. Let W_R be the threshold value that needs to be exceeded by the cumulative weight/concentration of the reactants for R to proceed. Let $w_1, w_2, \ldots, w_i, \ldots, w_n$

be the weights/concentrations of the reacting proteins respectively. The disjunctive representation of this reaction can be drawn as follows. For each protein p_i, construct a dummy transition D_i where both the incoming and outgoing edges have weight w_i. Construct a pre-place S to R such all dummy transitions become pre-transitions to S. Now let the edge weight between S and R be W_R. Thus the firing rules for the different transitions become:

1. D_i: Number of tokens from $p_i \geq w_i \; \forall$ i $\in \{1.....n\}$

2. R: Number of tokens from S $\geq W_R \Rightarrow \sum_{i=1}^{n} w_i \geq W_R$

The firing rule for R imposes the threshold condition that we want to be included while representing reactions. Also it gives one added advantage. It allows the assumption of weighted sum of the components for successful execution of the reaction. That is, all reactants no longer need to be present in its entirety for the reaction to proceed. As long as the cumulative concentration of the reactants exceeds the reaction threshold, it will work, which makes more sense as in nature that is what happens [9].

5.1.2 Additions

We augment the above model by adding more proteins that exist in the yeast cell, and are believed (not yet proved) to play some role in the pheromone response process. The proteins taking part in the pheromone response process are known to interact physically with other proteins. We looked up the *Saccharomyces* Genome Database [3] for all proteins which are known to have physical interaction with the core protein components (Ste5, Ste11, Ste7, Ste20, Ste50, Fus3, Dig1, Dig2, Ste12, Sst2, Far1, Cdc24, Cdc42, Bem1, Ste2, Ste3, Ste4, Ste18, GPA1 and Tec1) of the pathway which are already known to us. For each of those proteins we then look up their properties to see if they are known to play some role in pheromone interaction pathways. We

retain all proteins who are known to have some role related to pheromone exchange or those whose functions are unknown. The rationale of adding these proteins is to give the model enough options so that it can to some extent simulate how the cell might use these proteins in the pheromone pathway.

Chapter 6

Petri net representation of the pathway

6.1 Model

We use the pheromone response pathway structure provided by Sachmann et al [15]. In that representation each protein component is represented by a place and each reaction by a transition. We modify this model so as to incorporate the idea of requisite kd values for the different reactions. To this end we transform the preplaces of all transitions to a single place (marked red in our model) which has inputs from different reactant places. To conform to the Petri Net design, we add a dummy transition to each reactant place as shown in the previous section. Only for transitions with Ste-type proteins as pre-places are left unchanged. The benefit of having a single pre-place to a transition which originally required several pre-places is that it emphasizes the notion of weighted cumulative concentration of the reactants. Table 6.2 [15] gives a list of all the protein components and their symbols used in our model. Table 6.4 [15] gives a list of all the transitions, their symbols and biological reactions

that they represent. We then add more proteins that are known to interact with various component proteins of the pheromone pathway. We obtain these additional proteins from the yeast genome database. The steps followed are described below. First, for each of the 20 protein components, in the core pathway namely Ste5, Ste11, Ste7, Ste20, Ste50, Fus3, Dig1, Dig2, Ste12, Sst2, Far1, Cdc24, Cdc42, Bem1, Ste2, Ste3, Ste4, Ste18, GPA1 and Tec1, we list all proteins that are known to interact with them physically. From this list we select only those proteins that are known to participate in the pheromone pathway reactions. Some proteins from this list overlap with the list of core component proteins. So they are not used in the model. So finally the list thus obtained (Table 6.5) contains 37 new proteins, which we will add to the pathway. We take these 37 additional proteins and add them to our network structure in the following manner. For each protein i which has j as a neighboring protein, we make i participate in all the reactions in which j is a reactant. In terms of our model, i becomes a preplace to all the post-transitions of j. Table 6.6 lists transitions and their preplaces, in other words, the reactants responsible for each reaction.

After adding the additional proteins we add *regulatory edges* (colored blue) in Figure 6.1 in the network to control the order in which transitions may fire in the network. We define *regulatory edges* as bidirectional egdes of weight one between a place and a transition which makes sure that the transition cannot fire until that place has at least one token. Also, bidirectionality ensures that the token content of the place is not affected by the firing of the transition. We illustrate this with the help of Figure 6.1. In Figure 6.1, reaction T_1 produces compound P_1, which participates in reaction T_2. Protein P_0 participates in reaction T_2 which in turn produces P_2. In the figure the bidirectional edge (blue edge) between P_1 and T_2 is a *regulatory edge* that makes sure that T_2 will not fire until P_1 is produced by T_1 irrespective of the amount of P_0 present.

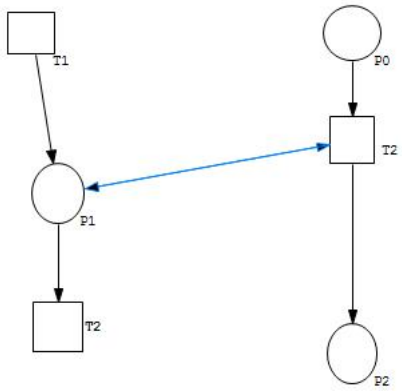

Figure 6.1: Example of Regulatory edge

Figure 6.2 illustrates the full structure of our representation of the pheromone pathway. In the structure, all places starting with p refer to the core protein components of the pathway listed in Table 6.2 and all places starting with a refer to the newly added proteins listed in Table 6.5. The places marked red represent the weighted sum listed in Table 6.7. All transitions starting with t are the transitions listed in Table 6.4. All logical nodes (connector which allow a node to participate in multiple activities in different areas) are marked in grey. The edges marked in blue are the regulatory edges introduced in the structure.

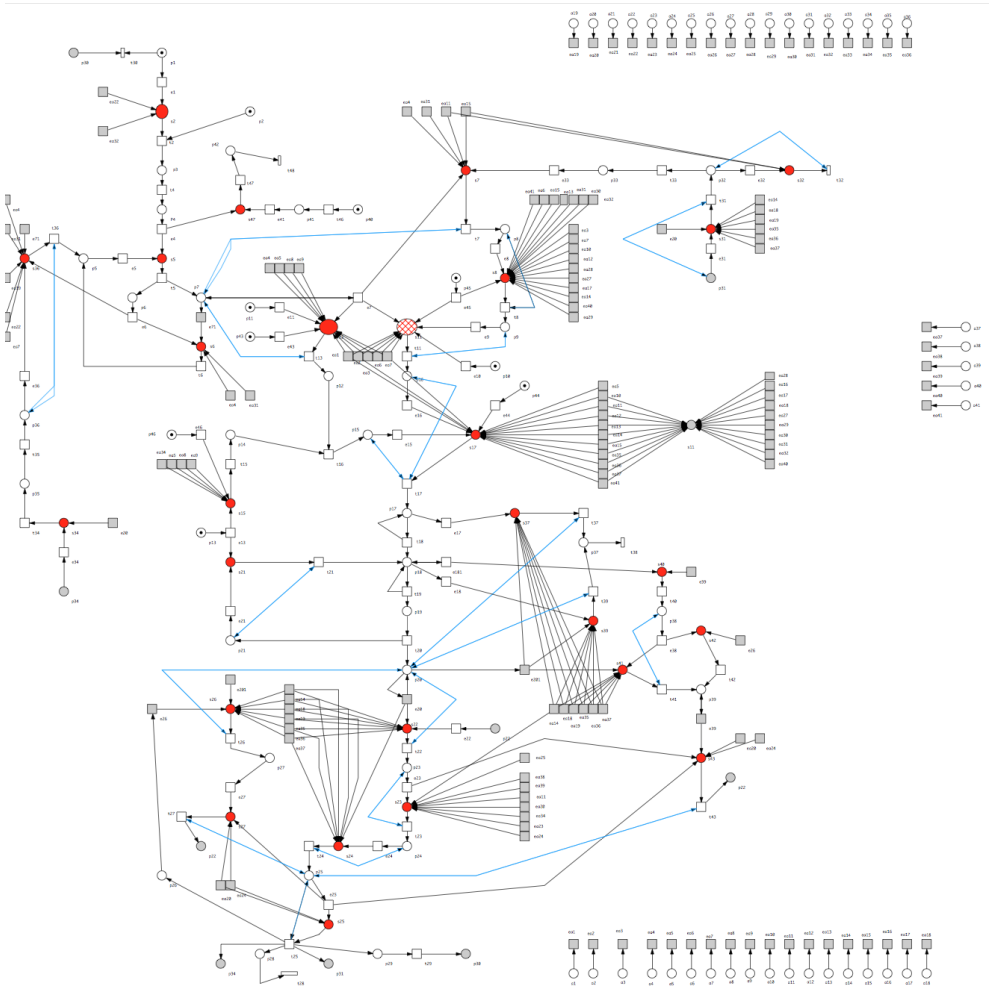

Figure 6.2: Full Petri net representation of the pathway

Table 6.1: Places of the model

Symbol	Place Name	Biological species
p1	alpha-factor	pheromone released by an MATα cell in the surroundings
p2	Ste2_receptor	mating pheromone receptor of the modelled MATa cell
p3	receptor_factor_complex	complex consisting of the α factor and the Ste2 receptor
p4	receptor_complex	the above named complex is activated by a conformation change
p5	trimer_bound_to_receptor	heterotrimeric G protein, which is coupled to the Ste2 receptor
p6	G_alpha_GTP	dissociated Gα subunit (exchange of GDP to GTP in this monomer)
p7	G_beta_gamma_dimer	G-protein G$\beta\gamma$ subunits in a dimer form
p8	Cdc24	Cdc24, i.e., guanine nucleotide exchange factor of Cdc42
p9	Cdc42(at_pm)	Cdc42 located at the plasma membrane
p10	Ste20	protein kinase Ste20
p11	Ste5(scaffold)	Ste5, acting as a scaffold protein
p12	Ste5/Ste11	protein complex consisting of ste7 and Fus3
p13	Fus3	MAP kinase Fus3
p14	Ste7/Fus3	protein complex consisting of Ste7 and Fus3
p15	MAPK_complex	MAPK complex consisting of Ste5,Ste11,Ste7 and Fus3
p16	Ste20_at_pm	Ste20 located at the plasma membrane, i.e., near the MAPK complex
p17	complex2	as complex1, but Ste11 is activated additionally
p18	complex3	as complex2, but Ste7 is activated additionally
p19	complex4	as complex3, but Fus3 is activated additionally
p20	Fus3PP	dissociated Fus3 in the activated form
p21	complex_without_Fus3	as complex4, but without Fus3
p22	repr_complex	complex containing Ste12 repressed by Fus3 or Kss1 and Dig1/Dig2
p23	Dig1/Dig2	Ste12 inhibitors, i.e., cofactors for the repression
P24	free_Ste12	Ste12 released out of the repression complex
p25	Ste12	activated transcription factor Ste12
p26	Msg5	phosphatase Msg5 being able to deactivate Fus3 or Kss1
p27	Fus3_dephos	deactivated Fus3
p28	other genes	pheromone regulated genes encoding mating related cell responses
p29	Bar1_in_nucleus	synthesised protease Bar1 located in the nucleus
p30	Bar1	Bar1 secreted in the cell environment
p31	inactive_Far1	synthesised Far1 located in the nucleus in an inactive form
p32	Far1	Far1 activated by phosphorylation
p33	Far1_in_cytosol	active Far1 located in the cytosol

Table 6.2: Places of the model (continued)

p34	Sst2_in_nucleus	synthesised Sst2 located in the nucleus in an inactive form
p35	phos_Sst2	Sst2 activated by phosphorylation
p36	Sst2	active Sst2 located in the cytosol
p37	inactive_component	complex labelled for degradation by phosphorylation
p38	phos_Kss1	MAP kinase Kss1 activated by phosphorylation
p39	unphos_Kss1	inactive Kss1
p40	Akr1	protein Akr1 located at plasma membrane
p41	Yck1/Yck2_at_pm	kinases Yck1/Yck2 being able to label the Ste2 for degradation
p42	inactive_receptor	receptor labelled for ubiquitination and endocytosis
p43	Ste11	protein kinase Ste11
p44	Ste50	protein kinase Ste50
p45	Bem1	protein Bem1
p46	Ste7	protein kinase Ste7

Table 6.3: Transitions of the model

Symbol	Transition Name	Biological Event
t1	MATalpha_cell(surroundings)	MATα cell secretes its mating pheromone
t2	binding_factor_to_receptor	α-factor binds to the Ste2 receptor
t3	receptor_synthesis	synthesis of the cell surface Sst2
t4	receptor_conformation_change	conformation change of the receptor
t5	division(in_alpha_subunit:GDP)→GTP	dissociation of the Gα subunit of the G-protein
t6	hydrolysis_GTP→GDP	hydrolysis reassociates Gα with G$\beta\gamma$
t7	interact_through_Far1	G$\beta\gamma$ interacts Far1 transmitted with Cdc24
t8	Cdc42:GDP→GTP	Cdc24 activates Cdc42
t9	active_Cdc42_constitutive_at_pm	constitute active Cdc42 attending the processes
t10	Ste20_input	source of Ste20
t11	Ste20_activated	Cdc42 at plasma membrane and Bem1 activates Ste20
t12	Ste5_input	source of Ste5
t13	Ste5_binds_Ste11	Ste5 binds Ste11
t14	Fus3_synth	synthesis of kinase Fus3
t15	Fus3_binds_Ste7	Ste7 binds Fus3
t16	complex-formation	Ste5/Ste11 binds Ste7/Fus3
t17	Ste20_phos_Ste11	phosphorylation of Ste11 by Ste20
t18	Ste11_phos_Ste7	phosphorylation of Ste7 by Ste11
t19	Ste7_phos_Fus3	phosphorylation of Fus3 by Ste7
t20	Fus3PP-release	release of activation Fus3 out of the MAPK complex
t21	binding_free_Fus3	remaining MAPK complex binds Fus3
t22	Ste12_inhibit_phos	phosphorylation of Ste12 inhibitors Dig1/Dig2 by Fus3PP
t23	Ste12-release	release of Ste12 out of the repression complex
t24	Ste12_phos	phosphorylation of Ste12 by Fus3PP
t25	transcr_activation	transcription activation of pheromone regulated genes
t26	Fus3PP_dephos	dephosphorylation of Fus3PP by Msg5
t27	repression_through_Fus3	Ste12 repression through inactive Fus3 and Dig1/Dig2
t28	cell_fusion	processes leading to the fusion of the two haploid cells
t29	transport_out_of_cell	Bar1 transport into the cell environment
t30	factor_destruction	Bar1 transmitted destruction of the α-factor
t31	Far1_phos	phosphorylation of Far1 by Fus3PP
t32	cell_cycle_arrest in_G1	Far1 caused arrest in the cell cycle phase G1

Table 6.4: Transitions of the model (continued)

t33	transport_out_of_cell	Bar1 transport into the cell environment
t34	Sst2_phos	phosphorylation of Far1 by Fus3PP
t35	transport_out_of_nucleus	Sst2 transport out of the nucleus
t36	accelerated_hydr_GTP→GDP	accelerated hydrolysis reassociates the G-protein
t37	Ste11_neg_phos	Fus3PP labels the MAPK complex at Ste11 for degradation
t38	degradation	degradation of the MAPK complex
t39	Ste7_neg_phos	Fus3PP labels the MAPK complex at Ste7 for degradation
t40	Ste7_phos_Kss1	phosphorylation of Kss1 by Ste7
t41	accelerated-dephos-Kss1	deactivation of phosphorylation Kss1 by Fus3PP
t42	Kss1_dephos	dephosphorylation of phosphorylated Kss1 by Msg5
t43	repression_through_Kss1	Ste12 repression through inactive Kss1 by Msg5
t44	tech_input	techinal:the repressed Ste12 complex assumed to be present
t45	Akr_synthesis	synthesis of Akr1
t46	Akr1_binds_Yck1/Yck2	Akr1 binds Yck1/Yck2
t47	receptor_phos	labelling of Ste2 for degradation
t48	ubiquit_endocytosis	ubiquitination and endocytosis of the receptor

Table 6.5: Additional Interacting Proteins

Symbol	Protein Name	Neighboring Components
a1	CBK1	STE5,STE20,STE50
a2	PTC1	STE5,STE20
a3	CLA4	STE11,CDC24,CDC42,BEM1
a4	DSE1	STE11,STE4
a5	HOG1	STE11,STE7,STE50
a6	PBS2	STE11,BEM1
a7	SHO1	STE11,STE20,STE50,SST2,CDC24
a8	SPA2	STE11,STE7
a9	SPH1	STE11,STE7
a10	RGA2	STE20,CDC24,CDC42,BEM1
a11	CLN2	STE20,DIG1,DIG2,FAR1
a12	ENT2	CDC24,STE20
a13	EXO84	STE20,BEM1
a14	BOI1	STE20,FUS3,DIG1,DIG2,CDC24,CDC42,BEM1
a15	CDC28	STE20,FAR1,BEM1
a16	GIC1	STE50,CDC42
a17	GIC2	STE50,CDC24,CDC42
a18	BN1	FUS3,CDC42
a19	MPT5	FUS3,SST2
a20	KDX1	TEC1,DIG1,DIG2,STE12
a21	KSS1	STE5,STE11,STE7
a22	WHI3	TEC1,SST2,STE2
a23	BZZ1	DIG1,DIG2
a24	HMLALPHA1	DIG1,DIG2,STE12
a25	HYM1	DIG1,DIG2
a26	YCK2	DIG2,STE3
a27	RSR1	CDC42,BEM1,CDC24
a28	SEC15	CDC24,BEM1
a29	EXO70	CDC42,BEM1
a30	SEC3	CDC42,BEM1
a31	RHO1	BEM1,STE4
a32	SEC6	BEM1,STE2
a33	AKR1	BEM1,STE2
a34	DIB1	STE7,DIG1
a35	YHR131C	STE20,FUS3
a36	BDF2	STE20,FUS3
a37	SAS10	STE20,FUS3
a38	RBS1	DIG1,DIG2
a39	YJR003C	DIG1,DIG2
a40	AXL2	CDC24,CDC42,BEM1
a41	BEM4	STE20,CDC24,CDC42

Table 6.6: Transitions and pre-places

Transitions	Meaning	Pre-places
t2	α-factor binds to the Ste2 receptor	s2 and p2
t4	conformation change of the receptor	p3
t5	dissociation of the Gα subunit of the G-protein	s5
t6	hydrolysis reassociates Gα with G$\beta\gamma$	s6
t7	G$\beta\gamma$ interacts Far1 transmitted with Cdc24	s7
t8	hydrolysis reassociates Gα with G$\beta\gamma$	p8 and s8
t11	Cdc42 at plasma membrane,Ste20 and Bem1 activates Ste20	p9,p10 and s11
t13	Ste5 binds Ste11	p11,p43 and s13
t15	Ste7 binds Fus3	p46 and s15
t16	Ste5/Ste11 binds Ste7/Fus3	p12 and p14
t17	phosphorylation of Ste11 by Ste20	p15,p16,p44 and s17
t18	phosphorylation of Ste7 by Ste11	p17
t19	phosphorylation of Fus3 by Ste7	p18
t20	release of activation Fus3 out of the MAPK complex	p19
t21	remaining MAPK complex binds Fus3	p21 and s21
t22	phosphorylation of Ste12 inhibitors Dig1/Dig2 by Fus3PP	s22
t23	release of Ste12 out of the repression complex	p23 and s23
t24	phosphorylation of Ste12 by Fus3PP	p24 and s24
t25	transcription activation of pheromone regulated genes	p25 and s25
t26	dephosphorylation of Fus3PP by Msg5	s26
t27	Ste12 repression through inactive Fus3 and Dig1/Dig2	p25 and s27
t28	processes leading to the fusion of the two haploid cells	p28
t29	Bar1 transport into the cell environment	p29
t30	Bar1 transmitted destruction of the α-factor	p1 and p30
t31	phosphorylation of Far1 by Fus3PP	s31
t32	Far1 caused arrest in the cell cycle phase G1	s32
t33	Bar1 transport into the cell environment	p32
t34	phosphorylation of Far1 by Fus3PP	s34
t35	Sst2 transport out of the nucleus	p35
t36	accelerated hydrolysis reassociates the G-protein	s36
t37	Fus3PP labels the MAPK complex at Ste11 for degradation	p17 and s37
t38	degradation of the MAPK complex	p37
t39	Fus3PP labels the MAPK complex at Ste11 for degradation	p18 and s39
t40	phosphorylation of Kss1 by Ste7	p18 and s40
t41	deactivation of phosphorylation Kss1 by Fus3PP	s41
t42	dephosphorylation of phosphorylated Kss1 by Msg5	s42
t43	Ste12 repression through inactive Kss1 by Msg5	p25 and s43
t46	Akr1 binds Yck1/Yck2	p40
t47	labelling of Ste2 for degradation	s47
t48	ubiquitination and endocytosis of the receptor	p42

Table 6.7: Description of weighted components

Name	Constituents
s2	p1,a22,a26,a32,a33
s5	p4,p5
s6	p6,p7
s7	p7,p33,a11,a15
s8	p45,a3,a7,a10,a12,a28,a27,a17,a14,a40,a41,a6,a15,a13,a31,a30,a32
s11	p7,p45,a1,a2,a3,a6,a7,a5,a10,a11,a12,a13,a14,a15, a35,a36,a37,a41,a16,a17,a18,a27,a29,a30,a31,a32,a40
s13	p7,a1,a2,a3,a6,a7,a4,a5,a8,a9,a21
s15	p13,a21,a34,a5,a8,a9
s17	a1,a2,a3,a6,a7,a5,a10,a11,a12,a13,a14,a15,a35,a36,a37,a41
s21	p13
s22	p20,p22,a14,a10,a19,a35,a36,a37
s23	a14,a25,a26,a38,a39,a11,a20,a21,a23,a24
s24	p20,a14,a10,a19,a35,a36,a37
s26	p26,p20,a14,a10,a19,a35,a36,a37
s27	p27,s20,s24
s31	p31,p20,a14,a18,a19,a21,a22,a7
s32	p32,a11,a15
s34	p20,p34
s36	p7,a19,a21,a22,a7
s37	p20,a14,a18,a19,a35,a36,a37
s39	p20,a14,a18,a19,a35,a36,a37
s40	p39
s41	p20,p38,a14,a18,a19,a35,a36,a37
s42	p26,p38
s47	p41,p4

Chapter 7

Experiment and Results

7.1 Experimental Setup

The Petri net model as described in the previous chapter can be visualized using the Snoopy 2.0 tool [2]. Using the export feature offered by Snoopy, we converted the graphical rerpresentation of the model into ANDL (Abstract Net Definition Language) format [11]. It is basically a textual representation of the network. The places in the network are represented as variables. The variables are initialized with numbers indicating the initial token content of the corresponding places. For instance, $V_i = 5$ in the Petri net means that there is a place named V_i with 5 tokens in it. Each transition is described as result of the changes in its preplaces and postplaces. For instance,

$$T_1 : : [p_2 + 34] \& [p_3 + 23] \& [p_1 - 10]$$

means transition T_1 has place p_1 as its *preplace* and p_2 and p_3 as its *postplaces*. For T_1 to fire, p_1 must have at least 10 tokens. When T_1 fires, 10 tokens are removed from p_1 and 34 and 23 tokens are placed into p_2 and p_3, respectively.

Using the ANDL description, we developed a Java program that generates ran-

dom networks for the model. In the program the places are represented by an array of variables. Each transition is represented by an object of a Rule class. Once all the transitions have been created, they are stored in a list called *rules*. As explained in Chapter 5, due to the absence of real world data about the kd values for the different reactions in the pathway, we generate all the edge weights in our model randomly. The range of values for the edge weights used in our experiments is between 1 and 100 (extremities included). The places representing the components α-factor, Ste2-receptor, Ste20, Ste5, Fus3, Akr1, Ste11, Ste7, Ste50 and Bem1 were provided with initial concentration values. Let ψ represent the set of these 10 core component proteins. All places representing the additional components were also provided with initial concentration values. Let λ represent the set of all 41 additional protein components in our model. From here on, the above mentioned places will be referred to as initial-places. For a given value of concentration of all the proteins in sets ψ and λ, the network is simulated. For each transition in the list *rules*, it is checked whether the transition is enabled or not. If yes, it is fired. After a single pass through the entire list, it is checked whether the transition producing Ste12 has fired or not. If yes, then the pathway has responded successfully and the resultant concentration values of the different proteins are recorded. If not, the entire list is traversed repeatedly. A counter is kept to avoid an infinite loop. After each pass through the list *rules*, the counter is incremented. However, if after a pass through the entire list, no transition fires, then the network has reached dead state. So the counter is not incremented and the execution is terminated. Tthe execution is also terminated if the counter exceeds a pre-fixed value.

7.2 Simulating the Network

7.2.1 Simulation Experiments

To simulate the pathway, we carry out three different experiments, each of which is discussed below. For the yeast pheromone pathway, apart from the structure of the pathway, exact kd values for each reaction are not known. From the literature, it can be seen that some experiments do provide possible kd values for some reactions. However, such values cannot be used in a generic way because they are specific to particular experiments. We have assumed that the value of kd for each reaction lies withing the range $\{1, 2, \ldots, 100\}$ [9]. In absence of real life data, we generate the kd value for each reaction randomly within the range $\{1, 2, \ldots, 100\}$ i.e. we assign weights to the different edges in the network structure randomly from $\{1, 2, \ldots, 100\}$. The values allowed for each edge are discrete as Petri nets do not allow interchange of fractional tokens. For each experiment, the range of values of concentration allowed for the proteins in set ψ is $\{1, 2, \ldots, 100\}$ (since Petri nets only allow integer number of tokens to be exchanged). The range of values for proteins in set λ vary in each experiment. Also in the simulation, values of all elements in each set ψ or λ change together. That is, when one protein in set ψ has a concentration value of 10, all the other proteins in ψ are also given the same value. The same is done for λ. In the rest of the thesis when we say "value for ψ" we mean the value of the initial concentration of the proteins in ψ; similarly "value for λ" means the value of the initial concentration of the proteins in λ. In biological context, when we are simulating a network with its randomly generatd edge weights, the edge weights represent different conditions the cell is subjected to while it tries to respond to pheromone.

1. **Experiment 1**: The range of values of initial concentration for the proteins in λ is set to be between 1 and 10. For any network, for a given value of initial

concentration of proteins in ψ and for a given value of initial concentration of proteins in λ, it is checked if the transition that produces protein Ste12 has fired or not. Production of Ste12 indicates that mating will happen and we call such an output a *positive* response. On the other hand, if Ste12 is not produced, it indicates that the cell will not mate and we call this output a *negative* response by the network. Based on the range of values allowed for both ψ and λ there are in total 1000 combinations (100 values from the set ψ times 10 values from the set λ) of initial concentration values of proteins in ψ and in λ. We generate 13962 networks and check for the response of the pathway in each of them. The objective of Experiment 1 is to identify conditions (i.e., different edge weights) under which the cell responds positively to the pheromone pathway.

2. **Experiment 2**: We take the 13962 networks generated in Experiment 1, and isolate the networks based on their responses. The ones which gave negative response are put in set *neg*, while the ones with positive response are put in set *pos*. We again run the simulation on each of the networks in *neg* but now we set the range of concentration of the proteins in λ to be $\{11, 12, \ldots, 20\}$. For each network we check, which combination of values for elements in ψ and λ yield positive response. If the response is positive, the amount of Ste12 produced, and the concentrations of the core and additional proteins are recorded. The objective of Experiment 2 is to test if the cell can overcome the conditions which made it respond negatively in Experiment 1, by using more concentration of proteins in the set λ.

3. **Experiment 3**: To have a better understanding of which proteins in λ play a more significant role in the pathway, we divide the set λ into sets σ and ς such that $\lambda = \varsigma \cup \sigma$ and $\sigma \cap \varsigma = \emptyset$. The proteins CBK1, PTC1, DSE1, SPA2, SPH1,

MPT5, KDX1, HYM1, DIB1, YHR131c, BDF2, SAS10, RBS1 and YJR003c from λ are placed in σ. The rest are placed in ς. We hypothesise that the proteins in s contribute more to the pheromone pathway than the ones in ς and hence consider them to be more significant in their role in the pathway [9]. To simulate this we set the range of values for the concentration of those proteins to be $\{11, 12, \ldots, 20\}$. For the proteins in ς, the range is set to be $\{1, 2, \ldots, 10\}$. For all networks in set *pos* from Experiment 2, we run the simulation and look for positive responses. So the objective of Experiment 3 is to test the above mentioned hypothesis.

7.2.2 Results

1. **Result of Experiment 1**: From the 13962 generated networks, 11816 networks gave negative response. That is, for all 1000 combinations of values of initial concentrations of the proteins in ψ and λ, in each of the above mentioned 11816 networks, the transition which results in the production of protein Ste12 did not fire. The remaining 2146 networks gave positive a response. The output of networks giving positive response are of 2 types.

   (a) A network starts giving positive response when the value for ψ is $\geq$ some value $x \in \{1, 2, \ldots, 100\}$ and the value for $\lambda \geq 1$. For instance, if a network starts giving positive response when the value for ψ is 74 and the value for λ is 1, it means that, for this particular network with its set of edge weights (hence worth called *configuration* of the network), as soon the as value for ψ exceeds 74, it will give a positive response irrespective of the concentrations of the proteins in λ.

   (b) A network starts giving positive response when the value for ψ is $\geq$ some

$x \in \{1, 2, \ldots, 100\}$ and the value for λ exceeds some value $y \in \{2, \ldots, 10\}$. For instance, if a network starts giving a positive response when the value for ψ is 74 and the value for λ is 5, that means, for this particular network with its corresponding configuration to respond positively, it is not sufficient that the values for ψ become 74; The value for λ also needs to exceed value 5.

2. **Result of Experiment 2**: Out of the 11816 networks, 10840 networks still gave negative response. The remaining 976 networks responded positively. That is, out of these 976 networks, each one started giving positive responses when the value for ψ is $\geq$ some value $x \in \{1, 2, \ldots, 100\}$ and the value for λ exceeds some value $y \in \{12, \ldots, 20\}$. That is, by increasing the initial concentration level of the proteins in λ these networks changed their response from negative in Experiment 1 to positive in this experiment. So this means for these 976 networks, the additional proteins in λ play a significant role in deciding how the network responds to the pathway. Changing a prior negative response to a positive one indicates that these proteins might potentially be able to compensate for the lack of some of the core protein components in the pathway if present in sufficient amount.

3. **Result of Experiment 3**: Based on the output of each network, the networks can be classified into 3 categories.

   (a) CS: This class CS (Class Same) represents those networks that gave positive responses in both Experiments 2 and 3 using the same combination of values for its proteins. That is, if a network gave a positive response in Experiment 2 with values x as the value for ψ and y as the value for λ, it gives positive response in Experiment 3 as well with the same combination

of values; x as the value for ψ and y as the value for σ. For instance, if a network in CS gave a positive response in Experiment 2 when the value for ψ exceeded 74 and the value for λ exceeded 5, it gives a positive response in Experiment 3 when the value for ψ exceeded 74 and the value for σ exceeded 5. Out of the 976 networks (from *pos*) used for this experiment 352 of them were placed in class *CS* because of their output.

(b) CD: This class CD (Class Different) represents those networks which gave positive responses in both Experiments 2 and 3 but using the different combination of values for its proteins. For instance if in Experiment 2, the network had initial concentration values x for the proteins in ψ and y for those in λ, in Experiment 3 it has x as initial concentration value for proteins in ψ and z for those in σ where $y \neq z$. Such a network is placed in class *CD*. Out of the 976 networks, 76 of them were placed in class *CD*.

(c) CN: This class CN (Class Negative) represents those networks which gave positive responses in Experiment 2 but now give negative response in Experiment 3. 548 networks from set *pos* gave negative response and were placed in class *CN*.

The distribution of the 3 classes in the set *pos* is shown in Figure 7.1.

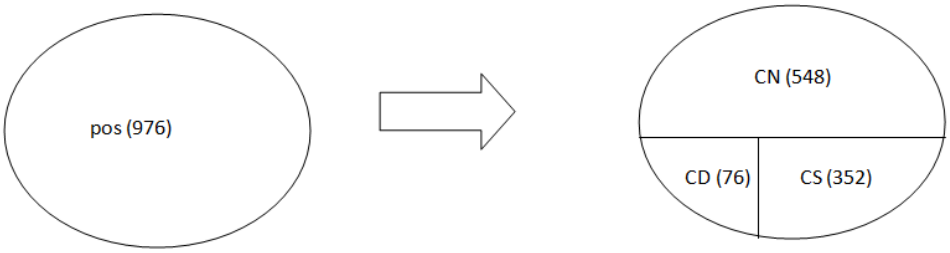

Figure 7.1: Distribution of CS, CD and CN

7.2.3 Interpretation of Results

1. Experiment 1: Networks which give a positive response when the value for ψ is $\geq$ some value $x \in \{1, 2, \ldots, 100\}$ and the value for $\lambda \geq 1$ indicate that for these networks with their corresponding set of edge weights, the additional proteins in λ play no significant role in controlling their responses. The response is based solely on the initial concentration of the core component proteins in ψ. Networks which start giving a positive response when the value for ψ is $\geq$ some value $x \in \{1, 2, \ldots, 100\}$ and the value for $\lambda \geq$ some y where $y \in \{2, \ldots, 10\}$ indicate that for these networks with their given configuration, depend on the additional proteins in λ for modulating their response to the pheromone pathway. That is, for these networks it is the additional proteins in λ which makes the response positive when the value for ψ is not sufficient. In a biological context, such networks show that under those conditions the yeast cell uses the proteins in λ to facilitate mating. Networks with negative responses indicate the conditions under which a cell will not mate for any combination of initial concentrations of its different proteins.

2. Experiment 2: The 976 networks which start responding positively indicate that the amount of concentration for proteins in ψ or λ allowed in Experiment 1 was not sufficient for them to give a positive response. So the cell compensated by using more amounts of those additional proteins in λ to facilitate mating. The increase of the range of allowable values for λ by us, simulate the cell using more concentration of proteins than what it was using in Experiment 1. These networks support our hypothesis that the cell probably uses one or more additional proteins to respond favorably to the pheormone pathway when it is unable to produce a positive response using just the core component proteins.

3. Experiment 3: Networks in class *CS* tell us that for these networks with their corresponding configurations the set of proteins in σ play a more significant role in the pheromone pathway than the rest of the proteins in ς. This indicates that a particular network does not require higher concentrations of all the proteins in λ to change its response from negative to positive. The proteins in σ are alone capable of doing so. So these networks represent conditions under which the cell rely more on the proteins in σ than those in ς to facilitate a change in response from negative to positive.

7.3 Analysis of Experiments

7.3.1 Development of Decision Trees

In order to identify reasons that might determine whether a network responds positively or negatively we use *decision trees* to identify important attributes in the network. "Decision tree learning is a method for approximating discrete-valued target functions, in which the learned function is represented by a decision tree. Learned trees can also be re-represented as sets of if-then rules to improve human readability" [13]."Decision trees classify instances by sorting them down the tree from the root to some leaf node, which provides the classification of the instance. Each node in the tree specifies a test of some attribute of the instance, and each branched descending from that node corresponds to one of the possible values for this attribute" [13]. We use Weka 3.6 (Waikato Environment for Knowledge Analysis) [4] software for this purpose. We consider each edge in the network as its different attributes. Figue 7.2 shows an example of a small portion of the decision tree generated by our data. Each node or attribute is actually an edge weight from a place to a transition in the network or vice versa. In the tree the values of the weights for edges from places to

```
a31TOea31 <= -21
|   a4TOea4 <= -21
|   |   p3TOt4 <= -29
|   |   |   t2TOp3 <= 34: n (1825.0/9.0)
|   |   |   t2TOp3 > 34
|   |   |   |   e4TOs5 <= 9: n (330.0)
|   |   |   |   e4TOs5 > 9
|   |   |   |   |   p4TOe4 <= -31
|   |   |   |   |   |   t4TOp4 <= 29: n (736.0/2.0)
|   |   |   |   |   |   t4TOp4 > 29
|   |   |   |   |   |   |   p3TOt4 <= -78: n (506.0/3.0)
|   |   |   |   |   |   |   p3TOt4 > -78
|   |   |   |   |   |   |   |   s8TOt8 <= -86: n (171.0)
|   |   |   |   |   |   |   |   s8TOt8 > -86
|   |   |   |   |   |   |   |   |   e41TOs47 <= 86
|   |   |   |   |   |   |   |   |   |   t21TOp18 <= 15: n (111.0)
|   |   |   |   |   |   |   |   |   |   t21TOp18 > 15
```

Figure 7.2: A sample Decision Tree

transitions are written as negative integers. They should be read as positive values (this is dependent of how attributes were provided to the decision tree learner). So the node a31Toea31 is the value of the edge weight between place a31 and transition ea31 in the network and while checking its value should be considered compared with 21 instead of -21. The very first node at the top is the root of the tree. Each line represents a level in the tree. The straight lines continuing down a node ends at a node with the same attribute but with different checking condition. We start traversing the tree from the root, i.e., the very first node in the tree. If the value of that attribute satisfies that condition we go down that branch i.e. to the next level (next line in the figure). Else we follow the straight line from that node, to its alternate condition. The straight line below node a31TOea31 goes down to a node, where it is checked if attribute a31TOea31 is greater than 21. So at every node, the corresponding condition is checked and a branch is selected accordingly. After tranversing the tree from the root, at the fourth level the value of attribute t2TOp3 is checked. If the value is $\leq$ 34, then a leaf is reached. That is the decision tree

concludes by looking at the different attribute values in the trees from root to node t2TOp3, that the network will respond negatively. The tree also mentions that how many networks it has correctly and incorrectly classified at that leaf. According to this tree, it has correctly classified 1825 networks as having negative responses and 9 networks incorrectly. at level 4. In this manner the entire tree can be read to predict the response of a network.

The performance of any learning method depends on well it classifies a dataset that it has not encountered before. *Cross validation* is a method which allows to improve the performance of the learner (in this case the decision tree). The idea is to divide up the entire dataset randomly into a training set (dataset used for trainng the learner) and a testing set (dataset used for testing the learner). The performance of the learner is then evaluated on this testing set. *10-fold cross validation* is a variation of cross validation where the data set is divided into 10 sets. Each time one of the 10 sets is used as a testing set and the remaining 9 become the training set. The whole process is repeated 10 times [1].

1. **Experiment 4**: We take the output of Experiment 1 in Section 7.2.1 and divide the output into two classes P and N. Networks that give postive responses are put in class P while the ones with negative response are put in class N. For each network, each of its edge weights is listed as an attribute for that network followed by its class P or N. Once the list is completed for all 13962 networks, it is given to the J48 decision tree program implemented by Weka 3.6 [4] as an input. A 10-fold cross validation is carried out to get a better estimate of the performance of the decision tree. We compare the different nodes at each level of a decision tree across all the ten trees generated by Weka. This is done to look for attributes which get tested most often (in more than 5 out of 10 trees) at the same level. We look at the first 4 levels starting from the root of each

tree.

2. **Experiment 5**: We take the output of Experiment 2 in Section 7.2.1 and divide the output into two classes P and N based on their response as mentioned in Experiment 4. We create a dataset by listing each edge weight of each network followed by their corresponding classes. The dataset if fed to the J48 tree in Weka and 10-fold cross validation is carried out. We compare the nodes at each level across all the 10 trees for the first 4 levels for look for common attributes that get tested often (in more than 5 out of 10 trees) at the same level across all trees.

3. **Experiment 6**: We divide the output of Experiment 3 in Section 7.2.1 into 3 classes *CS*, *CD* and *CN*, based on their individual responses. These 3 classes are the same ones that we described in Experiment 3 of Section 7.2.1. Once all the networks have been classified, a data set describing the attribute and class of each network is created as mentioned above. The data set is fed to J48 of Weka and a 10-fold cross validation is carried out. We compare the nodes at each level across all the 10 trees for the first 4 levels for look for common attributes that get tested often (in more than 5 out of 10 trees) at the same level across all trees.

7.3.2 Results

Figure 7.4 gives the summary of the classification result for **Experiment 4**. From the summary we can see that the tree correctly classifies a network approximately 78% of the time. The confusion matrix tells us that the decision tree has more efficiency ($10298/11816 \sim 87.15\%$) in correctly predicting a network with negative response than a network with positive response ($662/2146 \sim 30.84\%$). We identify the most

commonly compared nodes across all 10 trees generated for the first 4 levels. They are listed in Figure 7.5 and Figure 7.6. Each color represents the most common node compared in that level across all 10 trees. Figure 7.5 contains nodes common in the first 3 levels while Figure 7.6 shows nodes common in the fourth level of the trees. From these figures the nodes can be arranged in an increasing order of importance in the following manner. Higher level indicates more importance, more number also adds to the significance of a node. Fig 7.3 gives the color coding scheme followed in

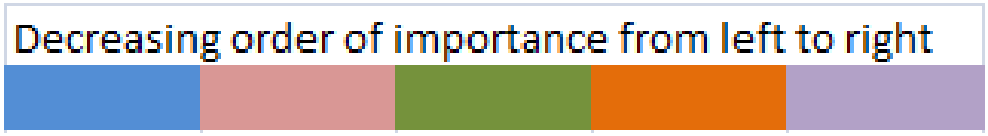

Figure 7.3: Color coding scheme depicting degree of importance

the tables. Table 7.1 shows this. Table 7.1 tells us which nodes are examined first by the decision tree while classifying a given network.

Table 7.1: Hierarchy of nodes in Experiment 4

Node	Biological Meaning	Value
a4Toea4	Minimum concentration of DSE1 required for it to participate in the pathway	7
t6TOp5	Amount of trimer bound to receptor due to hydrolysis of GTP	10
p3TOt4	Minimum concentration of receptor-factor complex required for conformation change of the receptor	
a31TOea31	Minimum concentration of RHO1 required for it to participate in the pathway	11
ea4TOs6	Edge weight between dummy transition for DSE1 and weighted compound s6	
t2TOp3	Amount of receptor-factor complex formed due to binding of α factor to receptor	13
t2TOp3	Amount of receptor-factor complex formed due to binding of α factor to receptor	50
e31TOs6	Edge weight between dummy transition for RHO1 and weighted compound s6	51

```
Correctly Classified Instances       10960               78.4988 %
Incorrectly Classified Instances      3002               21.5012 %
Kappa statistic                          0.1789
Mean absolute error                      0.216
Root mean squared error                  0.4553
Relative absolute error                 83.0051 %
Root relative squared error            126.2462 %
Total Number of Instances            13962

=== Detailed Accuracy By Class ===

                 TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.308     0.128      0.304     0.308     0.306      0.576    p
                 0.872     0.692      0.874     0.872     0.873      0.576    n
Weighted Avg.    0.785     0.605      0.786     0.785     0.786      0.576

=== Confusion Matrix ===

   a     b   <-- classified as
 662  1484 |     a = p
1518 10298 |     b = n
```

Figure 7.4: Summary of Experiment 4

	Level 1	Level 2		Level 3			
	Node1.1	Node 2.1	Node 2.2	Node 3.1	Node 3.2	Node 3.3	Node 3.4
Tree 1	a4TOea4, 7	p3TOt4, 32	t6TOp5, 9	a31TOea31, 11	t2TOp3, 12	e39TOs40, 4	ea4TOs6, 36
Tree 2	a4TOea4, 7	p3TOt4, 34	t6TOp5, 10	a31TOea31, 11	t2TOp3, 28	ea35TOs39, 89	ea4TOs6, 19
Tree 3	a4TOea4, 6	p3TOt4, 32	t6TOp5, 16	a31TOea31, 11	a31TOea31, 3	t16TOp15, 52	ea4TOs6, 19
Tree 4	a4TOea4, 7	p3TOt4, 32	t6TOp5, 10	a31TOea31, 11	t2TOp3, 13	a40TOea40, 41	ea4TOs6, 31
Tree 5	a4TOea4, 7	p3TOt4, 32	t6TOp5, 9	a31TOea31, 11	a31TOea31, 3	ea9TOs13, 94	ea4TOs6, 39
Tree 6	a4TOea4, 6	p3TOt4, 32	t6TOp5, 10	a31TOea31, 11	a31TOea31, 3	ea14TOs26, 13	ea4TOs6, 39
Tree 7	a4TOea4, 7	a31TOea31, 11	ea4TOs6, 39	p3TOt4, 36	t17TOp17, 9	t25TOp31, 8	a4TOea4, 3
Tree 8	a4TOea4, 7	a31TOea31, 11	t6TOp5, 9	p3TOt4, 36	ea31TOs6, 18	ea28TOs11, 89	ea4TOs6, 31
Tree 9	a4TOea4, 6	p3TOt4, 32	t6TOp5, 16	a31TOea31, 11	t2TOp3, 13	ea14TOs31, 95	ea4TOs6, 19
Tree10	a4TOea4, 6	p3TOt4, 32	t6TOp5, 10	a31TOea31, 11	t2TOp3, 12	ea14TOs26, 13	ea4TOs6, 17

Figure 7.5: Common nodes observed in Experiment4

Figure 7.7 gives the summary of the classification result for **Experiment 5**. From the summary we can see that the tree correctly classifies a network approximately 87% of the time. The confusion matrix tells us that the decision tree has an efficiency of $209/976 \sim 21.41\%$ in correctly predicting a positively responding network and an efficiency of $10072/10840 \sim 92.91\%$ in predicting a network with negative response. We identify the most commonly compared nodes across all 10 trees generated for the first 4 levels. They are listed in Figure 7.8 and Figure 7.9. Each color represents the most common node compared in that level across all 10 trees. Figure 7.8 contains nodes common in the first 3 levels while Figure 7.9 shows nodes common in the

	Level 4							
	Node 4.1	Node 4.2	Node 4.3	Node 4.4	Node 4.5	Node 4.6	Node 4.7	Node 4.8
Tree 1	t2TOp3, 50	ea31TOs6, 52	t17TOp17, 21	e1TOs2, 24	ea7TOs13, 13	ea17TOs11, 18	t17TOp17, 12	
Tree 2	t2TOp3, 50	ea31TOs6, 51	t2TOp3, 4	e1TOs2, 24	t36TOp5, 13	p34TOe34, 43	ea5TOs17, 91	t2TOp3, 7
Tree 3	t2TOp3, 50	ea31TOs6, 52	t2TOp3, 13	p7TOe71, 19	ea14TOs26, 92	a27TOea27, 48	p18TOt19, 96	
Tree 4	t2TOp3, 50	ea31TOs6, 51	s2TOt2, 10	a31TOea31, 3	ea30TOs8, 84	t4TOp4, 21	a4TOea4, 5	
Tree 5	t2TOp3, 50	ea31TOs6, 51	t2TOp3, 13	e71TOs36, 21	ea28TOs11, 89	t25TOp31, 8	a4TOea4, 3	
Tree 6	t2TOp3, 50	ea25TOs23, 96	t2TOp3, 13	ea16TOs11, 42	t2TOp3, 31	t37TOp37, 86	t25TOp31, 8	t17TOp17, 12
Tree 7	p2TOt2, 51	t2TOp3, 9	p17TOe17, 89	ea31TOs6, 18	e24TOs24, 88	ea2TOs13, 88	p18TOt19, 93	
Tree 8	t2TOp3, 50	t2TOp3, 29	p2TOt2, 9	a31TOea31, 4	e38TOs42, 92	t4TOp4, 21	s5TOt5, 77	
Tree 9	t2TOp3, 50	ea31TOs6, 51	e6TOs6, 5	e1TOs2, 24	a4TOea4, 2	a27TOea27, 48	a4TOea4, 3	
Tree10	t2TOp3, 50	t4TOp4, 96	e6TOs6, 5	e1TOs2, 24	p5TOe5, 47	t37TOp37, 86	t16TOp15, 7	s5TOt5, 77

Figure 7.6: Common nodes observed in Experiment4

fourth level of the trees. Like before we arrange the nodes in an increasing order of importance following the previously mentioned convention. Table 7.2 shows this. It tells us which nodes are examined first by the decision tree while classifying a given network. Figure 7.10 gives the summary of the classification result for

```
Correctly Classified Instances       10281               87.0091 %
Incorrectly Classified Instances      1535               12.9909 %
Kappa statistic                          0.1432
Mean absolute error                      0.1319
Root mean squared error                  0.3545
Relative absolute error                 87.0133 %
Root relative squared error            128.7696 %
Total Number of Instances            11816

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.214    0.071      0.214     0.214     0.214      0.547    p
                 0.929    0.786      0.929     0.929     0.929      0.547    n
Weighted Avg.    0.87     0.727      0.87      0.87      0.87       0.547

=== Confusion Matrix ===

   a     b   <-- classified as
 209   767 |     a = p
 768 10072 |     b = n
```

Figure 7.7: Summary of Experiment 5

Experiment 6. From the summary we can see that the tree correctly classifies a network approximately 94% of the time. The confusion matrix tells us about the effieciency of the decision tree in correctly classifying instances of CS, CN and CD. We identify the most commonly compared nodes across all 10 trees generated for the first 4 levels. For this particular experiment and dataset common nodes were found

	Level 1	Level 2		Level 3			
	Node1.1	Node 2.1	Node 2.2	Node 3.1	Node 3.2	Node 3.3	Node 3.4
Tree 1	a31TOea31, 21	a4TOea4, 21	s6TOt6, 22	p3TOt4, 29	s6TOt6, 46	ea31TOs6, 23	a31TOea31, 11
Tree 2	a31TOea31, 21	a4TOea4, 21	t18TOp18, 11	p3TOt4, 29	s6TOt6, 48	p18TOt19, 60	s6TOt6, 22
Tree 3	a31TOea31, 21	a4TOea4, 21	s6TOt6, 15	p3TOt4, 29	s6TOt6, 46	t18TOp18, 11	ea37TOs26, 87
Tree 4	a31TOea31, 21	a4TOea4, 21	s6TOt6, 22	p3TOt4, 27	t15TOp14, 2	ea31TOs6, 23	t20TOp21, 89
Tree 5	a31TOea31, 21	a4TOea4, 21	ea41TOs8, 2	p3TOt4, 29	s6TOt6, 46	s6TOt6, 15	
Tree 6	a31TOea31, 21	a4TOea4, 21	s6TOt6, 15	p3TOt4, 27	t6TOp5, 10	t18TOp18, 11	ea37TOs26, 87
Tree 7	a31TOea31, 21	a4TOea4, 21	t18TOp18, 11	p3TOt4, 27	p2TOt2, 98	p18TOt19, 60	s6TOt6, 22
Tree 8	a31TOea31, 21	a4TOea4, 21	t18TOp18, 11	p3TOt4, 27	s6TOt6, 46	p18TOt19, 60	ea31TOs6, 37
Tree 9	a31TOea31, 21	a4TOea4, 21	s6TOt6, 22	p3TOt4, 29	s6TOt6, 46	ea31TOs6, 22	a31TOea31, 12
Tree10	a31TOea31, 21	a4TOea4, 21	t18TOp18, 11	p3TOt4, 27	s6TOt6, 46	p18TOt19, 45	s6TOt6, 15

Figure 7.8: Common nodes observed in Experiment 5

	Level 4							
	Node 4.1	Node 4.2	Node 4.3	Node 4.4	Node 4.5	Node 4.6	Node 4.7	Node 4.8
Tree 1	t2TOp3, 34	t2TOp3, 28	ea4TOs6, 70	t17TOp17, 37	e8TOs8, 96	t18TOp18, 10	ea37TOs26, 10	p17TOe17, 89
Tree 2	t2TOp3, 34	t2TOp3, 27	ea4TOs6, 70	t17TOp17, 35	ea2TOs13, 57	ea31TOs6, 22	t15TOp14, 12	
Tree 3	t2TOp3, 34	t2TOp3, 29	ea4TOs6, 70	t17TOp17, 33	ea36TOs41, 53	ea31TOs6, 22	a31TOea31, 12	ea39TOs23, 81
Tree 4	t2TOp3, 34	t2TOp3, 27	s6TOt6, 46	p31TOe31, 85	p5TOe5, 14	a31TOea31, 11	e46TOs15, 93	
Tree 5	t2TOp3, 34	t2TOp3, 27	ea4TOs6, 70	t17TOp17, 17	ea31TOs6, 23	ea37TOs26, 87		
Tree 6	t2TOp3, 35	t2TOp3, 27	p14TOt16, 9	s6TOt6, 48	p18TOt19, 45	ea31TOs6, 23	ea35TOs39, 12	ea39TOs23, 81
Tree 7	t2TOp3, 34	t2TOp3, 27	s6TOt6, 48	ea11TOs17, 42	ea31TOs6, 23	t20TOp21, 91		
Tree 8	t2TOp3, 34	t2TOp3, 27	ea4TOs6, 70	t6TOp5, 4	t25TOp29, 67	s6TOt6, 7	t17TOp17, 5	
Tree 9	t2TOp3, 34	t2TOp3, 27	ea4TOs6, 70	t17TOp17, 35	p25TOe25, 6	t18TOp18, 11	t6TOp5, 6	a15TOea15, 94
Tree10	t2TOp3, 34	t2TOp3, 27	ea4TOs6, 70	a4TOea4, 3	ea36TOs31, 41	ea31TOs6, 23	t15TOp14, 12	

Figure 7.9: Common nodes observed in Experiment 5

across the trees for the first three levels only. They are listed in Figure 7.11. Each color represents the most common node compared in that level across all 10 trees. We arrange the nodes in an increasing order of importance following the previously mentioned convention. Table 7.3 shows this. It tells us which nodes are examined first by the decision tree while classifying a given network.

7.3.3 Interpretation of Results

The tables of most commonly tested attributes obtained from Experiment 4, 5 and 6 reveal that those are the nodes or attributes that drive the decision into classifying a network. So in the context of biological network, these attributes probably represent important conditions that regulate the cell response to pheromone.

Table 7.2: Hierarchy of nodes in Experiment 5

Node	Biological Meaning	Value
a31TOea31	Minimum concentration of RHO1 required for it to participate in the pathway	21
a4TOea4	Minimum concentration of DSE1 required to participate in the pathway	21
s6TOt6	kd value for the reaction:Hydrolysis of G-protein	22
p3TOt4	Minimum concentration of receptor-factor complex required for conformation change of the receptor	28
s6TOt6	kd value for the reaction:Hydrolysis of G-protein	46
t2TOp3	Amount of receptor-factor complex formed due to binding of α factor to receptor	34
t2TOp3	Amount of receptor-factor complex formed due to binding of α factor to receptor	27
ea4TOs6	Edge weight between dummy transition for DSE1 and weighted compound s6	
t17TOp17	Concentration of MAPK complex after Ste20 phosphorylates Ste11	
ea31TOs6	Edge weight between dummy transition for RHO1 and weighted compound s6	23

Table 7.3: Hierarchy of nodes in Experiment 6

Node	Biological Meaning	Value
a4TOea4	Minimum concentration of DSE1 required to participate in the pathway	21
s6TOt6	kd value for the reaction:Hydrolysis of G-protein	
ea4TOs6	Edge weight between dummy transition for DSE1 and weighted compound s6	

```
Correctly Classified Instances       10033               94.3394 %
Incorrectly Classified Instances       602                5.6606 %
Kappa statistic                          0.1017
Mean absolute error                      0.0431
Root mean squared error                  0.1909
Relative absolute error                 91.7961 %
Root relative squared error            124.8394 %
Total Number of Instances            10635

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.114     0.021       0.143    0.114      0.127      0.365    CS
                 0.975     0.87        0.968    0.975      0.971      0.404    CN
                 0.014     0.005       0.019    0.014      0.017      0.21     CD
Weighted Avg.    0.943     0.839       0.937    0.943      0.94       0.402

=== Confusion Matrix ===

   a    b    c   <-- classified as
  36  275    5 |    a = CS
 208 9996   46 |    b = CN
   8   60    1 |    c = CD
```

Figure 7.10: Summary of Experiment 6

	Level 1	Level 2		Level 3			
	Node1.1	Node 2.1	Node 2.2	Node 3.1	Node 3.2	Node 3.3	Node 3.4
Tree 1	a4TOea4, 21	t6TOp5, 14		ea20TOs25, 97	s6TOt6, 48		
Tree 2	a4TOea4, 21	ea4TOs6, 36		t37TOp37, 7	e5TOs5, 98		
Tree 3	a4TOea4, 21	p20TOe20, 85	s6TOt6, 48	p3TOt4, 17	p3TOt4, 5	ea4TOs6, 71	a31TOea31, 18
Tree 4	a4TOea4, 21	s6TOt6, 28		ea4TOs6, 36	a4TOea4, 3		
Tree 5	a4TOea4, 21	p20TOe20, 82	s6TOt6, 48	ea35TOs37, 97	p17TOt18, 98	t17TOp17, 17	
Tree 6	a4TOea4, 21	ea4TOs6, 30		e20TOs24, 8	e5TOs5, 98		
Tree 7	a4TOea4, 21	t6TOp5, 14		ea25TOs23, 6	s6TOt6, 48		
Tree 8	a4TOea4, 21	s6TOt6, 48		ea4TOs6, 54	a31TOea31, 18		
Tree 9	a4TOea4, 21	s6TOt6, 26		ea4TOs6, 30	a4TOea4, 3		
Tree10	a4TOea4, 21	s6TOt6, 43		ea4TOs6, 30	t17TOp17, 18		

Figure 7.11: Common nodes observed in Experiment 6

Chapter 8

Conclusions

8.1 Discussion of Results

The simulation experiments revealed 3 kinds of result. Experiment 1 tells us about the different conditions under which a cell will respond to a pheromone or not. Under some conditions a cell does not respond at all. If a cell responds positively, there are two possible methods for its response: either the response is solely dependent on the initial concentration of its core component proteins in ψ or the response is to some extent dependent on the concentration of the proteins in λ as well. Experiment 2 looks for possible changes that the cell might adopt so that it can mate in conditions which made it respond negatively in Experiment 1. This is simulated by allowing the cell to utilize more concentration of proteins in λ. The results reveal that the cell can overcome the detrimental effects of the conditions by using more concentration of additional proteins in λ. These 2 experiments provide evidence that employing more concentration of proteins might be one of the ways that the cell uses to adapt itself in inhibiting conditions to facilitate mating. Experiment 3 tries to look for specific proteins in λ that might be resonsible for allowing a cell to change it response

to pheromone from positive to negative. The results reveal that in some case the protein set σ is sufficient in regulating the response of the cell. In other cases, the requirements for the proteins in σ are more stringent. Results of Experiments 4, 5 and 6 reveal that there are certain conditions (edge weights) in the model that are more important in determining whether a cell will respond positively or not.

8.2 Future Work

As a follow up of this work, we would like probe more about the functionality of the proteins in set λ. In Experiment 3 we looked at the performance of a subset of proteins (σ) in λ. We plan to extend our simulation to individual proteins in the set σ. This can be done by isolating a particular protein and varying its available concentration in the simulations. There is also scope of improving the model on several aspects. In our experiments the conditions that the cell is subjected to, in the simulation i.e., the edge weights used in the model are generated randomly. We can add some constraints on how weights are assigned to these edges, so that simulation can mirror real-world phenomena more closely. Also, in our model the number of tokens exchanged during interaction of places and transitions are integers as ordinary Petri net allows only that. However, in real life, the kd value of reactions can hardly be expected to be integers all the time. So we would like to modify our model so that it can handle the exchange of fractional tokens among its nodes. Now in the pheromone pathway, evidence of negative feedback loop has been found, which has not been implemented in our model. So we would also like to explore some other variant of Petri net which allows implementation of such negative feedback loops. Finally we would like to extend our work to other unicellular organisms apart from yeast, to study their pheromone pathways and try to identify possible siimlarities

between the pheromone pathway across species.

Bibliography

[1] *Cross validation*, http://www.cs.cmu.edu/~schneide/tut5/node42.html.

[2] *Snoopy - petri net editor and animator*, http://www-dssz.informatik.tu-cottbus.de/.

[3] *Saccharomyces genome database*, http://www.yeastgenome.org/.

[4] *Weka 3: Data mining software in java*, http://www.cs.waikato.ac.nz/ml/weka/.

[5] *Yeast*, http://en.wikipedia.org/wiki/Yeast.

[6] *Analysis of biological networks edited by bjorn h. junker and falk schreiber*, Wiley-Interscience Publication, 2008.

[7] Claudine Chaouiya, *Petri net modelling of biological networks*, BRIEFINGS IN BIOINFORMATICS **8** (2007).

[8] Simon Hardy and Pierre N. Robillard, *Modeling and simulation of molecular biology systems using petri nets: Modeling goals of various approaches*, Journal of Bioinformatics and Computational Biology **2** (2004).

[9] Dr. Steven Harris, personal communication.

[10] Monika Heiner, Ina Koch, and Jurgen Will, *Model validation of biological pathways using petri nets - demonstrated for apoptosis*, BioSystems **75** (2004).

[11] FEI LIU, MONIKA HEINER, and CHRISTIAN ROHR, *The manual for colored petri nets in snoopy - $qpn^c/spn^c/cpn^c/ghpn^c$*, Tech. report, Faculty of Mathematics, Natural Sciences and Computer Science, Brandenburg University of Technology, Cottbus, March 2012.

[12] Hiten Madhani, *From a to α yeast as a model for cellular differentiation*, COLD SPRING HARBOR LABORATORY PRESS, 2007.

[13] TOM M. MITCHELL, *Machine learning*, The McGraw-Hill Companies, Inc., 1997.

[14] Tadao Murata, *Petri nets: Properties, analysis and applications*, Proceedings of the IEEE **77** (1989).

[15] Andrea Sackmann, Monika Heiner, and Ina Koch, *Application of petri net based analysis techniques to signal transduction pathways*, BMC Bioinformatics (2006).

[16] Xiaowei Zhu, Mark Gerstein, and Michael Snyder, *Getting connected: analysis and principles of biological networks*, Genes Dev. **21** (2007).