COMPARATIVE ANALYSIS OF GENE PREDICTION METHODS AND DEVELOPMENT OF A FUNGAL GENOME DATABASE SYSTEM

by

Skanth Ganesan

A THESIS

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

Dr. Stephen D. Scott

Dr. Steven Harris & Dr Etsuko Moriyama

Lincoln, Nebraska

August, 2004

COMPARATIVE ANALYSIS OF GENE PREDICTION METHODS AND DEVELOPMENT OF A FUNGAL GENOME DATABASE SYSTEM

Skanth Ganesan, M.S.

University of Nebraska, 2004

Advisors: Drs. Stephen D. Scott and Etsuko Moriyama

Fungi represent one of the three major eukaryotic kingdoms with plants and animals. A vast number of fungi are filamentous and they have enormous health, economic, and ecological impacts on our human life. Multiple fungal genome projects have been planned and some new draft genomes have been recently completed. Multiple gene prediction programs are being used for identifying protein coding genes from these fungal genomes. However, existing gene-models built for unrelated organisms have been often applied to new fungal genomes because no model from specific fungal species is available for their optimum gene prediction. The objectives of this thesis are to analyze various gene mining methods, to extract genomic variables from various fungal genomes, and to develop an integrated genome database system that will facilitate more efficient genome annotation of filamentous fungi. The results obtained by analyzing three filamentous fungal genomes (*Neurospora crassa*, *Aspergillus nidulans*, and *Fusarium graminearum*) showed that each possesses a surprisingly large number of predicted genes with no apparent homologue in any other organism, thereby highlighting the need for accurate gene prediction programs. Three gene mining methods (GLIMMER, GLIMMERM, and GenScan) were used against the *N. crassa* genome and the performance for identifying short gene candidates specific to the species was examined. The results showed that GLIMMER, although it was developed primarily for prokaryotic genomes, as well as GLIMMERM, appeared to be useful for the fungal genome

annotation. More use of these gene prediction methods on fungal genomes should be considered. The extracted genomic variables will help us understand the genome specific features among fungi and other organisms. Integrating all the above information through a database system will help our understanding of the fungal genomes and facilitate optimizing gene prediction for fungal genome projects.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my family for their blessings and prayers, and God, for providing determination and success.

I am extremely grateful to my advisors: Dr. Etsuko Moriyama and Dr. Steven Harris, for providing an opportunity to work with them and expand my knowledge during the course of the program. I would like to acknowledge my CSE advisor: Dr. Stephen Scott and committee member Dr. Jitender Deogun for careful reading of thesis and providing valuable insight.

Special appreciation to the members of Moriyama lab. It was wonderful to work with you all. I want to specially thank our system administrator Yavuz Yavuz for his various help in the successful running of this project. My deepest thanks to the members of Lincoln Bhajan group. No words can express my thankfulness to Dr. Ram Bishu & family, Nataraj & family and Sampath & family, for making my stay in Lincoln enjoyable and memorable for life. I would like to thank my brother Gopal and sister Jayashree for their support and constant encouragement.

I want to finally thank the Department of Computer Science & Engineering, School of Biological Sciences, Plant Sciences Initiative and University of Nebraska - Lincoln for giving me a chance to do graduate research and for funding this project.

Table of Contents

Chapter 1 Introduction

1.1 Objectives of the thesis

Fungi, plants, and animals represent the three major kingdoms of eukaryotic organisms. Fungi form a large eukaryotic kingdom comprising more than 100,000 species, including yeasts and molds. Due to their relatively small genome sizes, fungi have been especially suitable for genome analysis. *Saccharomyces cerevisiae*, the budding yeast, was one of the first eukaryotic species whose genome was completely sequenced [1]. The genome of another yeast species, *Schizosaccharomyces pombe*, the fission yeast, has been also sequenced completely and analyzed in detail [2]. However, the information gained from these two yeast species represents only a small fraction of the fungal kingdom. This is why a large number of genome projects for non-yeast (filamentous) fungal species is now being planned.

These recent fungal genome projects require more efficient gene prediction tools. Due to the difference in genome organization, the methods optimized for the yeast and other organism genomes cannot accurately identify all of the genes in the filamentous fungal genomes. There have been also only a few studies on gene prediction problems specific to filamentous fungal genomes.

The goals of this thesis are to collect genomic information that can facilitate optimizing various gene prediction methods especially for filamentous fungal genomes, and to develop a support database system. The project has three components: 1) analysis of different gene mining methods, 2) extraction of genomic variables and comparative analysis of different fungal genomes, and 3) development of an integrated fungal genome

database system. The focus was placed on a filamentous fungus, *Neurospora crassa*, genome. The plan for the future is to extend the database to include other filamentous fungus species, e.g., *Fusarium graminearum* and *Aspergillus nidulans*. The data extracted contain detailed information on gene structure from fungal genomes such as intron and exon lengths and base composition in introns and exons. The extracted genomic variables will be useful to understand the genome specific features among fungi and other organisms. Integrating these information through a database system will help our understanding of fungal genomes and improvement of existing gene prediction methods.

1.2 Significance of fungal research

Fungi have spread through diverse natural habitats living on the degradation of a large variety of organic materials. Fungal activities affect human, other animals, and plants in many ways and hence are studied in greater details. Fungi serve as important models for biomedical research, and provide a wide range of evolutionary comparisons.

Fungi are divided into two broad groups, *Ascomycetes* and *Basidiomycetes*. The main difference between these two groups is in the way which they produce their microscopic spores. *Basidiomycetes* produce the spores externally on the end of specialized cells called *basidia,* whereas *Ascomycetes* produce spores internally inside of a sac called an *ascus*. *Basidiomycetes* include mushrooms, bracket fungi, and several mold like fungi called rusts and smuts (e.g., *Tremella fuciformia*, *Puccinia triticium*). These fungi damage grains, food crops, and other plants. *Ascomycetes* include unicellular

yeasts, cup fungi, truffles, morrels, and mildews (e.g., *Saccharomyces cerevisiae*, *Aspergillus nidulans, Neurospora crassa*). They are destructive parasites of food crops.

Fungi are also grouped based on their morphology: filamentous or non-filamentous. Filamentous fungi are unique organisms producing a wide range of natural products called secondary metabolites. These compounds are very diverse in structure and perform functions that are not always known. Filamentous fungi have bodies composed of thread-like long cells called *hyphae*. The filamentous cells are connected end-to-end and grow in a branching fashion forming a network. Filamentous fungi belong to both *Ascomycetes* and *Basidiomycetes*. The two yeast species, *Saccharomyces cerevisiae* and *Schizosaccharomyces pombe*, whose complete genomic sequences are already available, are non-filamentous fungi, and they belong to *Ascomycetes*.

The filamentous fungi are extremely useful to synthesize a wide range of economically important compounds, enzymes, and secondary metabolites, including antibiotics. They are the most important group of plant pathogens, causing very significant losses in crop yield world-wide. And they have been used as model organisms for understanding broad aspects of eukaryotic cellular regulation.

1.3 Fungal genome project

Fungal genomes are modest in size (ranging from 7 to 40 Mb) with few repeats. The high gene density makes them extremely cost effective for gene discovery. The efforts to expand genome sequencing to filamentous fungi date back to the late 1990s as the vast majority of fungi are filamentous. However, there are also difficulties in predicting genes in filamentous fungal genomes as described later.

The fungal genome project is coordinated by the Whitehead Institute. They recognized that existing genomic sequences; i.e., the budding (*S. cerevisiae*) and fission yeast (*S. pombe*) genomes, do not provide sufficient information on the entire fungal diversity. The current plan of fungal genomics is to approach in a kingdom-wide manner by selecting a balanced collection of fungal species maximizing the overall values for comparative genomics, evolutionary studies, and eukaryotic biology. The Fungal Genome Initiative (http://www.broad.mit.edu/annotation/fungi/fgi/) has identified a set of 44 new fungi as immediate targets for sequencing with an emphasis on clusters of related species, of which eight fungal genome assemblies have been fully released and two more are currently under prerelease. Figure 1.1 is a taxonomic representation of the fungal species proposed by the Fungal Genome Initiative. Table 1.1 contains the details of species whose complete genomes are available. As mentioned earlier, the genome sizes of the fungal species are smaller in size (40 Mb or smaller) in comparison to the other eukaryotes (e.g., the size of the human genome is 3,200 Mb), as only a small portion of the fungal genomes has redundant (repeated) information and most fungal genomes have very short introns (between 50 and 200 bp).

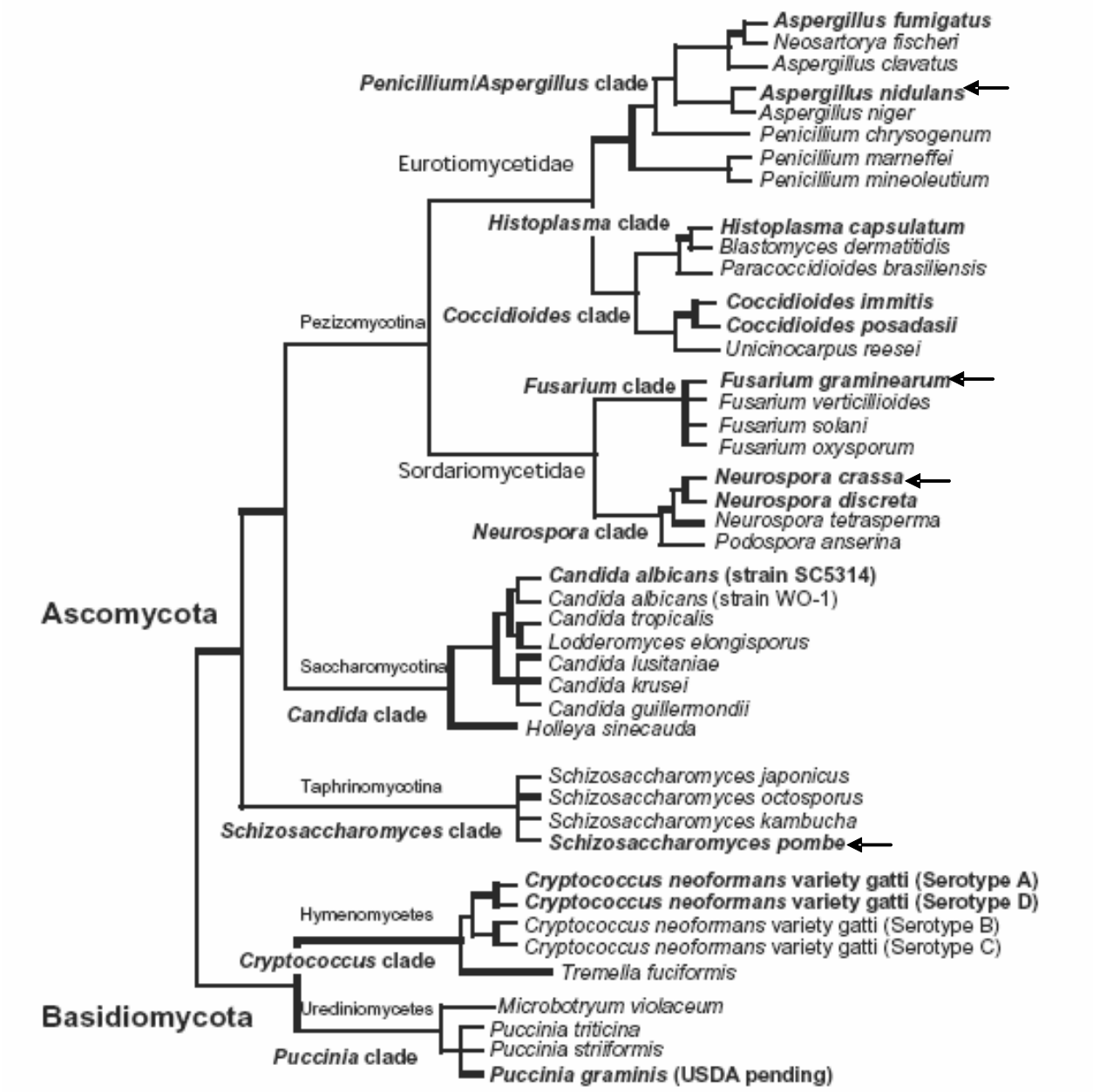


Figure 1.1 Taxonomic relationships of fungal species whose genomes are proposed for complete sequencing (taken from the Whitehead website: http://www.broad.mit.edu/annotation/fungi/fgi/). The best-supported branches with current data are indicated with thick lines. The fungal species used in this study are marked with arrows. Two species used in this study but not listed in the figure are *Saccharomyces cerevisiae* belonging to the *Candida* clade, and *Magnaporthe grisea* belonging to the *Sordariomycetidae* subclass.

Table 1.1 Complete fungal genomes available.

Species	Genome project status	Genome size (Mb)
Neurospora crassa	Annotated assembly released	40
Magnaporthe grisea	Annotated assembly released	40
Aspergillus nidulans	Assembly released	31
Fusarium graminearum	Assembly released	40
*Cryptococcus neoformans**	Assembly released	20
*Saccharomyces paradoxus**	Annotated assembly released	12
*Saccharomyces mikatae**	Annotated assembly released	12
*Saccharomyces bayanus**	Annotated assembly released	12
Ustilago maydis	Assembly in pre-release	20
Coprinus cinereus	Assembly in pre-release	38
*Saccharomyces cerevisiae**	Annotated assembly released	12
*Schizosaccharomyces pombe**	Annotated assembly released	14

Note: Non-filamentous fungi are marked with *, others are filamentous fungi.

1.4 Gene prediction methods

Computational gene identification plays an important role in genome projects. A challenge in bioinformatics is to rationally design analysis methods to interrogate the exponentially increasing sequence information. Various gene prediction methods have been developed. Some of these programs predict protein-coding regions in genomic sequences, while others predict a set of exons/introns and explicitly assemble genes. The two widely used approaches for genome annotation are similarity methods (extrinsic methods) and gene prediction methods (intrinsic methods). Only a half or fewer of genes can be annotated by searching similarities to other known genes or proteins, and remaining genes need predictive methods to be identified. Many gene prediction programs are currently publicly available. Some of the well-known gene prediction programs are GenScan, HMMGene, GeneMark, and Pombe. The methods used in these programs include hidden Markov models, linear discriminant analysis, and probabilistic models of gene structure that rely on features such as compositional differences and signals [3, 4]. Some representative methods are described in detail in Chapter 3.

1.5 Gene prediction problems specific to filamentous fungi

The problems common in these gene identification programs are that the algorithms must be trained using information gathered from a set of known genes and the quality of prediction strategies employed varies from organism to organism. The primary reason is that the method depends strongly on the gene samples in the training set and how the training set represents the entire gene set in the genome in question. Despite such limitations, gene models previously built for an organism are often applied to newly

sequenced unrelated organisms, for which no model or method has yet been tuned. For example, most of the programs used earlier in finding genes in *N. crassa*, a filamentous fungus species, failed because of the improper model used for this organism [5]. Recently, the German Neurospora Genome Project used gene modeling to predict genes in the *N. crassa* genome [6]. Gene modeling was based on prediction obtained with FGENESH [7]. FGENESH is based on hidden Markov model similar to Genie [8] and GenScan [9]. It is trained on 3,218 *N. crassa* genes. Predictions by GeneMark, GenScan, and GENEFINDER as well as significant matches to sequences of Expressed Sequence Tags (ESTs) and known genes were incorporated for corrections where appropriate. Gene predictions were restricted to open reading frames (ORFs) longer than 100 codons (300 bp). Prediction performance of the above methods (GeneMark and GenScan) was comparable to some extent to the predictions of FGENESH. The current Whitehead fungal genome annotation is based on a combination of FGENESH, FGENESH+, and GENEWISE.

One problem probably specific to but common among filamentous fungal genomes is that they contain many very short genes, shorter than 50 codons. Usually these short genes are beyond the detection power of available gene prediction methods and many methods simply ignore such short genes. Also several gene candidates have been predicted to contain extremely long introns. They could be indeed a single coding region or alternatively each exon could represent a small coding sequence.

In order to optimize gene prediction methods, more detailed information on gene structure, intron and exon lengths, splice site signals, base composition, codon usage bias and so on, needs to be collected from various fungal genomes. However, such

information crucial to gene annotation is not readily available in any existing genome databases. Current genome databases (e.g., *Arabidopsis thaliana*, yeast) focus more on developing data representation, visualizations, and query tools [10].

1.6 How this thesis is organized

The remaining part of the thesis is organized as follows. Chapter 2 presents an overview of the project. Chapter 3 describes the fungal genomes and datasets used. Chapter 3 also deals with various methods used in this study. It includes details on the BLAST similarity search methods of transmembrane prediction methods and gene prediction programs. In Chapter 4, the implementation of the fungal database system is described. It explains the various tables that are used for designing the system. Chapter 5 describes how the different prediction methods were used and the results of various analyses. Finally Chapter 6 concludes the thesis and presents some suggestions for the future work.

Chapter 2 Overview of the Project

The overall process of the project is split into the following four components:

1) **Comparative analysis of the fungal genomes**. The entire ORF sets of three filamentous fungal species (9,541 ORFs of *Aspergillus nidulans*, 11,640 ORFs of *Fusarium graminearum*, and 10,082 ORFs of *Neurospora crassa*; all available from the Whitehead genome project) were searched against the non-redundant (NR) database from the National Center for Biotechnology Institute (NCBI), which includes approximately two million sequence entries across various organisms. The search was also performed against five fungal genomes including *Magnaporthe grisea, Aspergillus nidulans, Fusarium graminearum, Saccharomyces cerevisiae*, and *Schizosaccharomyces pombe*, which contain approximately 43,000 ORFs in total. These searches can provide insights on how these filamentous fungal genes (or ORFs) are similar to those of other organisms and help identify genes/ORFs that are characteristics of these fungal genomes.

2) **Analysis of various gene mining methods**. GenScan, GLIMMER, and GLIMMERM methods were compared and potential advantages of using these methods on fungal genomes were identified. These methods were applied to the entire genomic data of *N. crassa*. Their performance was compared with the existing genome annotation by the Whitehead genome project. New gene candidates previously not annotated were identified and examined.

3) **Extraction of genomic variables**. Various gene and genomic information were extracted from three fungal genomes (*N. crassa, S. pombe,* and *S. cerevisiae*). It includes nucleotide frequencies (single-, di-, and trinucleotides) and lengths of introns

and exons. The extracted genomic variables were compared among fungus species to understand the genome specific features.

4) **Development of integrated genome database**. The database was constructed based on the *N. crassa* genomic data to compile information used in various gene prediction methods (described above). The database stores, for example, entry files of *N. crassa* sequences, results of various sequence analyses (e.g., base composition), results from similarity search against various databases (NCBI non-redundant database and fungal genome databases), and the results using various gene prediction programs. The stored data can be visualized using a web-based graphical interface. This database system would support the development of organism specific gene prediction methods in the future.

The flow chart given in Figure 2.1 explains the scheme of the entire project.

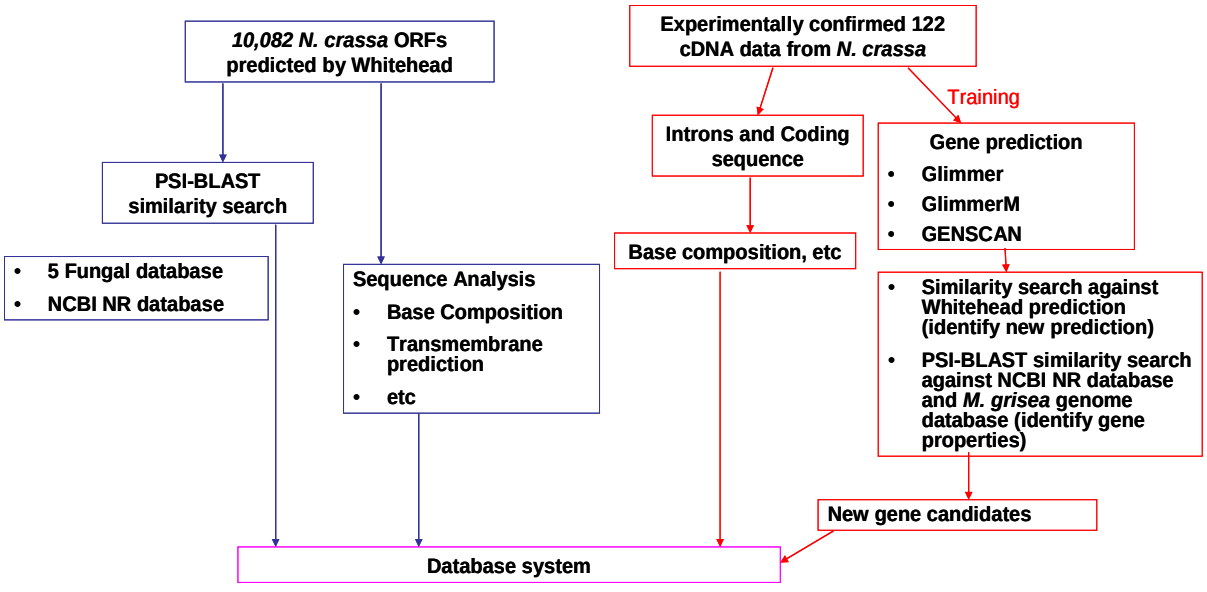


Figure 2.1 Schema of the project

Chapter 3 Materials and Methods

3.1 Datasets

3.1.1 Fungal genome databases

Six fungal genomes were used for this study: *Neurospora crassa, Magnaporthe grisea, Aspergillus nidulans, Schizosaccharomyces pombe, Saccharomyces cerevisiae*, and *Fusarium graminearum*. Except *S. pombe* and *S. cerevisiae*, other four species are filamentous fungi. These fungal species are marked with arrows in Figure 1.1. The predicted coding sequences (CDS) and translated protein data as well as the genomic DNA were obtained from the Whitehead website (www.broad.mit.edu/annotation/fungi/fgi/) and from the National Center for Biotechnology Information (NCBI; www.ncbi.nlm.nih.gov).

- ***Neurospora crassa***

Neurospora crassa is one of the most widely studied filamentous fungi and serves as a model for eukaryotic organisms. The size of the *N. crassa* genome is approximately 40 Mb comprising of seven chromosomes. The Whitehead genome project has 821 contigs containing approximately 11,000 predicted genes/ORFs.

- ***Aspergillus nidulans***

This filamentous fungus species is one of the critical fungal systems in genetics and cell biology. It is important because it is closely related to a large number of other *Aspergillus* species of industrial and medical significance; e.g., *A. niger*, *A. oryzae*, *A. flavus*, and *A. fumigatus*. *A. nidulans* is a member of the *Ascomycetes*. The size of the *A.*

nidulans genome is approximately 31 Mb. The Whitehead genome project has 248 contigs containing estimated 9,541 protein coding genes.

- ***Fusarium graminearum***

Fungi in the genus *Fusarium* cause a variety of seedling disease on nearly every species of cultivated plants. The size of the *F. graminearum* genome is approximately 40 Mb. The Whitehead genome project has 511 contigs containing estimated 11,640 genes. It is also a filamentous fungus.

- ***Magnaporthe grisea***

Magnaporthe grisea, a filamentous fungus and the causal agent of rice blast disease, is one of the most devastating threats to food security worldwide. The size of the *M. grisea* genome is approximately 40 Mb contained in seven chromosomes. The Whitehead genome project has 2,273 contigs containing estimated 11,109 genes. *M. grisea* is considered to be closely related to *N. crassa*, and both belong to the *Sordariomycetidae* subclass (Figure 1.1).

- ***Schizosaccharomyces pombe***

Schizosaccharomyces pombe or the fission yeast is a single-celled free living fungus sharing many features with cells of more complicated eukaryotes. It is a non-filamentous fungus. The 13.8 Mb genome of *S. pombe* is distributed on sixteen chromosomes and contains approximately 5,000 genes. The complete genome data were downloaded from the NCBI website.

- ***Saccharomyces cerevisiae***

Saccharomyces cerevisiae or the budding yeast has the 12 Mb genome distributed over three chromosomes. The genome contains approximately 6,000 genes. It is also a

non-filamentous fungus, and the complete genomic data were downloaded from the NCBI website. *S. cerevisiae* belongs to the *Candida* clade (Figure 1.1).

3.1.2 NCBI non-redundant database

The NCBI non-redundant (NR) protein database contains protein sequence entries compiled from a variety of sources, including SwissProt, PIR, PRF, PDB, and translations from annotated coding sequences in GenBank and RefSeq. As of July 4th 2004, the database contains 1,921,076 sequences from various organisms including fungi, animals, plants, and prokaryotes. Although it is called "non-redundant" the database does contain multiple submissions of the same gene sequence.

3.1.3 Training datasets used for gene prediction methods

Experimentally confirmed cDNA sequences of *N. crassa, S. pombe*, and *S. cerevisiae* were used for training GLIMMER, a gene prediction method. They were collected from the NCBI nucleotide database. The training datasets include 120, 157, and 613 cDNA sequences from *N. crassa*, *S. cerevisiae*, and *S. pombe*, respectively.

All the fungal genome sequences as well as the NR database were formatted locally before their use. Formatting protein or nucleotide databases is required before these databases can be used by BLAST similarity search programs. The BLAST programs are explained in the next section.

3.2 Sequence analysis programs

3.2.1 BLAST programs

The BLAST (**B**asic **L**ocal **A**lignment **S**earch **T**ool) programs are widely used tools for searching protein and DNA databases for sequence similarities [1]. BLAST programs compare protein or DNA queries with protein or DNA databases in any combinations: **blastp** compares an amino acid query sequence against protein sequence databases; **blastn** compares a nucleotide query sequence against nucleotide sequence databases; **blastx** compares a nucleotide query sequence translated in all six reading frames against protein sequence databases; **tblastn** compares a protein query sequence against nucleotide sequence databases dynamically translated in all six reading frames; and **tblastx** compares the six-frame translations of a nucleotide query sequence against the six-frame translations of nucleotide sequence databases.

BLAST is a heuristic algorithm that attempts to optimize a specific similarity measure. The BLAST program requires computation time proportional to the product of the lengths of the query sequence and database searched. Since the rate of change in database sizes currently exceeds that of processor speeds, computers running BLAST are subjected to increasing load.

In this study, **blastp** is used for finding similar sequences in protein databases. Like other BLAST programs, **blastp** is designed to find local regions of similarity. When sequence similarity spans the whole sequence, **blastp** will also report a global alignment, which is the preferred result for identifying closely related homologous proteins. **blastp** filters out regions of low complexity from database sequences by default. It is a fast

single iteration process and gives a good idea on relationships between the query and hit sequences.

3.2.1.2 PSI-BLAST (**P**osition-**S**pecific **I**terated **BLAST**)

PSI-BLAST is an iterative program to search databases for proteins with distant similarity to a query sequence [12, 13]. The basic strategy of PSI-BLAST is to first construct a multiple alignment from the output of a **blastp** protein similarity search (the first iteration). A position-specific score matrix (PSSM) is generated from this multiple alignment, and the databases are searched again using the PSSM as a query. The process may be iterated multiple times as new significant similarities are found. The subsequent searches become more and more flexible (sensitive and less specific) by incorporating more substitution possibilities at each amino acid position. Therefore, database searches using PSSMs are much better able to detect weak relationships than are those that use simply a sequence as the query.

PSI-BLAST draws its power from two sources. The first is improved estimation of the probabilities with which amino acids occur at various positions, leading to a more sensitive scoring system. The second is relatively precise definition of the boundaries of important motifs. Each PSSM constructed has the length precisely equal to that of the original query sequence. The same gap penalties are used throughout the procedure and there is no position specific penalty.

3.2.1.3 Drawbacks

A major potential problem of PSI-BLAST is so called "PSSM corruption." At the end of each iteration, PSI-BLAST constructs a multiple alignment, from which it abstracts a PSSM. If a sequence *S* that is unrelated to the original query sequence *Q* is included in the multiple alignments, then the resulting PSSM will produce highly significant alignments to sequences related to *S* as well as those related to *Q* (but not related to *S*). Such a PSSM is said to have been corrupted, and the search results from further iterations are unreliable [13]. With PSI-BLAST, a single corrupted PSSM can yield many false positives with very low *E*-values. The results of a corrupted search can be almost completely meaningless, and this casts considerable doubts on the reliability of results even if the majority of searches are uncorrupted. A corrupted search can also consume a great quantity of computing time, exhaust all virtual memory causing a crash, or produce a huge volume of incorrect outputs, limiting the applicability of PSI-BLAST to large-scale, automatic annotation projects.

One may attempt to avoid the PSSM corruption problem by setting the parameter *h* to a sufficiently low value. The parameter *h* defines the maximum *E*-value for a similar sequence to be included in the multiple alignments. For most queries, the threshold $h = 0.001$ is sufficient to avoid the corruption. However, even at this or much lower values of *h*, still a small percentage could yield corrupted PSSMs. In this project, the E-value of 0.005 was used as the maximum threshold to avoid this problem but not to make the search too restrictive.

E-values for a given database sequence do not remain constant between PSI-BLAST search iterations. This is particularly the case between the first and subsequent

search iterations. In the first search round, matches between query and database sequences are scored using a static scoring matrix (e.g., BLOSUM62). In contrast, successive search rounds determine scores by comparing database sequences to the PSSM created based on the multiple alignment obtained in the previous round of search. As more sequences are added to build PSSMs as the search iterates, the scores for matching sequences will change. In general, the scores tend to increase (and the E-values tend to decrease) in the later iterations as the PSSMs become more flexible (match with more less similar sequences); the opposite changes are also possible. Therefore, only the scores (or E-values) from the first iteration represent the actual similarities between the query sequence and hits. In this project, therefore, the E-values from the first PSI-BLAST iteration were used to represent the similarity level between the *N. crassa* ORFs and hit sequences.

3.2.1.4 Usage

All BLAST programs were run locally with the following command lines:

- blastall –p blastp –d <db> -i <query> -o <output> -e <expectation value>
- blastpgp –d <db> -i <query> -j <iterations> -o <output> -e <expectation value> –h <threshold value>

where

-p specifies program used (blastp, blastn, etc);

–d denotes the formatted protein database using formatdb;

-i is the protein query sequence file;

-o is the output file;

–e is the threshold expectation value to keep the hit results (0.005 was used);

-j is the number of iterations (5 was used);

-h is the E-value threshold for inclusion in PSSM (0.005 was used).

All hit sequences with the E-values better than the given *h* (0.005) are used for constructing the PSSM. The substitution matrix used for the first iteration by default is BLOSUM62 and gap penalties are 11 and 1 for opening and extending, respectively. 'blastpgp' command is used to run PSI-BLAST. The program stops when the database search finds no new sequence hit that are better (lower) than the E-value threshold set by the –h option even if the number of iterations is still fewer than that set with –j option.

In this study, PSI-BLAST search was done using individual *Neurospora crassa* protein sequence against the locally formatted NR database downloaded from the NCBI. PSI-BLAST search was also carried out against the five fungal genomes (*Magnaporthe grisea, Aspergillus nidulans, Schizosaccharomyces pombe, Saccharomyces cerevisiae*, and *Fusarium graminearum*). These search results provide us insights on the relationships among fungal and other organismal genes. If there is no hit, it tells us the uniqueness of those fungal genes.

3.2.1.5 Description of output

The output of BLAST programs displays the similarity score of the hit, description of the hit sequence, expectation value (E-value), and alignment statistics showing % identities (proportion of the identical amino acid pairs in the alignment), % positives (proportion of the amino acid pairs that are biochemically similar in the alignment), and the number of gaps, as well as the pairwise alignment between the query

and hit sequence. There are two types of scores: the raw score is given without any units, and the normalized score is in bits unit. A higher score indicates a greater similarity, and high scoring hits have smaller E-values. The E-value is the number of alignments that are expected at random given the size of the search space, scoring matrix, gap penalties, and similarity score. The smaller the E-value of the hit in the database search, more significant the hit is and less likely it being a random hit.

3.2.2 Transmembrane prediction programs

3.2.2.1 Introduction

Membrane proteins are important in many cellular processes and functions in all biological systems. They are, for example, receptors for neurotransmitters or hormones, form ion channels, or serve as the respiratory chain. Transmembrane proteins have a specialized kind of secondary structure architecture, which allows a part of the protein, hydrophobic alpha-helical regions, to be embedded in the membrane, and the remaining parts, hydrophilic regions, to be outside of the membrane. The prediction of transmembrane regions in proteins is an important aspect of bioinformatics as transmembrane proteins form nearly 25% of all proteins [14, 15].

Earlier approaches were based on simple hydrophobicity analyses [16-19]. They used information only on the amino acids that contribute to the formation of transmembrane helices. The drawback of these methods is that the results depend on fixed hydrophobicity thresholds. Some helices may be missed if they fall just under the threshold used. This is not unusual in proteins with many membrane-spanning helices that form a bundle in which non-hydrophobic residues may make contacts between

helices. Because of a large difference in physical environments of the membrane-spanning segments and the cytoplasmic or extracellular parts of the membrane proteins resulting in different amino acid compositions, it is reasonable to expect that more accurate prediction methods can be developed if more information on the amino acid compositions in both segments (transmembrane and loop regions) is considered. Amino acid composition will reflect the diverse roles of membrane proteins in cells and in different environment.

Using hidden Markov models (HMMs) is very well suited for predictions of transmembrane regions because it can incorporate hydrophobicity, charge bias, helix lengths, and grammatical constraints into one model for which algorithms for parameter estimation and prediction already exist. The basic principle is to define a set of states, each corresponding to a region or specific site in the proteins being modeled. By defining states for transmembrane helix residues and other states for residues in loops and those on either side of the membrane, and by connecting them in a cycle, we can produce a model that in architecture closely resembles the biological system. Each state has an associated probability distribution over the 20 amino acids characterizing the variability of amino acids in the region (state) it models. Two of the most widely used transmembrane prediction methods: HMMTOP [20] and TMHMM [15, 21], were used in this study and they are discussed next.

3.2.2.2 HMMTOP (**H**idden **M**arkov **M**odel for **TO**pology **P**rediction**)**

3.2.2.2.1 Method

HMMTOP [20] is based on the hypothesis that the differences between amino acid distributions in various structural parts are the main driving force in folding membrane proteins; i.e., the topology of transmembrane proteins may be determined by the simple fact that the amino acid compositions of the various structural parts do show maximum differences rather than by enforcing specific compositions in these parts. The sum of divergence values between the distributions of amino acids in the structural parts and in the whole protein is used to measure the difference. The topology of membrane proteins can be determined if their amino acid sequences can be segmented to some parts (e.g., inside, outside, and within the membrane) in such a way that the product of the relative frequencies of the amino acids of these segments along the amino acid sequence is maximized.

The prediction method based on this model has three steps. First, the initial values of HMM parameters (the initial state and the state transition probabilities) have to be set. The initial parameters can be chosen from random values or from predetermined values. The next step is the optimization of these parameters for the amino acid sequence(s) studied. The third step is to find the best state sequence by the *viterbi* algorithm given the model and parameters. Elements of the state sequence show the localization of each amino acid in the query sequence. Default values of the parameters and the pseudocount array have been obtained from the amino acid sequences of transmembrane proteins whose topologies are experimentally well defined. Since optimization of the parameters can work for multiple sequences, prediction can be made

using multiple sequence information. One of the advantages of HMM, on the other hand, is that related proteins do not have to be aligned before the prediction (the sequences need to have a certain level of similarities, however).

3.2.2.2.2 Drawbacks

Even though HMMTOP predictions are highly accurate (>80% accuracy) [20], there are several weak points in this method. One of them is that, even using multiple sequences, the same predicted topology is not guaranteed for each sequence. The next point is related to the multiple optima problem in the optimization process. Since the *baum-welch* algorithm cannot find the global optimum of the likelihood function, the correct way to handle this problem may be by an exhaustive search for the optimum. Because of the huge computational demand for searching, every iteration was started from the same point.

3.2.2.2.3 Running the Program

The program is available from http://www.enzim.hu/hmmtop. The program can interpret multiple sequences in two different ways. In the *mpred* mode, prediction will be provided for the first sequence interpreting other sequences as homologues to the first one. The homologous sequences provided need not be aligned. In the *spred* mode, HMMTOP simply evaluates the input sequences one by one, providing independent prediction for each of them using only single sequence information. It supports three input sequence formats (FASTA, NBRF/PIR, and SWISSPROT) and offers various

output formats. A sample output of HMMTOP is included in Appendix A. HMMTOP program was run locally with the following command line:

hmmtop –if=infile –of=outfile –lf=logfile –pi=spred –sf=FAS

- -if=name, --input_file=name [name of the input sequence file. If name is – then the program reads from the standard input].
- -of=name, --output_file=name [name of the output sequence file. If this option is omitted or name is – then the program writes to the standard output].
- -sf=format, --sequence_format=format [format of the sequence(s). Format may be FAS for fasta format (default), PIR for NBRF/PIR format, or SWP for SWISSPROT format].
- -pi=mode, --process_inputfile=mode [treats sequences in input file as single or homologous sequences. The mode may be *spred* or *mpred*].

The *spred*, single prediction mode, was used in this study. The input file contained the 10,082 *N. crassa* protein sequences downloaded from the Whitehead Institute.

3.2.2.3 TMHMM

3.2.2.3.1 Method

TMHMM is also based on an HMM approach. As shown in Figure 3.1, it models various regions of a membrane protein: helix caps, middle of helix, regions close to the membrane, and globular domains [21].

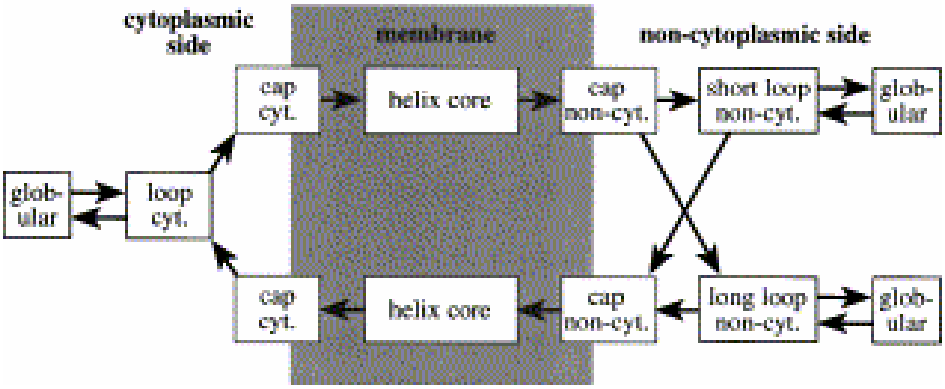


Figure 3.1 Layout of the hidden Markov model used in TMHMM (taken from Krogh *et al.* [15]).

Due to the different residue distributions on the different sides (inside or outside) of the membrane, seven different states are used: one for the helix core, two for caps on either side, one for loops on the cytoplasmic side, one each for short and long loops on the non-cytoplasmic side, and one for globular domains in the middle of each loop. The amino acid emission probabilities of all states of the same type are estimated collectively.

The transmembrane helix is modeled by two cap regions of five residues each, surrounding a core region of variable length (5-25 residues). This allows for helices to be 15-35 residues long. The HMM parameters including the probabilities of the 20 amino acid residues in each state and the probabilities determining the length distribution of transmembrane helices have been estimated from a set of 160 proteins in which the locations of the transmembrane helices are known.

Prediction of the transmembrane helices is done by finding the most probable topology given the HMM. This gives a set of exact helix boundaries. However, there

could be many equally probable ways to place the helix boundaries, and some regions show predicted transmembrane helices with fairly low probabilities. In such cases, it would be useful to compare the three probabilities for a given residue: if being in a transmembrane helix, in the cytoplasmic side, or in the extracellular side.

3.2.2.3.2 Drawbacks

In general, there are several types of mis-predictions that can occur when predicting the topology of a membrane protein. The simplest errors are over- and under-predictions: i.e., predicting a transmembrane region where none should be actually present or missing a true transmembrane region. Another type of errors is that two adjoining transmembrane regions are joined together, so that they are predicted as a single long transmembrange region. Similarly, a long transmembrane region can be falsely predicted as being two short regions. TMHMM predicts transmembrane helices from single sequences with a high level of accuracy (>80%) [15]. The number of falsely merged helices and the number of falsely split helices are very low due to the natural modeling of helix lengths and the grammar used in the HMM. The main type of errors made by TMHMM is to predict signal peptides as transmembrane helices. And one of the most common mistakes by TMHMM is to reverse the direction of proteins when there is only one transmembrane segment.

3.2.2.3.4 Running the program

The server is available at http://www.cbs.dtu.dk/services/TMHMM/ with their default parameters.

- **Input**: The program takes proteins in fasta format**.** It recognizes the 20 amino acids, and the characters B, Z, and X are treated equally as unknown. Any other character is changed to X.
- **Output**: Sequence identifier, length of the protein sequence, and the expected number of amino acids in transmembrane helices. See Appendix B for an example output.

The 10,082 *N. crassa* protein sequences downloaded from the Whitehead Institute were used as an input in a single fasta file.

3.2.2.4 Kyte-Doolittle hydropathy plot

Kyte-Doolittle hydropathy plot [16] gives information on the possible structure of a protein. A hydropathy plot can indicate potential transmembrane or surface regions in proteins. Hydrophobic regions, i.e., transmembrane candidate regions, achieve positive values (or negative depending on the set of scores used). Each amino acid is given a hydrophobicity score between -4.5 and 4.5. The score of 4.5 is the most hydrophobic and the score of -4.5 is the most hydrophilic (or *vise versa*). A "window size" is the number of amino acids whose hydrophobicity scores will be averaged and assigned to the middle position in the window. Setting window size to 5-7 is suggested to be a good value for finding putative surface-exposed regions, whereas a window size of 19-21 yields a plot in which transmembrane domains stand out sharply, with values of at most 1.6 at their centers. The method starts with the first window of amino acids and calculates the average of all the hydrophobicity scores in that window. Then it moves down one amino acid (or a certain number of amino acids) and calculates the average of all the

hydrophobicity scores in the second window. This process continues to the end of the protein, computing the average score for each window and assigning it to the amino acid at the middle position in the window. The average scores are then plotted on a graph. The y axis represents the hydrophobicity scores and the x axis represents the amino acid position in the protein sequence.

In this study, I used a sliding window size of 16 amino acids (aa). The window size was selected after testing on different values. The window is shifted every 10 aa. The region is predicted to be transmembrane when the average hydrophobicity values in successive windows goes from positive value to below -0.05 and back to a higher value. The example plot is found in Appendix C.

3.2.3 Other sequence analysis

3.2.3.1 Cell wall protein prediction

The fungal cell wall is a dynamic structure. Its composition, properties, and form constantly change during the cell cycle and depending on growth conditions. Filamentous fungi are always in intimate contact with their surroundings. Their cell wall proteins are known to have amphipathic transmembrane regions as well as regions of high serine/threonine (Ser/Thr) contents. Initial analyses of the amino acid content in the known six *N. crassa* cell wall protein sequences have shown high concentration of serine/threonine, around 50%, compared to the genomic average of 25%. Based on this observation, a window size of 100 amino acids was used to search protein regions with more than 20% of the Ser/Thr content from each of *N. crassa* proteins.

Combination of high Ser/Thr content and having a transmembrane region is required to predict cell wall proteins. Appendix C shows the plots of Ser/Thr contents and hydrophobicities along the amino acid sequence of a known *N. crassa* cell wall protein (NCU00039.1).

3.2.3.2 Base composition analysis

Base composition analysis was done for the entire predicted ORF sets of *Neurospora crassa*, *Schizosaccharomyces pombe*, and *Saccharomyces cerevisiae* genomes. The analysis was done both for the coding exons and introns. The frequencies of four single nucleotides (A, T, G, and C), 16 dinucleotides (AA, AT, AG, AC, and so on), and 64 trinucleotides were computed. The same sets of analyses were done also for the experimentally confirmed set of cDNAs, as well as intron sequences (described in Chapter 5). These analyses should reveal any species-specific bias involved in these genomic properties. It will provide an idea how these genomic information should be incorporated in gene prediction methods.

3.3 Gene prediction programs

3.3.1 Introduction

A major goal of genome projects is to identify all genes in a given organism. Consequently, the development of automated gene-finding procedures has become one of the most active areas of research in bioinformatics. Protein-coding DNA sequences exhibit characteristics that distinguish them from non-coding sequences. For prokaryotic organisms, the task of gene identification is relatively easy as prokaryotic genomes are

rather small and genes are not interrupted by introns. Here, all open reading frames (ORFs) exceeding some threshold length are likely to code proteins. The gene-finding problem is much more complicated for eukaryotic organisms where the density of genes in the genome is about two orders of magnitude lower than in bacterial genomes and genes typically consist of multiple exons separated by introns of varying length. The commonly used approach for gene prediction is to train computer programs to recognize sequences that are characteristic of known exons in genomic DNA sequences. The patterns used to predict genes include intron-exon boundaries and upstream promoter sequences. However, in eukaryotes, these signals are poorly defined, and therefore cannot be searched by a simple pattern-matching technique as used with prokaryotes.

During the past few years, various prediction methods have been developed to identify genes in eukaryotic genome sequences [22, 23]. Recent studies show, however, that the reliability of these methods is limited for large genomic sequences as they cannot locate all possible exons encoded in the sequence [24]. Moreover, many gene-prediction programs have originally been tested on genomic sequences of only a few kilo bases (kb) in length where each sequence contained only a single gene. The performance of standard gene-prediction methods drops significantly when tested under more realistic conditions usually containing multiple genes [25].

Practically all existing gene-prediction programs rely on information derived from known genes. Major differences between existing methods are in how they assess if a stretch of genomic DNA looks as known genes. Two approaches are used. *Ab-initio* or *intrinsic* methods use content statistics such as ORF length or codon usage together with sequence signals like splice junctions to distinguish coding from non-coding regions.

GLIMMER [26, 27], GRAIL [28], GENEID [29], GenScan [30], and GeneMark [31] are among the most popular *ab-initio* programs. By contrast, *extrinsic* methods work by comparing genomic sequence to known ESTs or proteins in databases and check if a piece of the genomic sequence is similar to any known genes or proteins. This idea has been implemented in GENEWISE [32] and PROCRUSTES [33].

Neither *ab-initio* nor *extrinsic* methods can elucidate perfectly the complex and variable genomic structure of higher eukaryotic organisms. Their genes contain a large number of small exons separated by long intervening sequences (introns). Furthermore, in the actual genomes, some non-coding sequences could exhibit features of typical coding sequences (e.g., pseudogenes) and *vice versa*. Moreover, a large fraction of higher eukaryotic coding exons are very short, which cannot be effectively detected by commonly used gene-prediction programs. The following sections describe three gene-prediction methods: GLIMMER, GLIMMERM, and GenScan, used in this study for detecting mainly small exons in the *Neurospora crassa* genome. They were chosen among several others based on their possible ability to detect small coding regions, sensitivity and specificity values when used on other genomes.

3.3.2 GLIMMER (**G**ene **L**ocator and **I**nterpolated **M**arkov **M**odel**er)**

GLIMMER is a computational gene finder that was initially developed to predict genes in prokaryotic genomes [27, 28]. Gene finders for prokaryotes have an advantage in that genomes tend to be gene-rich, containing 90% coding sequences. One major problem is to correctly identify the genes when two or more open reading frames (ORFs) overlap. GLIMMER uses a technique called interpolated Markov model (IMM), a

generalization of Markov chain methods, to identify coding regions in microbial sequences. GLIMMER 1.0 has been used as the gene finder for some bacterial genomes (*Borrelia burgdorferi*, *Treponema pallidum*, *Chlamydia trachomatis*, and *Thermotoga maritima*) [27, 28]. GLIMMER 2.0 has several technical improvements to the GLIMMER 1.0 algorithm and works better in resolving overlapping ORFs.

GLIMMER uses an approach based on frequency of occurrence of nucleotides in a DNA to determine the relative weights of oligomers that have different lengths from 1 to 9 bp. First IMMs are created for the six open reading frames (three frames for each of the two strands: forward and reverse), and then used to score the entire ORFs. When there is an overlap between two high scoring ORFs, the overlapped regions are scored separately to determine the more likely gene.

3.3.2.1 Interpolated Markov models (IMMs)

A Markov chain contains a sequence of random variables X_i (i is the position in the sequence), where the probability distribution for each variable depends only on the preceding k variables X_{i-1}, … , X_{i-k} for some constant k. In the case of DNA sequences, the random variables X_i takes the value from the set of four nucleotide bases (a, c, g, and t). Depending on the order of the Markov chain used, the constant k takes values from 0 to 8. For example, a fixed first-order Markov chain is specified completely by a matrix of 16 conditional probabilities: $p(a|a)$, $p(a|c)$, $p(a|g)$, … , $p(t|t)$, where each of the terms represents the probability of the current base given the previous base. A second-order Markov chain predicts a base by looking at the two previous bases. In general, for a k^{th}-order Markov model, the number of probabilities we need to look into is 4^{k+1} for each

reading frame. In a 0^{th}-order model, the matrix contains only the individual probabilities of the four nucleotides (a, c, g, and t). In the case of a first order, the 16 dinucleotide (aa, ac, ag, at, …, tg, tt) probabilities are calculated by looking at the previous base. A second-order model gives the probability of 64 trinucleotides (aaa, aag, aac, aat,…, ttg, ttt). In principle, using longer oligomers is always preferable to using shorter ones, but only if sufficient data is available to produce probability estimates. Currently most of the gene finders use a 5^{th}-order fixed Markov chains (it uses hexamer nucleotide or di-codon frequencies) as they have proven to be effective for gene predictions [31, 34].

IMMs are generalization of fixed order Markov chains. The main difference between IMMs and fixed Markov models is that IMMs use varying number of bases for each prediction rather than making decision in advance regarding the number of bases to consider. This allows IMMs to be sensitive depending on the frequencies of particular oligomers in a genome. For example, if some 5-mers (oligomers having five bases) occur too infrequently, their probabilities cannot be estimated reliably, and they will not be used in the model. On the other hand, if some 8-mers occur sufficiently frequently, IMM use this longer context to make better predictions. Thus it has all the additional information for prediction.

From the training data sets, GLIMMER computes the probability for each nucleotide base (a, c, g, or t) following all *k*-mers ($0 \leq k \leq 8$). For each *k*-mer, weights are computed for use in different models. These weights and Markov models are interpolated to produce a score for each base in any potential coding region. The logs of

scores are summed to score each coding region. The probability that the model M generates the sequence S, P(S|M), is computed as

$$P(S|M) = \sum^{n}_{x=1} \mathbf{IMM}_8 (S_x)$$

Where S_x is the oligomer ending at the position x, and n is the length of the sequence. $\mathbf{IMM}_8 (S_x)$ is the 8th-order interpolated Markov model score computed as

$$IMM_k(S_x) = \lambda_k(S_{x-1}) * P_k(S_x) + [1- \lambda_k(S_{x-1})] * IMM_{k-1}(S_x)$$

where $\lambda_k(S_{x-1})$ is the numeric weight associated with the *k*-mer ending at the position x-1 in the sequence S, and $P_k(S_x)$ is the estimate obtained from the training data of the probability of the base located at x in the k^{th}-order model.

$$P_k(S_x) = P(s_x|S_{x,i}) = f(S_{x,i})/(\sum_{b\varepsilon\{a,c,g,t\}} f(S_{x,i}, b))$$

where $f(S)$ denotes the number of occurrences of the string $S = s_1s_2\ldots s_n$. GLIMMER uses two criteria to determine $\lambda_k (S_x)$. The first criterion is simply the frequency of occurrence. The current default threshold value is 400. The default threshold value gives 95% confidence that the sample probabilities are within 5% of the true probabilities from which the sample was taken. When there are insufficient sample occurrences of a context string (oligomer), additional criteria are employed to assign a λ value. For a given context string $S_{x,i}$ of length i, observed frequencies of the base $f(S_{x,i},a)$, $f(S_{x,i},c)$, $f(S_{x,i},g)$, and $f(S_{x,i},t)$ are compared with previously calculated IMM probabilities using the next shorter context, $IMM_{i-1}(S_{x,i-1}, a)$, $IMM_{i-1}(S_{x,i-1}, c)$, $IMM_{i-1}(S_{x,i-1}, g)$, and $IMM_{i-1}(S_{x,i-1}, t)$. Using a χ^2 test the two values are compared. If the values differ significantly, then the observed values are used. If they are consistent with IMM values, a lower value is given as they offer less predictive value. The value of $\lambda_k(S_x)$ that we associate with $P_k(S_x)$ can be regarded as a measure of our confidence in estimating the true probability. The

number of parameters we need to estimate grows exponentially with the level of the order and higher the order, the parameter estimates can be less reliable.

3.3.2.2 The GLIMMER system

The GLIMMER system consists of two programs: *build-imm* and *glimmer* (or *glimmer2*). The program *build-imm* takes an input set of sequences. The set can be complete genes or parital ORFs. It builds and outputs the interpolated Markov model. The second program *glimmer* uses this IMM to identify genes in a genomic sequence. GLIMMER does not use sliding windows to score the coding regions. Instead it identifies all ORFs that are longer than the threshold value and scores them in six possible reading frames. The ORF is assumed to have only one stop codon after the start codon in the sequence. It selects the frame that scores the highest for further examination of overlaps. If there is an overlap between reading frames, it selects the overlapped regions and scores them separately. Overlapping ORFs are resolved based on the length and a separate score computed for their overlapped regions. Suppose that A and B are two ORFs that overlap. If the overlap scores higher in A's reading frame and A is longer than B, we reject B. If the overlap scores higher in B's reading frame and B is longer than A, we reject A. Otherwise, both A and B are marked as "suspect".

GLIMMER 2.0 has resolved some of the prediction problems of GLIMMER 1.0. GLIMMER 1.0 occasionally discarded a gene due to the placement of its start codon in the 5' direction resulting in an overlap with another gene. GLIMMER 2.0 resolves overlapping problems by incorporating extra rules. The scoring is similar to that of GLIMMER 1.0 for potential overlapping genes, but the system attempts to move the

locations of the start codons much more aggressively. In the case of ORFs A and B overlap, there are four different orientations to be considered. The process of evaluating overlaps is performed in an iterative fashion to prevent unnecessary rejection of genes. The current version also helps to find genes that were missed earlier due to the high probability threshold score.

GLIMMER is the primary microbial gene finder at The Institute of Genomic Research (TIGR), and has been used to annotate the complete genomes of *Borrelia burgdorferi*, *Treponema pallidum, Thermotoga maritima, Deinococcus radiodurans, Mycobacterium tuberculosis*, and non-TIGR projects including *Chlamydia trachomatis* and *Chlamydophila pneumonia*e. The accuracy rates of gene prediction for bacterial and archaebacterial genomes (including *Escherichia coli*, *Haemophilus influenzae*, *Helicobacter pylori*, *Bacillus subtilis*, *Mycoplasma genitalium*) were close to 98% [26]. It has not been used for eukaryotic genomes.

Even though GLIMMER is developed for prokaryotic genomes, it could be still useful for finding eukaryotic genes. Many small eukaryotes, for example, have relatively high gene density and contain short genes without interrupted by introns, and these genes cannot be detected by commonly used prediction programs.

The GLIMMER system including the source codes was downloaded from The Institute of Genomic Research website (http://www.tigr.org). The example output of the program is listed in Appendix D.

3.3.2.3 The GLIMMERM system

The GLIMMERM algorithm uses the same IMM scoring method used in GLIMMER 2.0 and was developed specifically for eukaryotes having a gene density of less than 20%.

The splice site predictor algorithm in GLIMMERM [8] captures dependencies among neighboring bases in a small window around each splice junction (16 and 29 bp around the 5′-donor and 3′-acceptor sites, respectively) [35]. The algorithm takes advantage of the fact that the coding and non-coding sequences switch at the splice junction and detects this switch with two second-order Markov chains, one models coding sequence and another non-coding sequence. The length of each of these coding or non-coding context windows is currently fixed at 80 bp.

Potential coding regions are evaluated by a scoring function based on decision trees that estimate the probability that a DNA subsequence is coding. Subsequences are evaluated according to their putative type: intron, initial exon, internal exon, final exon, and single-exon gene. Each such subsequence is run through ten different decision trees built with the OC1 induction system [10] that can take multiple numeric feature values. The probabilities obtained with the decision trees are averaged to produce a smoothed estimate of the probability that the given subsequence is of a certain type. A putative gene model is then accepted only if the IMM score for the coding sequence in the correct reading frame exceeds a fixed threshold.

The main assumptions of this program are:

- The coding region of every gene begins with a start codon ATG,
- The gene has no in-frame stop codons except the very last codon, and

- Each exon is in a consistent reading frame with the previous exon.

These constraints significantly enhance the efficiency of computing the optimal gene models by restricting the search space of the dynamic programming algorithm. The dynamic programming algorithm processes sequences from left to right searching for stop codons. At each stop codon, it searches back in the 5′ direction (right to left) finding all possible genes using this stop codon, and chooses the highest scoring gene. The only positions that are considered as possible intron donor and acceptor sites are those that score above the threshold determined by the Markov chains. The algorithm is run separately on direct and complementary strands of the input. GLIMMERM rejects overlapping genes by going through the list of putative genes. Overlap occurs when two models share a common stop codon and have different exon locations. If the genes overlap by less than 30 bp (the default value), then the overlap is ignored and both are considered possible genes. If the overlap is more than 30 bp, then they are rescored using the IMM and a gene with the best score is retained.

GLIMMERM was used for a malaria parasite (*Plasmodium falciparum*) genome and showed the rates of sensitivity and specificity for nucleotide level recognition above 94% and 97%, respectively [37]. GLIMMERM's accuracy of 93% on a plant genome, *Arabidopsis thaliana* [37], was comparable to the accuracy of 95% and 94% for GeneMark.hmm and GenScan, respectively [37].

3.3.3 GenScan

GenScan is a general-purpose gene identification program used to analyze genomic DNA sequences from a variety of organisms including human, other vertebrates,

invertebrates, and plants [30, 38]. For each genomic sequence, the program determines the most likely gene structure under a probabilistic model of the gene structural and compositional properties of the genomic DNA for the given organism.

The probabilistic model used by GenScan accounts for many of the essential gene structural properties of genomic sequences: e.g., typical gene density, typical number of exons per gene, distribution of exon sizes for different types of exons; and also many of the important compositional properties of genes: e.g., the reading frame-specific hexamer composition of coding regions versus the reading frame-independent hexamer composition of introns and intergenic regions, and the position-specific composition of the translation initiation, termination signals, TATA box, cap site, and poly-adenylation signals. Importantly, novel models of the donor and acceptor splice sites are used, which capture potentially important dependencies between positions in these signals. For human and other vertebrate sequences, separate sets of model parameters are used, which account for the many differences in gene density and structure observed in genomic regions that exhibit distinct nucleotide composition (G+C%). GenScan has an additional feature that draws a representation of the resultant prediction showing all putative exons in their respective positions on both strands and whether they are leading, internal, or terminal, and a simplified scoring scheme.

GenScan uses a homogeneous 5^{th}-order Markov model of non-coding regions and three periodic 5^{th}-order Markov models of coding regions. The parameters are typically estimated using the maximum likelihood method, that is, by using the observed conditional frequencies obtained from an appropriate training set of known genes to estimate the corresponding conditional probabilities. Nucleotides are generated

according to the probabilistic rules derived from an underlying hidden Markov process. It is parameterized for G+C content. The training set containing exons and introns are divided into four categories depending on the G+C content of the sequence. The categories are: I (< 43% G+C), II (43-51% G+C), III (51-57% G+C), and IV (> 57% G+C). For each of these categories, separate initial state probabilities are computed by estimating the relative frequencies of various functional units in these categories. GenScan uses double-stranded models to allow for occurrences of multiple genes on either or both DNA strands unlike other programs, which analyze one strand at a time assuming the input sequence contains a single complete gene. The essential idea is that a precise probabilistic model a gene/genomic sequence looks like is specified in advance, and then, given a sequence, one determines which of the vast number of possible gene structures (involving any valid combination of states/lengths) has the highest likelihood given the sequence. It cannot handle overlapping transcription units and does not address alternative splicing.

GenScan program was designed primarily to predict genes in human/vertebrate genomic sequences; its accuracy level may be lower for other organisms. However, the vertebrate version of the program performed fairly well on an invertebrate (*Drosophila melanogaster*) sequences with accuracy per exon value of 68%. The maize and *Arabidopsis* versions (both are plants) also performed fairly well on their respective organisms with per exon accuracy of 78% and 67%, respectively. It differs from the majority of existing gene finding algorithms in that it allows partial genes as well as complete genes and the occurrence of multiple genes in a single sequence, on either or

both DNA strands. For prokaryotic or yeast sequences, the programs GLIMMER and/or GeneMark [34] are better in comparison to GenScan [30, 38].

3.3.4 Fungal gene predictions by GLIMMER, GLIMMERM, and GenScan

The gene prediction methods were chosen based on their ability to identify small genes. GLIMMER has been widely used for prokaryotes and is highly accurate for their gene detection. The program allows building models on any datasets. GLIMMERM, a modified version of GLIMMER, was used as it was developed mainly for eukaryotes with small genome sizes and can identify short genes from high density genomes. GenScan was chosen for a comparison purpose since it is a widely used prediction program.

- GLIMMER was trained using cDNA datasets of *N. crassa*, *S. cerevisiae*, and *S. pombe* obtained from NCBI as described in the section 3.1. The trained model was then used to extract putative genes from the *N. crassa* genomic sequences. Program *get-putative*, *extract*, *build-icm*, and *glimmer2* are all part of the GLIMMER package.

USAGE

- build-icm < tmp.train > tmp.model

  It builds the model using the training datasets in fasta format (tmp.train) and stores it in tmp.model

- glimmer2 Sequence tmp.model –g n | get-putative > g2.coord

  Using the trained model (tmp.model) and genomic sequence (Sequence), *glimmer2* predicts all possible gene locations. The most likely gene coordinates are extracted by

the program get-putative included in the GLIMMER package and stored in g2.coord. The –g option denotes the minimum gene length. 30 bp is used in this study.

- extract Sequence g2.coord > Nucleotide_Output

Using the stored coordinates (g2.coord), the ORFs are extracted from the genomic sequence (Sequence).

- GLIMMERM was run using pre-trained models of one filamentous fungus species (*Aspergillus fumigatus*) and two plant species (*Arabidopsis thaliana* and *Oryza sativa*) available from the GLIMMERM software package.

USAGE

- glimmerm Sequence -d directory of trained model > Output

- GenScan was run using pre-trained models of human and two plant species (*A. thaliana* and maize) available from the downloaded GenScan software package.

USAGE

- genscan Parameter_file_of_organism Sequence -cds > Output

The program takes in a parameter file of the trained model and a genomic sequence file (Sequence), and outputs the predicted ORFs. The –cds option prints the predicted nucleotide sequences.

Chapter 4 Database Implementation

The database was developed in order to compile information extracted from fungal genomes and other genomic information used in gene prediction methods. The database stores entry files of fungal sequences, results of various sequence analyses (e.g., base composition), results from similarity search against various databases, and the results using various gene prediction programs. This database system was developed to facilitate the future development of fungal specific gene prediction methods. It will help us understand the fungal genome specific properties.

The database consists of three parts: i) nucleotide sequence information of the three fungal genomes (*N. crassa*, *S. cerevisiae*, and *S. pombe*), ii) *N. crassa* protein sequence information including similarity search results against the entire NCBI Non-Redundant database, and iii) *N. crassa* protein similarity search results against five fungal genome databases. The overview of the database architecture is presented in Figure 4.1. MySQL was used to construct the database and PHP was used to build the web interface. The detailed descriptions for each table and the web interface are given next.

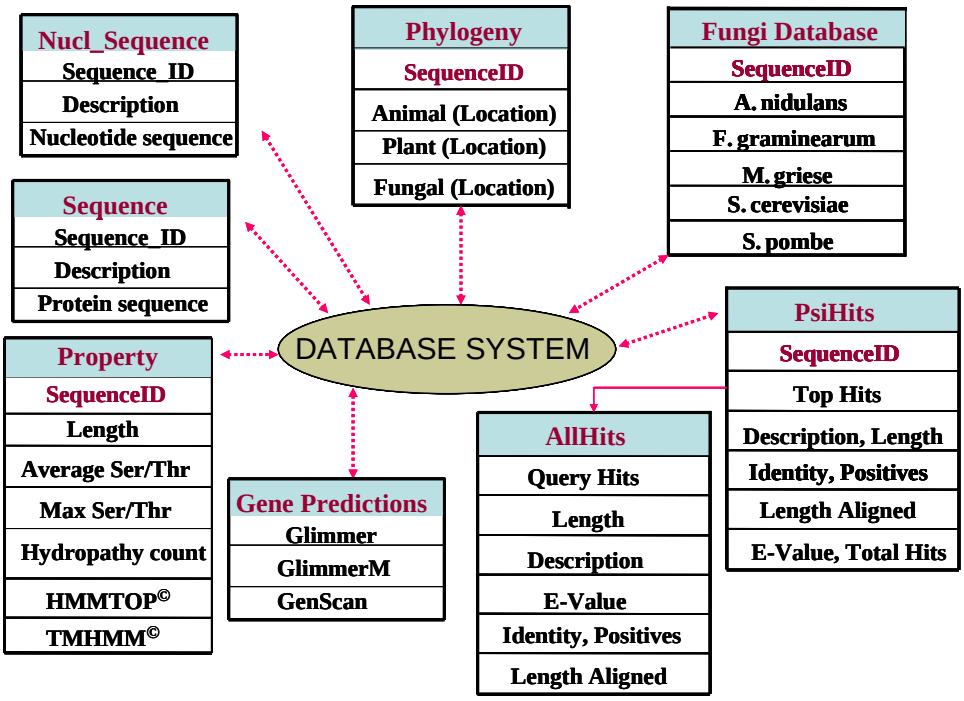


Figure 4.1 Overview of the database architecture.

4.1 SQL tables

The database contains eight tables. The query sequences are 10,082 of the *N. crassa* ORFs annotated by the Whitehead genome project.

1. **Sequence** contains protein sequence information on the entire *N. crassa* ORFs. It has three fields: sequence ID, description of the sequence, and protein sequence.
2. **Nucl_Sequence** contains nucleotide sequence information on the *N. crassa* ORFs. It has three fields: sequence ID, description of the sequence, and nucleotide sequence.
3. **PsiHits** contains information on the PSI-BLAST protein similarity search results against the NCBI NR database. The fields include: sequence ID, PSI-BLAST output file, sequence length (in amino acids), NCBI accession number of the top hit, E-value of the top hit, description of the top hit sequence, length of the hit sequence, number of identical amino acids aligned between the *N. crassa* query and top hit sequences,

number of (biochemically) similar amino acids aligned between the *N. crassa* query and top hit sequences, and length of the *N. crassa* query that aligned with the top hit. The information can be accessed using the sequence IDs (accession numbers of *N. crassa* ORFs), which is the primary key.

4. **AllHits** contains information on the entire hit list in the PSI-BLAST protein similarity search results. The fields include: sequence ID, amino acid length of the *N. crassa* query sequence, entire hits in the PSI-BLAST similarity search, their E-values, descriptions of the hit sequences, amino acid lengths of the hit sequences, number of identical amino acids aligned between the *N. crassa* query and each hit sequences, number of similar amino acids aligned between the *N. crassa* query and each hit sequences, and length of the *N. crassa* query that aligned with each hit. The information can be accessed using sequence IDs (the accession numbers of *N. crassa* ORFs), which is the primary key. Currently the database supports the first 100 hits with significant E-values (< 0.005).
5. **Property** contains information on: the *N. crassa* ORF accession number, sequence length in amino acids, average Serine/Threonine %, maximum Serine/Threonine % in a window of 100 amino acids, and numbers of transmembrane regions predicted by Kyte-Doolittle hydropathy plot, HMMTOP, and TMHMM.

   The program for calculating the Kyte-Doolittle hydropathy values was written locally using the window size of 16 amino acids and the average cut-off hydropathy value of -0.05. The information can be accessed using the *N. crassa* ORF accession number, which is the primary key.

6. **Phylogeny** contains information on: the *N. crassa* ORF accession number, accession number of the first hits of PSI-BLAST search that belong to animal, plant, and fungal kingdoms, their locations in the PSI-BLAST hit list, and whether there is any hits to each of the three kingdoms. The information can be accessed using the *N. crassa* ORF accession number, which is the primary key.

   Identifying the kingdom classification of PSI-BLAST hit sequences is done as follows. The accession numbers of the animal, plant, and fungal kingdoms were downloaded from the NCBI web site (http://www.ncbi.nlm.nih.gov). The accession numbers of PSI-BLAST hits were searched against this list of each kingdom. The search continues until there is a hit with any of the three kingdom lists. When there is no hit within the three kingdom lists, it can either mean that there was really no hit to organisms in the three kingdoms (the sequence could belong to prokaryotes, for example) or the taxonomy list does not contain the particular accession number. In either case, the kingdom identification field was kept empty.

7. **FungiDatabase** contains information on the PSI-BLAST protein similarity search results against the five fungal databases: *Aspergillus nidulans, Fusarium graminearum, Magnaporthe grisea, Saccharomyces cerevisiae*, and *Schizosaccharomyces pombe*. The fields include: the *N. crassa* ORF accession number, sequence length in amino acids, and whether there is any hit to each of the five fungal databases. The count of significant hits within the 0.005 E-value thresholds to each of the five fungal databases is also included. The information can be accessed using the *N. crassa* ORF accession number, which is the primary key.

8. **Gene Predictions** contains the ORFs predicted by prediction methods used against the *N. crassa* genomic sequences (described in chapter 5). Currently this table is not active and will be updated later.

4.2 Interface and queries

The interface provides various ways to access fungal genome information extracted by various sequence analyses. The stored information can be accessed using two primary web pages.

1. **PSI-BLAST Result Query** page serves as the cover page of the database. It is shown in Figure 4.2. It contains a brief description and the links to the other query pages.

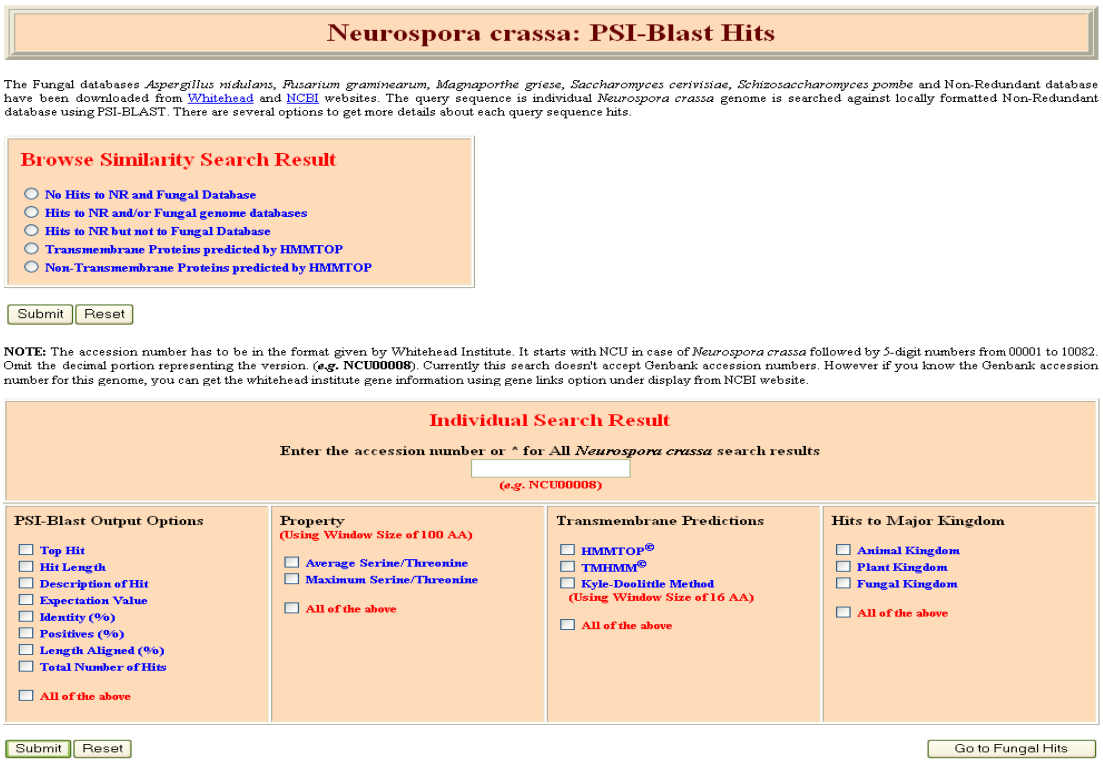


Figure 4.2 Screen image of the database cover page

The cover page has two query sections: *Browse Similarity Search Results* and *Individual Search Result*. The *Browse Similarity Search* section (Figure 4.3) presents the PSI-BLAST search results for the entire *N. crassa* ORFs in several groups. Figure 4.4 shows the output for one of the groups, "No Hits to NR and Fungal Database".

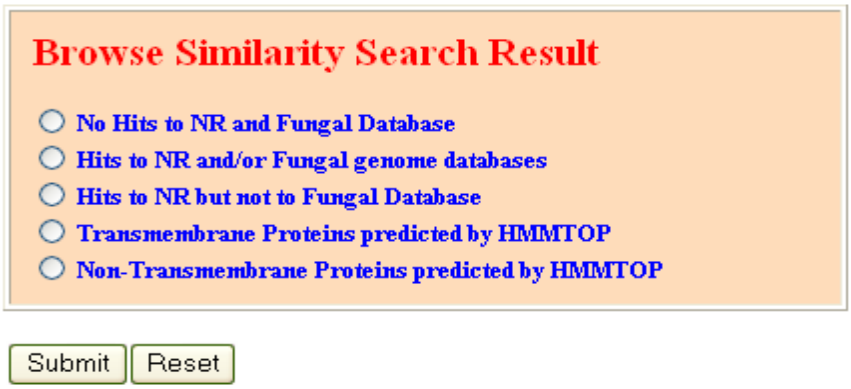


Figure 4.3 *Browse Similarity Search Result* section. The PSI-BLAST results are grouped based on whether there is any hit against the NCBI NR database or any one of the five fungal databases, and also whether predicted as transmembrane proteins.

Close the Window | Go back

PSI-BLAST results for the *Neurospora crassa* ORF group

Neurospora crassa ORF group: No Hits to any Database. Total Hits = 2211

SequenceID	Length of Hit (AA)	Avg Ser/Thr	Max Ser/Thr	HMMTOP count	TMHMM count	Kyle-Doolittle count
NCU00001	54	11.11	0	1	0	0
NCU00002	45	15.56	0	0	0	0
NCU00009	96	28.12	0	0	0	0
NCU00011	100	17	17	0	0	1
NCU00012	43	9.3	0	0	0	0
NCU00013	114	10.53	9	1	0	1
NCU00016	9	11.11	0	0	0	0
NCU00020	47	17.02	0	0	0	1
NCU00031	75	32	0	0	0	0
NCU00036	55	18.18	0	0	0	0
NCU00044	26	19.23	0	0	0	0
NCU00047	175	14.86	19	0	0	2

Figure 4.4 PSI-BLAST results presented when the option "No Hits to NR and Fungal Database" is chosen.

The *Individual Search Result* section (Figure 4.5) allows users to examine the PSI-BLAST similarity search results for each *N. crassa* ORF in detail.

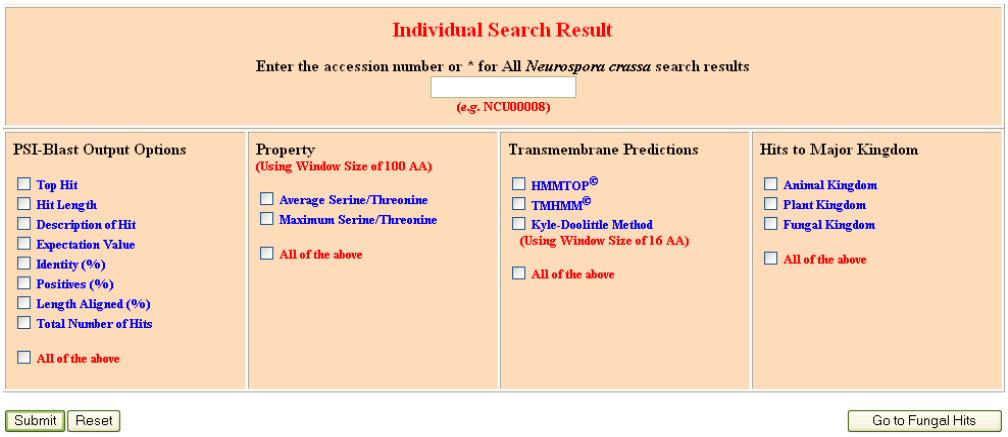


Figure 4.5 *Individual Search Result* section. The user can choose to browse the entire *N. crassa* search results in a spreadsheet format by giving "*" instead of an accession number to display customized information on an individual *N. crassa* ORF using its accession number.

Database Hits for Sequence: NCU00005 (Click the link for sequence)

PSI-Blast Results

Top Hit	NP_499588.1
Length of Hit (AA)	465
Description	Putative plasma membrane membrane protein, with at least 8 transmembrane domains, a coiled coil-4 domain, of eukaryotic origin (53.6 kD) [Caenorhabditis elegans] pir||T27415 hypothetical protein Y75B8A.16 - Caenorhabditis elegans emb|CAA22103.2| Hypothetical protein Y75B8A.16 [Caenorhabditis elegans]
E-Value	0
Identity (%)	20
Positives (%)	40
Aligned (%)	82
Total Hits	23

Accession Nos of Hits to Major Kingdom

Animal	NP_499588.1
Plant	NP_194493.2
Fungi	NP_011945.1

Transmembrane Prediction

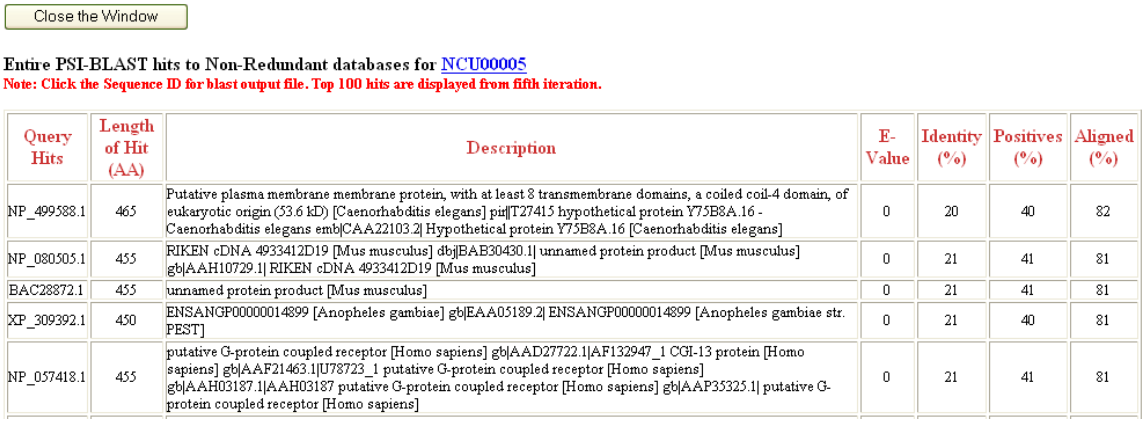
HMMTOP TM count	9
TMHMM TM count	9
Kyle-Doolittle count	8

Property

Average Serine/Threonine	17.57
Maximum Serine/Threonine	33

Figure 4.6 PSI-BLAST result page for an individual *N. crassa* ORF (NCU00005; all options in the Figure 4.5 were chosen). The protein and DNA sequences for this ORF can be obtained from the link to the accession number.

The link to the top hit accession number (NP_499588.1 in Figure 4.6) opens the table containing upto 100 hits of the PSI-BLAST search as shown in Figure 4.7.



Close the Window

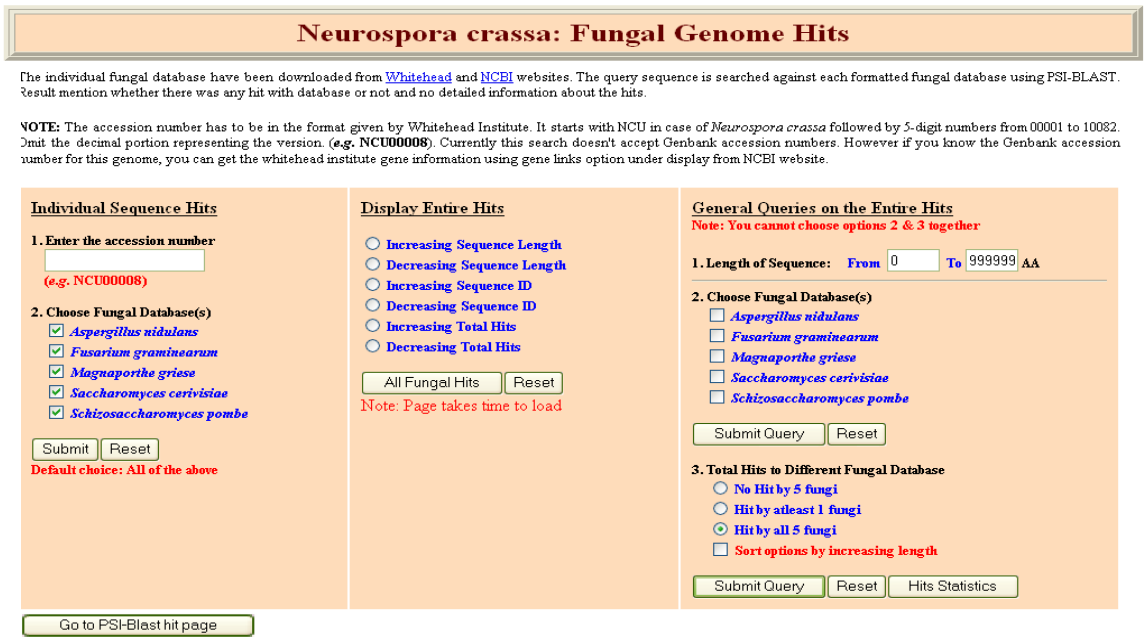
Entire PSI-BLAST hits to Non-Redundant databases for NCU00005

Note: Click the Sequence ID for blast output file. Top 100 hits are displayed from fifth iteration.

Query Hits	Length of Hit (AA)	Description	E-Value	Identity (%)	Positives (%)	Aligned (%)
NP_499588.1	465	Putative plasma membrane membrane protein, with at least 8 transmembrane domains, a coiled coil-4 domain, of eukaryotic origin (53.6 kD) [Caenorhabditis elegans] pir||T27415 hypothetical protein Y75B8A.16 - Caenorhabditis elegans emb|CAA22103.2| Hypothetical protein Y75B8A.16 [Caenorhabditis elegans]	0	20	40	82
NP_080505.1	455	RIKEN cDNA 4933412D19 [Mus musculus] dbj|BAB30430.1| unnamed protein product [Mus musculus] gb|AAH10729.1| RIKEN cDNA 4933412D19 [Mus musculus]	0	21	41	81
BAC28872.1	455	unnamed protein product [Mus musculus]	0	21	41	81
XP_309392.1	450	ENSANGP00000014899 [Anopheles gambiae] gb|EAA05189.2| ENSANGP00000014899 [Anopheles gambiae str. PEST]	0	21	40	81
NP_057418.1	455	putative G-protein coupled receptor [Homo sapiens] gb|AAD27722.1|AF132947_1 CGI-13 protein [Homo sapiens] gb|AAF21463.1|U78723_1 putative G-protein coupled receptor [Homo sapiens] gb|AAH03187.1|AAH03187 putative G-protein coupled receptor [Homo sapiens] gb|AAP35325.1| putative G-protein coupled receptor [Homo sapiens]	0	21	41	81

Figure 4.7 Display of all hits of PSI-BLAST search for an individual *N. crassa* ORF, NCU00005.1.

Fungal Database Query page provides some tools to view the results of PSI-BLAST similarity search for *N. crassa* ORFs against the five fungal genomic databases. It has three sections: *Individual Sequence Hits*, *Display Entire Hits*, and *General Queries on the Entire Hits* (Figure 4.8).



Neurospora crassa: Fungal Genome Hits

The individual fungal database have been downloaded from Whitehead and NCBI websites. The query sequence is searched against each formatted fungal database using PSI-BLAST. Result mention whether there was any hit with database or not and no detailed information about the hits.

NOTE: The accession number has to be in the format given by Whitehead Institute. It starts with NCU in case of *Neurospora crassa* followed by 5-digit numbers from 00001 to 10082. Omit the decimal portion representing the version. (***e.g.* NCU00008**). Currently this search doesn't accept Genbank accession numbers. However if you know the Genbank accession number for this genome, you can get the whitehead institute gene information using gene links option under display from NCBI website.

Individual Sequence Hits

1. Enter the accession number
(*e.g.* NCU00008)
2. Choose Fungal Database(s)
- *Aspergillus nidulans*
- *Fusarium graminearum*
- *Magnaporthe griese*
- *Saccharomyces cerivisiae*
- *Schizosaccharomyces pombe*

Submit | Reset
Default choice: All of the above

Display Entire Hits

- Increasing Sequence Length
- Decreasing Sequence Length
- Increasing Sequence ID
- Decreasing Sequence ID
- Increasing Total Hits
- Decreasing Total Hits

All Fungal Hits | Reset
Note: Page takes time to load

General Queries on the Entire Hits

Note: You cannot choose options 2 & 3 together

1. Length of Sequence: From 0 To 999999 AA
2. Choose Fungal Database(s)
- *Aspergillus nidulans*
- *Fusarium graminearum*
- *Magnaporthe griese*
- *Saccharomyces cerivisiae*
- *Schizosaccharomyces pombe*

Submit Query | Reset

3. Total Hits to Different Fungal Database
- No Hit by 5 fungi
- Hit by atleast 1 fungi
- Hit by all 5 fungi
- Sort options by increasing length

Submit Query | Reset | Hits Statistics

Go to PSI-Blast hit page

Figure 4.8 Fungal Database Query page.

The *Individual Sequence Hits* section has the options to choose from the hits to the five fungal databases. The default option is to choose all of the five fungal databases. Figure 4.9 shows the result page for NCU00050. Each "Yes" in the table is linked to the list of PSI-BLAST hits for the given fungal genomic database.

Database Hits for Sequence: NCU00050 (Click the link for sequence)

Aspergillus nidulans	*Fusarium graminearum*	*Magnaporthe griese*	*Saccharomyces cerivisiae*	*Schizosaccharomuces pombe*
Yes	Yes	-	Yes	Yes

Figure 4.9 Fungal database hits page for an individual *N. crassa* ORF, NCU00050.

The *Display Entire Hits* section (Figure 4.7) has options to display all of the five fungal genome search results with a customized sorting method based on increasing or decreasing *N. crassa* sequence lengths, increasing or decreasing *N. crassa* sequence IDs, or increasing or decreasing total number of hits (Figure 4.10).

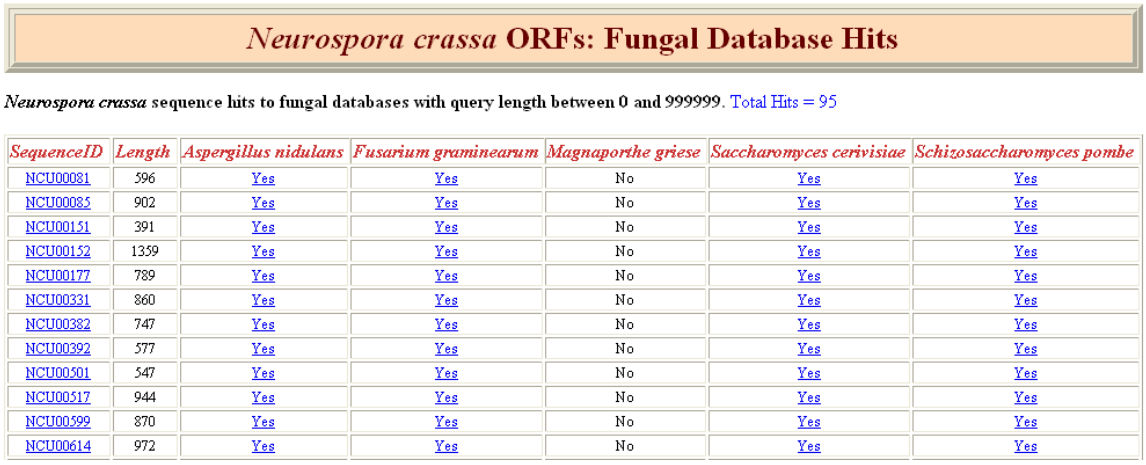
Neurospora crassa: Fungal Hits

Neurospora crassa Sequence Hits: Sorted by Decreasing Sequence Length

SequenceID	*Length*	*Aspergillus nidulans*	*Fusarium graminearum*	*Magnaporthe griese*	*Saccharomyces cerivisiae*	*Schizosaccharomyces pombe*
NCU07119	5141	Yes	Yes	Yes	Yes	Yes
NCU06468	4992	Yes	Yes	Yes	Yes	Yes
NCU06976	4367	Yes	Yes	Yes	Yes	Yes
NCU08501	4065	Yes	Yes	Yes	Yes	Yes
NCU00658	4007	Yes	Yes	Yes	Yes	Yes
NCU01379	3941	Yes	Yes	Yes	Yes	Yes
NCU00625	3409	Yes	Yes	Yes	Yes	Yes
NCU07702	3287	Yes	Yes	Yes	Yes	No
NCU05837	3209	Yes	Yes	Yes	Yes	Yes
NCU06294	3163	Yes	Yes	Yes	No	No
NCU00459	3054	Yes	Yes	Yes	Yes	Yes
NCU00949	3041	No	Yes	Yes	No	Yes
NCU05047	2992	Yes	Yes	Yes	Yes	Yes
NCU00274	2953	Yes	Yes	Yes	Yes	Yes
NCU05132	2898	Yes	Yes	No	No	No

Figure 4.10 Fungal database hits sorted by decreasing *N. crassa* sequence length. Only one sorting option can be chosen.

The *General Queries on Entire Hits* section provides more options to customize the display of the PSI-BLAST search results of the entire *N. crassa* ORFs against the five fungal databases (Figure 4.11).



***Neurospora crassa* ORFs: Fungal Database Hits**

Neurospora crassa **sequence hits to fungal databases with query length between 0 and 999999.** Total Hits = 95

SequenceID	*Length*	*Aspergillus nidulans*	*Fusarium graminearum*	*Magnaporthe griese*	*Saccharomyces cerivisiae*	*Schizosaccharomyces pombe*
NCU00081	596	Yes	Yes	No	Yes	Yes
NCU00085	902	Yes	Yes	No	Yes	Yes
NCU00151	391	Yes	Yes	No	Yes	Yes
NCU00152	1359	Yes	Yes	No	Yes	Yes
NCU00177	789	Yes	Yes	No	Yes	Yes
NCU00331	860	Yes	Yes	No	Yes	Yes
NCU00382	747	Yes	Yes	No	Yes	Yes
NCU00392	577	Yes	Yes	No	Yes	Yes
NCU00501	547	Yes	Yes	No	Yes	Yes
NCU00517	944	Yes	Yes	No	Yes	Yes
NCU00599	870	Yes	Yes	No	Yes	Yes
NCU00614	972	Yes	Yes	No	Yes	Yes

Figure 4.11 List of *N. crassa* ORFs that have hits to selected fungal genomes. All fungi except *Magnaporthe grisea* were chosen from the "Choose the Fungal Databases" options.

It has an option to filter the list based on the *N. crassa* ORF lengths. The third subsection of the *General Query* section allows users to display results based on: No Hit by 5 fungi, Hit by at least 1 fungi and Hit by all 5 fungi. These results are by default sorted according to the *N. crassa* ORF accession numbers. An option is provided to sort them based on the increasing length. There is also a button to display the summary statistics. The statistics page view is shown in Figure 4.12.

Summary Statistics for *Neurospora crassa* ORF Hits against Fungal Genome Databases

Number of Fungal Genomes	Number of *N. crassa* ORFs
No hit	2372
One	467
Two	668
Three	1557
Four	1140
Five	3878

Single Database Hits Statistics

Database	*Count*
Aspergillus nidulans	64
Fusarium graminearum	168
Magnaporthe griese	182
Saccharomyces cerivisiae	34
Schizosaccharomyces pombe	19
Total Single Hits	**467**

Double Database Hits Statistics

Database	*A. nidulans*	*F. graminearum*	*M. griese*	*S. cerevisiae*	*S. pombe*
A. nidulans	-	134	63	7	7
F. graminearum	134	-	408	8	9
M. griese	63	408	-	11	16
S. cerevisiae	7	8	11	-	5
S. Pombe	7	9	16	5	-

Triple Database Hits Statistics

A. nidulans	*F. graminearum*	*M. griese*	*S. cerevisiae*	*S. pombe*	Total Hits
-	-	X	X	X	6
-	X	-	X	X	13
-	X	X	-	X	60
-	X	X	X	-	43
X	-	-	X	X	2
X	-	X	-	X	15
X	-	X	X	-	9
X	X	-	-	X	44
X	X	-	X	-	29
X	X	X	-	-	1336

Quadruple Database Hits Statistics

Hits to Others except	*Count*
Aspergillus nidulans	82
Fusarium graminearum	22
Magnaporthe griese	95
Saccharomyces cerivisiae	592
Schizosaccharomyces pombe	349
Total Quadruple Hits	**1140**

Figure 4.12 Summary Statistics of N. crassa ORF hits against the five fungal databases. The summary statistics table shows the number of fungal genomes *N. crassa* ORFs have any hit against.

The interfaces to view other sequence information (e.g., base frequencies) and gene predictions are not yet implemented.

The database is currently accessible at the following URL:

http://bioservdb.unl.edu/~skanth/psiblast.php

Chapter 5 Results and Discussion

5.1 Fungal genome analysis at the protein level

The complete sets of predicted proteins from three filamentous fungal genomes (*Neurospora crassa, Aspergillus nidulans*, and *Fusarium graminearum*) were analyzed. The protein sequence data were downloaded from the Whitehead Institute web site. They included 10,082 proteins for *N. crassa*, 9,541 for *A. nudulans*, and 11,640 for *F. graminearum*. In order to identify any possible functions for the predicted proteins, a series of PSI-BLAST similarity search was conducted on the NCBI Non-Redundant (NR) protein database using each of the predicted proteins from *A. nidulans*, *F. graminearum*, and *N. crassa* as a query. The database hits were filtered using the E-value cut off of 0.005. Each of the top hits was further identified if it belongs to the animal, plant, or fungal kingdom. PSI-BLAST similarity search was also done against each of the five fungal genome databases from *N. crassa, A. nidulans*, *M. grisea*, *F. graminearum*, *S. cerevisiae*, and *S. pombe* to find any similar sequences from other fungal genomes. Whereas any significant similarity will give us a clue for the possible protein functions, if there is no hit to the database sequence, it implies uniqueness of these proteins (genes) to the fungal species.

In order to identify possible transmembrane proteins, transmembrane region predictions were done by HMMTOP, TMHMM, and Kyte-Doolittle methods. The numbers of identified transmembrane regions were often different among the three prediction methods. Although the numbers of identified regions were not the same between HMMTOP and TMHMM, the total number of proteins predicted to have one or

more transmembrane regions were very close. We decided to use prediction based on HMMTOP to classify the proteins as transmembrane (TM) and non-transmembrane (NonTM) proteins.

As described in Chapter 3, cell wall proteins are found to have high Ser/Thr contents. In this project, any protein regions longer than 100 aa that have Ser/Thr content higher than 20% were identified, and such proteins were considered as the candidates for cell wall proteins.

Table 5.1 summarizes the results of the PSI-BLAST similarity search and other protein sequence analyses. Around 10% of *A. nidulans* and *F. graminearum* sequences, and about 25% of *N. crassa* sequences were unique to the species (species specific); i.e., there was no hit to any other organisms including other fungal species. Further analysis showed that of the *N. crassa* specific 2,211 proteins, 937 were shorter than 100 aa. Such numbers were much smaller for *F. graminearum* and *A. nidulans*. In the case of the 1,182 *F. graminearum* specific proteins, only 80 were shorter than 100 aa, whereas only 77 were shorter among the 826 *A. nidulans* specific proteins. These species-specific sequences will become candidates for further experimental and in depth analysis. All three genomes had less than 2% of hits only by animals and/or plants but not by any fungus species. These genes are also of our future research interests. Why these genes do not have any homologue in other fungi but in animals/plants, and where and how they come from, are of great interest.

Table 5.1 Summary of similarity search against the NR database and fungal genomes

Genome (Total no of proteins)	**Category**	No of NonTM proteins[1]	No of TM proteins[1]
A. nidulans (9,541)	All	4,273	5,268
	Species specific[2]	604	222
	Fungal hits except *S. cerevisiae*[3]	1,863	1,330
	S. cerevisiae hit[4]	2,703	2,688
	Hit except fungi[5]	98	33
	Ser+Thr >= 20%[6]	2,723	2,536
F. graminearum (11,640)	All	6,712	4,928
	Species specific[2]	851	331
	Fungal hits except *S. cerevisiae*[3]	2,613	1,740
	S. cerevisiae hit[4]	3,133	2,799
	Hit except fungi[5]	115	58
	Ser+Thr >= 20%[6]	2,701	3,218
N. crassa (10,080)	All	3,990	6,092
	Species specific[2]	1,759	452
	Fungal hits except *S. cerevisiae*[3]	1,779	1,338
	S. cerevisiae hit[4]	2,437	2,156
	Hit except fungi[5]	117	44
	Ser+Thr >= 20%[6]	2,473	2,816

[1] Each protein was grouped as nontransmembrane (NonTM) or transmembrane (TM) proteins.
[2] No significant PSI-BLAST hit based on E-value threshold = 0.005 to any database sequence
[3] Significant PSI-BLAST hits found against any fungal databases except the *S. cerevisea* genome.
[4] Significant PSI-BLAST hits found only against the *S. cerevisea* genome.
[5] Significant PSI-BLAST hits found against animal and/or plant sequences, but no fungal genome hit.
[6] Number of proteins that have the Ser + Thr content higher than or equal to 20 %.

All of the three fungal genomes had high proportions of hits to any *S. cerevisiae* proteins. The numbers of *N. crassa*, *F. graminearum*, and *A. nidulans* proteins that have any hits to *S. cerevisiae* proteins were 5,391, 5,932, and 4,593, respectively. This accounts for about 50-60% of the entire proteins present in each species. More notably,

the remaining 40-50% of these fungal proteins does not share similarities with any *S. cerevisiae* proteins. Note that the three fungal species used for this study are all filamentous fungi, whereas *S. cerevisiae* is a non-filamentous fungus. *Saccharomyces cerevisiae* was the first fungal and the first eukaryotic genome that was completely sequenced, and has been used as representing the fungal kingdom genomes. However, the results described above clearly show that there might be a bias in using the *S. cerevisiae* genome to represent the entire fungal species. This was indeed one of the main reasons in this study to create species-specific database and gene prediction.

Sequences are almost equally split into transmembrane (TM) and non-transmembrane (NonTM) proteins. There are more NonTM proteins for fungal species specific proteins; all three species have three-fold or more of NonTM proteins. This latter ratio is more reasonable as the majority of proteins must be required for performing other than membrane-related cell functions. In general, 25% of the proteins are expected to be TM proteins in eukaryotic genomes. It implies that the over representation of TM proteins (50% or more) from the entire protein sets may be due to over prediction by the transmembrane prediction program used (HMMTOP).

5.2 Gene prediction by GLIMMER, GLIMMERM, and GenScan

As described in Chapter 3, many fungal genomes have relatively high gene density and contain many short genes that cannot be effectively detected by commonly used gene prediction programs. From many available gene prediction methods, three programs: GLIMMER, GLIMMERM and GenScan were chosen in this study, and their prediction performance was examined against the currently available genome

annotations. GLIMMER and GLIMMERM were chosen based on their possible ability to detect small coding regions. GLIMMER is the most hopeful even though it is developed for prokaryotic genomes as it can detect small ORFs and the program can be altered to have a minimum gene size. GLIMMERM, although it may underpredict splicing sites, hardly misses a gene completely. Thus this method might be also helpful to identify genes of small size. GenScan was added for a comparison purpose.

5.2.1 Gene prediction by GLIMMER

In order to train gene prediction methods, a set of genes with experimentally confirmed is required. It is ideal if such training dataset was obtained from the organism under the investigation. For this reason, experimentally confirmed cDNA sequences were obtained for three fungal species from the NCBI database as described in Chapter 3. The three datasets include: 120 *N. crassa*, 157 *S. cerevisiae*, and 613 *S. pombe* cDNA sequences. The larger dataset used by the prediction program FGENESH mentioned in Chapter 3 was not used as the aim of this study was to identify possible new genes using experimentally verified cDNA sequences. The dataset used by FGENESH includes a larger number of cDNA sequences that might not be experimentally confirmed. This will provide a way to verify already annotated genes by the Whitehead genome project, and to identify any new gene candidates independently.

GLIMMER 2.0 was trained using these three sets of cDNA sequences, and an interpolated Markov model was created for each of the three species. Trained GLIMMER 2.0 programs were used on the complete *N. crassa* genomic sequence obtained from the Whitehead website (including 821 contig sequences). Table 5.2 summarizes the analysis of GLIMMER prediction compared to the Whitehead annotation.

Table 5.2 GLIMMER predictions for the *N. crassa* genome.

Training set	No of Whitehead predicted ORFs	No of GLIMMER predicted ORFs	Predicted new ORFs		Exact match with Whitehead prediced ORFs	No hit with *M. grisea* genome	No hit with NR database
			Total >= 10aa	< 100aa & >= 10aa			
N. crassa	10,082	34,008	70	68	746	53	59
S. cerevisiae	10,082	44,087	320	314	582	27	216
S. pombe	10,082	46,191	437	428	598	209	390

The program predicted 34,008 possible ORFs when trained with the *N. crassa* cDNA dataset. This number is much larger than the one found in the Whitehead genome project, 10,082 ORFs. This is expected since GLIMMER (developed for prokaryotic genomes) does not predict exon-intron structures, and all predicted ORFs are considered as uninterrupted single exon gene. Therefore, multi-exon genes are likely to be recognized as multiple small genes by GLIMMER. Considering the average number of exons in *N. crassa* is 2.7, approximately four times higher numbers of predicted ORFs by GLIMMER seems to be reasonable.

In order to identify any new predictions compared to the Whitehead annotations, predicted ORFs were used as queries for blastn DNA similarity search against the Whitehead predicted *N. crassa* ORF set. 70 ORFs predicted by GLIMMER were not overlapped with any Whitehead annotated ORFs. After excluding very short (shorter than 10 aa) ORFs, there were still 68 new short gene candidates (shorter than 100 aa but longer than 9 aa). Although only less than 2% was the exact match with Whitehead predicted ORFs as shown in Table 5.2, as described above, this is not surprising since GLIMMER does not recognize multi-exon genes.

In order to check if any of these new short gene candidates have been already known in another fungal species, these new ORFs were used as queries for PSI-BLAST against the NCBI NR database and blastn against the *M. grisea* genomic database. *M. grisea* was used because this is one of the closest fungal species to *N. crassa* (See Figure 1.1). The default parameters were used for both blastn and PSI-BLAST searches. As shown in Table 5.2, the majority of the new short gene candidates did not have any similar sequence either in the NCBI NR database or in the *M. grisea* genome. These "no hit" ORFs could be false positives or actual new unique genes identified for the first time. On the other hand, 17 ORFs shared similarities with *M. grisea* ORFs. It is possible that these are real genes that have been missed by other prediction methods including the Whitehead genome project. Further experimental analysis is necessary to confirm these short gene candidates.

GLIMMER was also trained on other two fungal cDNA datasets: *S. cerevisiae* and *S. pombe*. These two trained programs predicted even more ORFs from the *N. crassa* genome, and shorter new gene candidates were identified (see Table 5.2). It is interesting that more short new gene candidates shared similarities with *M. grisea* when GLIMMER was trained based on *S. pombe*, which is non-filamentous fungus as is *S. cerevisiae*. It should be noted that the *S. pombe* training set included more cDNA samples (613) than the *N. crassa* set (120). Although the species is different, the *S. pombe* training set may have had a better representation of the actual genes in the *N. crassa* genome. The overall percentage of newly predicted genes that did not have a hit to any database was higher for *N. crassa* (85%) and *S. pombe* (89%) compared to *S. cerevisiae* (68%) trained results.

The new genes predicted by different models need to be examined further if predictions by different methods overlap.

5.2.2 Gene prediction by GLIMMERM

Due to technical problems and time constrains, GLIMMERM could not be trained using the three cDNA data sets used for training GLIMMER. Instead, the default models created with one filamentous fungal species (*Aspergillus fumigatus*) and two plant species (*Arabidopsis thaliana* and *Oryza sativa*) were used against the *N. crassa* genomic sequence. As shown in Table 5.3, GLIMMERM predicted 39,386, 26,058, and 20,608 possible ORFs using *A. fumigatus*, *A. thaliana*, and *O. sativa* models, respectively. The numbers of predicted ORFs were smaller than those of GLIMMER. This is expected since GLIMMERM (developed for small eukaryotic genomes) should be able to identify multi-exon genes.

Table 5.3 GLIMMERM predictions for the *N. crassa* genome.

Training set	No of Whitehead predicted ORFs	No of GLIMMERM predicted ORFs	Predicted new ORFs		Exact match with Whitehead prediced ORFs	No hit with *M. grisea* genome	No hit with NR database
			Total >= 10aa	< 100aa & >= 10aa			
A. fumigatus	10,082	39,386	82	82	1,952	45	79
A. thaliana	10,082	26,058	83	82	881	44	73
O. sativa	10,082	20,608	37	37	2,012	27	32

The program predicted between 500 to 2,500 new ORFs depending on the training set, and the majority of these new ORFs were shorter than 100 aa. However, many of these new ORFs were also shorter than 10 aa. After rejecting these very short ORFs the numbers of new short ORF candidates became comparable to those obtained by

GLIMMER (83 ORFs or fewer as shown in Table 5.3). Around 16 % of the new ORFs shorter than 100 aa were longer than 10 aa, and predictions exactly matching with the Whitehead annotations were between 8-20%, higher rates than GLIMMER's. It was expected that the model trained with *A. fumigatus*, a filamentous fungus, would perform better than other two plant models. However, the results showed that all the three models had similar performance based on the numbers of predictions.

5.2.3 Gene prediction by GenScan

GenScan could not be trained using the three cDNA training sets because the package does not provide the capability to build our own training models. Therefore, their default models created with human and two plant species (*A. thaliana* and maize) were used against the *N. crassa* genomic sequences. As shown in Table 5.4, the program predicted 7,994, 8,094, and 7,345 possible ORFs, with *A. thaliana*, maize, and human models, respectively. These prediction numbers were much smaller than those obtained by the other two prediction methods, and even smaller than the number given by the Whitehead prediction.

Table 5.4 GenScan predictions for the *N. crassa* genome.

Training set	No of Whitehead predicted ORFs	No of GenScan ORFs	Predicted new ORFs		Exact match with Whitehead prediced ORFs	No hit with *M. grisea* genome	No hit with NR database
			Total >= 10aa	< 100aa & >= 10aa			
A. thaliana	10,082	7,994	3	3	828	0	1
maize	10,082	8,094	4	4	808	4	3
human	10,082	7,345	3	3	1145	2	1

GenScan with any of the three training sets predicted very few new ORFs. The prediction using human as the training model seemed to perform slightly better than the other two models based on the number of predictions exactly matching with the Whitehead prediction. Overall about 10% of predicted ORFs agreed exactly with those predicted by Whitehead. Based on the data in Table 5.4, we can see that GenScan with the three models used does not seem to work well for the *N. crassa* gene prediction. The program needs to be trained with datasets obtained from the family closest to the genome under the study. However, even if a plant species, *O. sativa*, was used to train the model, GLIMMERM seemed to work better than GenScan. GenScan was developed primarily for larger and more complex eukaryotic genome annotation, and used mainly for human, mouse, and other vertebrates. These genomes contain much larger genes with longer introns compared to fungal genomes. Such differences in model architecture may explain the prediction difference among the three gene prediction methods.

5.2.4 Comparison of predictions among the three methods.

Of the three programs used for prediction, GLIMMER and GLIMMERM are better than GenScan. GenScan predicted fewer new genes and the number of predicted ORFs were smaller than predicted by the Whitehead. GLIMMER and GLIMMERM had few ORFs common between their predicted ORF sets. New ORFs predicted by GLIMMER using *N. crassa*, *S. cerevisiae*, and *S. pombe* training sets that were not also predicted by GLIMMERM were 47, 40, and 146, respectively. Similarly GLIMMERM had 71, 73, and 32 ORFs that were not also predicted by GLIMMER, with *A. fumigatus*, *A. thaliana*, and *O. sativa* models, respectively. All the new ORFs predicted by GenScan were distinct among the different training datasets. It would be interesting to see how

GLIMMERM performs if it is trained with cDNA datasets used by GLIMMER in this study. Better performance comparison is possible if different prediction programs are trained on the same datasets.

5.3 Nucleotide composition and gene feature analysis

As described in Chapter 3, gene prediction methods build their gene models based on the information present in the training set. In order to identify and understand any species-specific properties useful for gene prediction from fungal genomes, nucleotide frequencies and other gene properties were examined from the three fungal genomes: *N. crassa*, *S. cerevisiae*, and *S. pombe*. This analysis would reveal any common or different bias among these fungal genomes.

The three cDNA sets used in the GLIMMER training are small, including only 122-613 samples (see Chapter 3). However, these sequences include only the real genes with experimentally confirmed coding sequences. Thus their nucleotide frequencies should reflect the real genomic properties. On the other hand, the predicted ORF sets from the complete genomes are much larger, but they could include false positives, generated by the prediction methods. In order to see whether these two datasets show consistent properties, some genome statistics were compared between the two datasets. Table 5.5 compares base compositions of the three genomes between the two datasets. There is a good agreement in base composition between the small cDNA samples and the complete ORF sets of the three genomes. The distributions of base composition of the three training sets are comparable to those of predicted ORF sets. It justifies the use of the predicted ORF sets for various further analyses.

Table 5.5 Comparison of base composition between the small cDNA sample and complete predicted genome datasets for the three fungal species.

a) *Neurospora crassa*

	Base composition (%)					
	cDNA dataset: 122 cDNAs			Genome dataset: 10,082 ORFs		
Nucleotide	Mean ± SD[1]	Min[2]	Max[3]	Mean ± SD[1]	Min[2]	Max[3]
A	22.03 ± 2.92	14.18	28.19	23.68 ± 3.56	2.67	61.02
C	30.94 ± 2.42	22.28	40.36	28.69 ± 4.04	0	51.52
G	26.77 ± 2.42	20.44	32.34	27.14 ± 3.27	3.03	51.52
T	20.24 ± 2.28	15.35	27.60	20.47 ± 3.24	3.39	44.44

b) *Saccharomyces cerevisiae*

	Base composition (%)					
	cDNA dataset: 162 cDNAs			Genome dataset: 5,043 ORFs		
Nucleotide	Mean ± SD[1]	Min[2]	Max[3]	Mean ± SD[1]	Min[2]	Max[3]
A	31.45± 4.56	17.39	45.87	32.45 ± 3.85	15.43	47.44
C	19.52 ± 3.31	5.26	28.73	19.28 ± 3.00	6.67	43.27
G	22.29 ± 3.44	8.84	39.47	20.87 ± 2.61	7.60	32.67
T	26.73 ± 4.39	11.84	44.44	27.39 ± 3.47	10.25	48

c) *Schizosaccharomyces pombe*

	Base composition (%)					
	cDNA dataset: 613 cDNAs			Genome dataset: 5,845 ORFs		
Nucleotide	Mean ± SD[1]	Min[2]	Max[3]	Mean ± SD[1]	Min[2]	Max[3]
A	28.17 ± 4.29	16.72	41.80	30.07 ± 3.84	12.12	49.06
C	20.74 ± 3.21	11.63	32.23	19.16 ± 2.80	9.09	41.92
G	21.72 ± 2.64	13.98	36.79	20.06 ± 2.27	7.84	31.02
T	29.36 ± 3.19	16.67	40.31	30.71 ± 3.17	14.15	48.34

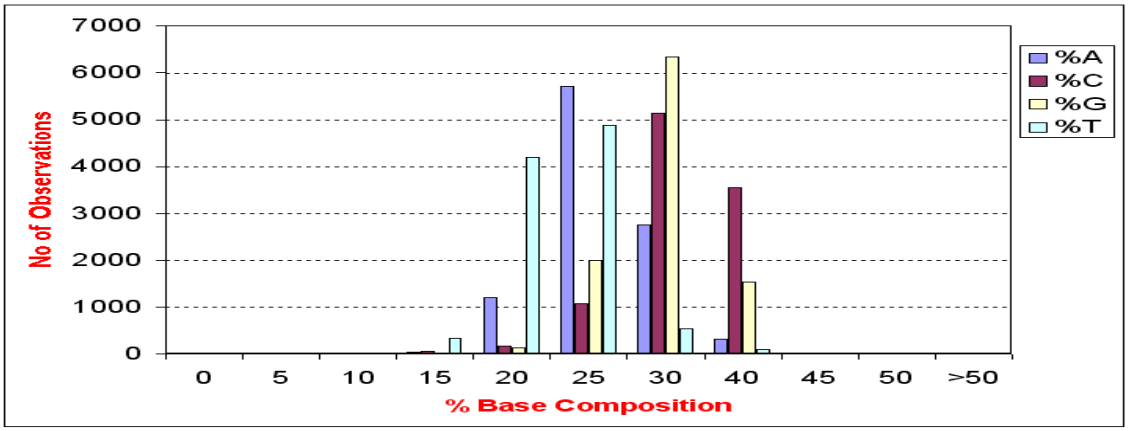
[1] SD: standard deviation, [2] Min: minimum base composition, [3] Max: maximum base composition.

Nucleotide frequencies were examined both for the coding sequences and introns. Figure 5.2 compares the single nucleotide frequencies (base compositions) of the coding sequences (ORFs) among the three fungal genomes (the similar histograms obtained from the cDNA datasets are included in Appendix E). The ORF sets of *N. crassa* show preference for nucleotides G and C (G+C% = 56), whereas *S. cerevisiae* and *S. pombe* have preference for nucleotides A and T (G+C% = 40). The cDNA data in Appendix E show the trends similar to their corresponding genome sets.

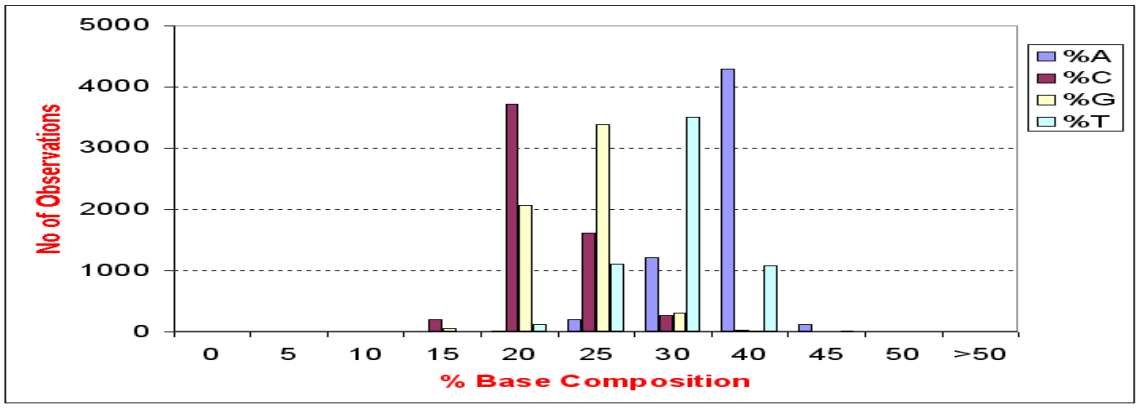
Table 5.6 shows the base composition (%) observed from intron sequences. Comparing to the data found in Table 5.5, an almost equal contribution of the four nucleotides with slightly lower G% is found in the introns of *N. crassa*. The intron sequences of the two yeast genomes, *S. pombe* and *S. cerevisiae*, are close to 70% A+T. Introns are non-coding sequences. Therefore, the base composition observed in these sequences should reflect the spontaneous mutation patterns. Based on Table 5.6, we can conclude that mutation patterns are not equal among four nucleotides. Mutation patterns are biased toward A and T nucleotides in *S. pombe* and *S. cerevisiae*. On the other hand, in *N. crassa*, mutations are biased toward non-G nucleotides, although the bias is not as pronounced as in the two yeast species.

The difference observed in Table 5.5 compared to Table 5.6 can be explained by the effect of functional constrains in the coding sequences. Appendix K shows the universal genetic code. Three consecutive nucleotides in coding regions form a unit called "codon", and each codon codes one of 20 amino acids. The frequencies of these 20 amino acids are not consistent among different proteins. Furthermore, mutations from

a. ***Neurospora crassa***
(Average base composition A: 23.68, C: 28.69, G: 27.14, and T: 20.47)



b. ***Saccharomyces cerevisiae***
(Average base composition A: 32.45, C: 19.28, G: 20.87, and T: 27.39)



c. ***Schizosaccharomyces pombe***
(Average base composition A: 30.07, C: 19.16, G: 20.06, and T: 30.71)

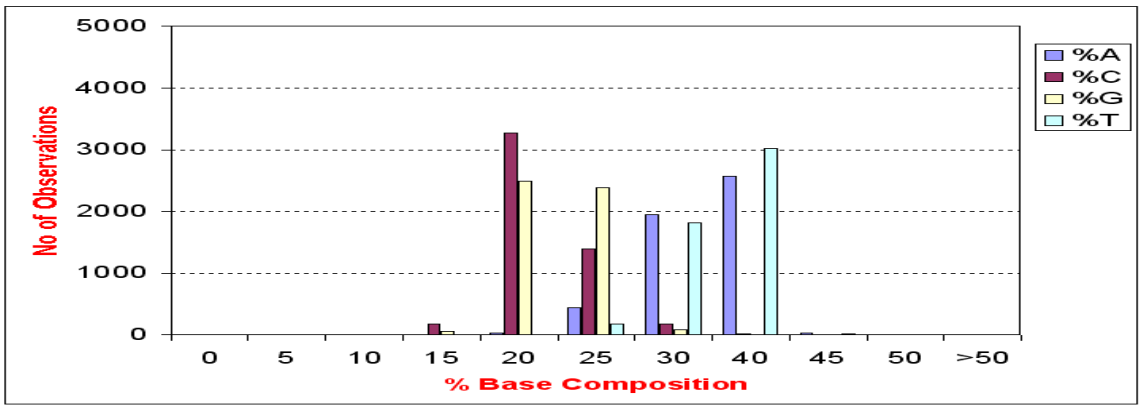


Figure 5.2: Frequency distribution of base composition from the three fungal genomes.

one nucleotide to another are under different levels of constraints depending on where the nucleotide is within the codon (first, second, or third codon position). As the code table shows, the majority of mutations between nucleotides at the third codon position do not change the coded amino acid (e.g., four codons CCT, CCC, CCA, and CCG all code the same amino acid, proline or Pro). On the other hand, such mutations are restricted for nucleotides at the first and second codon positions (e.g. a mutation between TTT and TTA changes coded amino acid between Phe and Leu). These constrains generate different patterns of base composition in coding sequences compared to introns and non-coding sequences. Compared to introns, coding sequences in all the three fungal genomes have relatively more G and C. This indicates that the constraint working on nucleotide substitutions in the coding regions is similar among the three fungal genomes. Understanding all of these differences is important in predicting gene structures (e.g. exons vs introns). It is also clear that species specific base composition data are necessary from both of coding and non-coding sequences for more accurate gene prediction.

The dinucleotide frequency was calculated as follows. First the occurrence of all possible nucleotide pairs (AA, AC, AG, …, TT) in each genomic sequence is counted by shifting the nucleotide position by 1 bp. Then all the occurrences are summed up (total number of dinucleotide pairs) and the proportion of each dinucleotide pair values is calculated from each genome. A similar procedure is used for calculating the trinucleotide frequencies. These frequency values can tell us the preference for particular nucleotide pairs and triplets in each genome.

Table 5.6 Comparison of base composition of introns from complete predicted genome datasets.

a) *Neurospora crassa*

	Base composition (%)		
	Genome dataset: 17,113 introns		
Nucleotide	Mean ± SD[1]	Minimum	Maximum
A	25.19 ± 6.09	5.17	72.32
C	25.37 ± 7.16	0.67	57.29
G	21.63 ± 6.45	2.50	54.17
T	27.80 ± 6.51	1.38	69.60

b) *Saccharomyces cerevisiae*

	Base composition (%)		
	Genome dataset: 267 introns		
Nucleotide	Mean ± SD[1]	Minimum	Maximum
A	31.46 ± 5.19	12.98	43.67
C	16.73 ± 3.85	7.14	43.28
G	16.77 ± 3.71	6.02	19.41
T	35.02 ± 5.88	28.57	53.17

c) *Schizosaccharomyces pombe*

	Base composition (%)		
	Genome dataset: 4,719 introns		
Nucleotide	Mean ± SD[1]	Minimum	Maximum
A	31.71 ± 6.45	8.47	57.69
C	13.25 ± 4.23	0	29.83
G	15.60 ± 4.09	3.39	32.78
T	39.39 ± 6.95	15	63.77

[1] SD: standard deviation.

Figures 5.3 and 5.4 summarize the dinucleotide frequencies observed in the coding sequences and introns from the three fungal genomes. The trinucleotide data are found in Appendices I and J. In the case of *N. crassa*, dinucleotides and trinucleoides including G or C occur more frequently compared to other combinations. However, the degree of variation in nucleotide combinations is not as high as in the other two fungi. Interestingly, AT happens almost twice as many as TA (4.7% vs. 2.7%). In the case of *S. pombe* and *S. cerevisiae*, dinucleotides and trinucleotides that include A or T are preferred over others. Dinucleotides AA, AT, and TT pairs occur quite often compared to other nucleotide pairs. Trinucleotides AAA and TTT are represented most in the two yeast genomes. Such bias becomes higher in intron sequences. Many of these biases appear to be consistent with the base composition (single nucleotide frequencies) described earlier. *N. crassa* tends to use more combinations including G and C, whereas other two species use more A and T related nucleotide combinations. AT bias in base composition is most pronounced in introns in *S. cerevisiae and S. pombe*.

Table 5.7 summarizes the lengths (bp) of introns and coding sequences (CDS) in the three fungi genomes. The table also gives the number of introns and ORFs in the three genomes. The most striking difference between the three fungal genomes is that the *S. cerevisea* genome has a very small number of introns. Both yeast species have smaller number of ORFs compared to *N. crassa*. *N. crassa* has twice as many ORFs. The difference is more pronounced in the intron numbers, *S. pombe* has much more introns than *S. cerevisea*, but *N. crassa* has even more introns.

a) *Neurospora crassa*

		Second nucleotide			
		A	C	G	T
First nucleotide	A	5.89 (858,168)	6.28 (913,794)	6.75 (982,624)	4.73 (688,011)
	C	7.53 (1,095,420)	8.22 (1,195,608)	7.07 (1,029,455)	6.22 (904,881)
	G	7.54 (1,096,121)	7.59 (1,105,004)	7.14 (1,038,698)	4.71 (686,169)
	T	2.68 (389,953)	6.94 (1,010,929)	6.04 (878,192)	4.63 (673,053)

b) *Saccharomyces cerevisiae*

		Second nucleotide			
		A	C	G	T
First nucleotide	A	11.78 (1,035,635)	5.82 (510,971)	6.53 (573,604)	8.63 (758,578)
	C	6.92 (607,617)	3.94 (346,314)	2.96 (260,848)	5.34 (469,234)
	G	7.43 (653,364)	3.83 (336,791)	4.35 (381,938)	4.81 (422,873)
	T	6.61 (580,827)	5.57 (489,937)	6.60 (579,921)	8.85 (778,381)

c) *Schizosaccharomyces pombe*

		Second nucleotide			
		A	C	G	T
First nucleotide	A	10.72 (776,858)	5.08 (368,407)	5.67 (410,795)	8.49 (615,377)
	C	5.92 (428,604)	3.79 (274,814)	3.22 (233,398)	6.30 (456,484)
	G	6.62 (480,029)	4.09 (296,834)	3.95 (285,949)	5.22 (378,355)
	T	6.69 (484,918)	6.26 (453,337)	7.06 (511,958)	10.91 (790,140)

Figure 5.3 Dinucleotide frequencies observed in ORF coding sequences. Frequencies are given in percentages and the numbers in parentheses are the numbers of each nucleotide pair.

a) *Neurospora crassa*

		Second nucleotide			
		A	C	G	T
First nucleotide	A	6.74 (155,035)	5.78 (151,524)	5.78 (133,152)	6.11 (140,651)
	C	7.17 (165,042)	6.89 (158,737)	4.37 (100,493)	7.22 (166,105)
	G	5.78 (133,090)	5.02 (115,522)	4.71 (108,269)	6.11 (140,674)
	T	5.53 (127,195)	7.15 (164,594)	6.77 (155,641)	8.04 (184,906)

b) *Saccharomyces cerevisiae*

		Second nucleotide			
		A	C	G	T
First nucleotide	A	11.51 (7,396)	5.58 (3,584)	5.52 (3,549)	9.91 (6,372)
	C	5.78 (3,715)	2.84 (1,828)	2.44 (1,567)	5.43 (3,487)
	G	6.05 (3,888)	2.97 (1,909)	2.84 (1,823)	5.23 (3,360)
	T	9.19 (5,902)	5.09 (3,275)	6.29 (4,042)	13.32 (8,560)

c) *Schizosaccharomyces pombe*

		Second nucleotide			
		A	C	G	T
First nucleotide	A	11.95 (16,419,491)	5.14 (706,440)	5.67 (779,460)	9.26 (1,272,990)
	C	6.13 (842,974)	3.27 (449,563)	2.86 (392,769)	5.65 (776,687)
	G	5.93 (814,863)	3.61 (495,271)	3.28 (450,778)	5.15 (707,540)
	T	8.01 (1,101,055)	5.89 (810,721)	6.16 (845,443)	12.04 (1,654,297)

Figure 5.4 Dinucleotide frequencies observed from intron sequences. Frequencies are given in percentages and the numbers in parentheses are the numbers of each nucleotide pairs.

Table 5.7 Comparison of sequence length (bp) between the intron and coding sequences

a) *Neurospora crassa*

CDS: 10,082 ORFs			Introns: 17,113 introns		
Mean ± SD	Min	Max	Mean ± SD	Min	Max
1,443.77 ± 1,230.22	30	32,463	135.43 ± 129.08	20	2074

b) *Saccharomyces cerevisiae*

CDS: 5,845 ORFs			Introns: 267 introns		
Mean ± SD	Min	Max	Mean ± SD	Min	Max
1,504.31 ± 1,143.64	51	14,733	241.66 ± 175.87	49	1,002

c) *Schizosaccharomyces pombe*

CDS: 5,043 ORFs			Introns: 4,719 introns		
Mean ± SD	Min	Max	Mean ± SD	Min	Max
1,437.89 ± 1,060.88	54	15,005	82.38 ± 68.03	30	817

The length distributions of complete ORF datasets are shown also in Appendix H. It shows that there are both very short (<= 100 bp) as well as extremely long (>= 20,000 bp) ORFs in all of the three fungal genomes. In the case of *N. crassa*, one ORF was predicted to be longer than 32 kb. For these extremely long ORFs, prediction mistakes may need to be considered. On the other hand, more short ORFs (<= 100 bp) were

observed in the *N. crassa* compared to the two yeast genomes. There are 46 ORFs in *N. crassa* shorter than 50 bp whereas the minimum ORF lengths are 51 and 54 bp in *S. cerevisiae* and *S. pombe*, respectively.

The length distributions of introns were again very different among the three fungal genomes (Appendix G). *N. crassa* has more than 11,000 introns shorter than 100 bp, whereas *S. cerevisiae* has fewer than 100 such short introns. In the *S. pombe* genome, there are approximately 3,500 introns shorter than 100 bp. The *N. crassa* genome also has 30 introns longer than 1000 bp compared to the other two yeast genomes (*S. cerevisea* has only one such intron and *S. pombe* has none). Both yeast genomes are more compact than *N. crassa*. As described earlier in Table 1.1, the two yeast genomes are close to 10 Mb, while the *N. crassa* genome is approximately 40 Mb. However, to achieve the compactness, *S. cerevisiae* has very few introns whereas *S. pombe* has more but shorter introns. The genome of *N. crassa*, a filamentous fungus, is much more complex with more and longer introns, compared to the two non-filamentous fungi. These observations exemplify again the necessity of species-specific optimization of gene prediction methods.

Chapter 6 Conclusions and Future Development

In this thesis, comparative analysis of three fungal genomes: *Neurospora crassa*, *Aspergillus nidulans*, and *Fusarium graminearum*, was performed. Various similarity searches performed using PSI-BLAST against the NCBI Non-redundant database and three other fungal genomes provided insights on the characteristics and uniqueness of these filamentous fungi genomes. The results obtained showed a large number of species-specific sequences from all of the three genomes. In particular, there were 915 *N. crassa* specific ORFs that were shorter than 100 aa, whereas much smaller numbers of such short ORFs (fewer than 100) were found from *F. graminearum* and *A. nidulans* genomes. These protein sequences can become candidates for further in depth analysis.

The Ser/Thr content was examined from their putative protein sequences and transmembrane regions were predicted using different prediction methods. Though the numbers of identified regions were not the same among the methods, the total numbers of proteins predicted as containing transmembrane regions were very close. However, there appeared to be still an overestimation of transmembrane proteins, and more investigation on transmembrane prediction is required.

Three gene mining methods: GLIMMER, GLIMMERM, and GenScan, were examined on their performance to predict *N. crassa* ORFs. Their performance was compared with the existing *N. crassa* genome annotation by the Whitehead genome project. New gene candidates previously not annotated were identified and examined. GenScan performed poorly compared to the other two programs. This was expected since GenScan was not trained with any fungal dataset, and also the method was optimized for larger vertebrate genomes that include longer and more complex gene

structures. In the future, training of the above programs should be carried out with the same datasets especially from filamentous fungi in order to validate the results. Comparison of the results across various gene prediction methods can be done to identify any common predictions.

Some genomic information including nucleotide frequencies was extracted from three fungal genomes: *Neurospora crassa, Schizosaccharomyces pombe* and *Saccharomyces cerevisiae*, and they were compared between coding and non-coding sequences and also among the genomes. Mutation patterns appeared to be different between *N. crassa* and other two non-filamentous fungal genomes, and such difference must have caused the different nucleotide frequencies observed among the three genomes. The length distributions showed that there are both very short (<= 100 bp) as well as extremely long (>= 20,000 bp) ORFs in all of the three fungal genomes. Especially for those extremely long ORFs, mistakes in gene prediction need to be considered. On the other hand, more short ORFs (<= 50 bp) were observed in the *N. crassa* genome compared to the two yeast genomes. The numbers as well as length distributions of introns were also very different among the three fungal genomes. Examining these genome-specific features should help us optimizing genome prediction methods for other, particularly non-filamenous fungal species. The future plan is to extend the analysis to *Fusarium graminearum* and *Aspergillus nidulans* in order to identify more filamentous fungi-specific genomic features.

Finally, a database was constructed based on the *Neurospora crassa* genomic data to compile information useful for various gene prediction methods. The stored data can be visualized using a web-based graphical interface. The plan is to add more genomic

information from other fungal genomes as well as other model organisms to help more detailed comparative analysis. This database development is expected to facilitate the development and optimization of fungal specific gene prediction methods in the future.

References

1. Goffeau,A., Barrell,B.G., Bussey,H., Davis,R.W. *et al*. (1996) Life with 6000 genes. *Science*, **274**, 546–67.
2. Wood,V., Gwilliam,R., Rajandream,M.A., Lyne,M. *et al*. (2002) The genome sequence of *Schizosaccharomyces pombe*. *Nature*, **414**, 871–80.
3. Burge, C. B. and S. Karlin (1998). "Finding the genes in genomic DNA." *Curr Opin Struct Biol* **8**(3): 346-54.
4. Claverie, J. M. (1997). "Computational methods for the identification of genes in vertebrate genomic sequences." *Hum Mol Genet* **6**(10): 1735-44.
5. Kraemer, E., J. Wang, *et al*. (2001). "An analysis of gene-finding programs for Neurospora crassa." *Bioinformatics* **17**(10): 901-12.
6. Mannhaupt, G., C. Montrone, *et al*. (2003). "What's in the genome of a filamentous fungus? Analysis of the Neurospora genome sequence." *Nucleic Acids Res* **31**(7): 1944-54.
7. Salamov, A. A. and V. V. Solovyev (2000). "Ab initio gene finding in Drosophila genomic DNA." *Genome Res* **10**(4): 516-22.
8. Kulp, D., D. Haussler, *et al*. (1996). "A generalized hidden Markov model for the recognition of human genes in DNA." *Proc Int Conf Intell Syst Mol Biol* **4**: 134-42.
9. Burge, C. and S. Karlin (1997). "Prediction of complete gene structures in human genomic DNA." *J Mol Biol* **268**(1): 78-94.

10. Schoof, H., P. Zaccaria, *et al.* (2002). "MIPS Arabidopsis thaliana Database (MAtDB): an integrated biological knowledge resource based on the first complete plant genome." *Nucleic Acids Res* **30**(1): 91-3.

11. Altschul, S.F., Gish, W., Miller, W., Myers, E.W. and Lipman, D.J. (1990). Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–10.

12. Altschul, S.F., Madden, T.L., Schäffer, A.A., Zhang, J., Zhang, Z., Miller, W. and Lipman, D.J. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–402.

13. Altschul, S.F. and Koonin, E.V. (1998). Iterated profile searches with PSI-BLAST – a tool for discovery in protein databases. *Trends Biochem. Sci.*, **23**, 444–47.

14. Tusnády, G.E and Simon, I. (2001). The HMMTOP transmembrane topology prediction server. *Bioinformatics* **17**, 849-50

15. Krogh, A., Larsson, B., von Heijne, G. and Sonnhammer, E.L. (2001). Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J Mol Biol.* **305**(3): 567-80.

16. Kyte, J. and Doolittle, R.F. (1982). A simple method for displaying the hydropathic character of a protein. *J. Mol. Biol.* **157,** 105–32.

17. Eisenberg, D., Schwartz, E., Komáromy, M. and Wall, R. (1984). Analysis of membrane and surface proteins sequences with the hydrophobic moment plot. *J. Mol. Biol.* **179**, 125–42.

18. Ponnuswamy, P.K. and Gromiha, M.M. (1993). Prediction of transmembrane helices from hydrophobic characteristics of protein. *Int. Peptide Protein Res.* **42**, 326–41.

19. Gromiha, M.M. and Ponnuswamy, P.K. (1995). Prediction of protein secondary structures from their hydrophobic characteristics. *Int. J. Peptide Proteins Res.* **45**, 225–40.

20. Tusnády, G.E. and Simon, I. (1998). Principles Governing Amino Acid Composition of Integral Membrane Proteins: Applications to Topology Prediction." *J. Mol. Biol.* **283**, 489-506.

21. Sonnhammer, E.L., von Heijne, G. and Krogh, A. (1998). A hidden Markov model for predicting transmembrane helices in protein sequences. I*n J. Glasgow et al., eds.*, *Proceedings of the Sixth International Conference on Intelligent Systems for Molecular Biology*, AAAI Press, Menlo Park, CA, 175–82.

22. Claverie, J.-M. (1997). Computational methods for the identification of genes in vertebrate genomic sequences. *Hum. Mol. Genet*. **6**, 1735-44.

23. Stormo, G.D. (2000). Gene-finding approaches for eukaryotes. *Genome Res.* **10**, 394-97.

24. Burset, M. and Guigó, R. (1996). Evaluation of gene structure prediction programs. *Genomics* **34**, 353-67.

25. Guigó, R., Agarwal, P., Abril, J.F., Burset, M. and Fickett, J.W. (2000). An Assessment of Gene Prediction Accuracy in Large DNA Sequences. *Genome Res.* **10**, 1631-42.

26. Delcher, A.L., Harmon, D., Kasif, S., White, O. and Salzberg, S.L. (1998). Improved microbial gene identification with GLIMMER *Nucleic Acids Res.,* **27**(23), 4636-41.

27. Salzberg, S., Delcher, A., Kasif, S. and White, O. (1998). Microbial gene identification using interpolated Markov models *Nucleic Acids Res.* **26**(2), 544-48.

28. Uberbacher, E.C. and Mural, R.J. (1991). Locating protein-coding regions in human DNA sequences by a multiple sensor-neural network approach. *Proc. Natl. Acad. Sci*. **88**, 11261-65.

29. Guigó, R., Knudsen, S., Drake, N. and Smith, T. F. (1992). Prediction of gene structure. *J. Mol. Biol.* **226**, 141-57.

30. Burge, C. and Karlin, S. (1997). Prediction of complete gene structures in human genomic DNA. *J. Mol. Biol.* **268**, 78-94.

31. Lukashin, A.V. and Borodovsky, M. (1998). GeneMark.hmm: new solutions for gene finding. *Nucleic Acids Res.* **26,** 1107-15.

32. Birney, E. and Durbin, R. (1997). Dynamite: A exible code generating language for dynamic programming methods used in sequence comparison. *Proceedings ISMB*, **5**, 56-64.

33. Gelfand, M.S., Mironov, A.A. and Pevzner, P.A. (1996). Gene recognition via spliced sequence alignment. *Proc. Natl. Acad. Sci.*, **93**, 9061-66.

34. Borodovsky, M. and McIninch, J. (1993). GeneMark: parallel gene recognition for both DNA strands. *Computers & Chemistry*, **17**(19), 123-33.

35. Majoros, W.H., Pertea, M., Antonescu, C. and Salzberg, S.L. (2003). GlimmerM, Exonomy and Unveil: three ab initio eukaryotic genefinders *Nucleic Acids Res.* **31,** 3601-4.

36. Murthy, S.K., Kasif, S. and Salzberg, S.L. (1994). A system for induction of oblique decision trees. *J. Artificial Intelligence Res.*, **2**, 1–32.

37. Pertea, M. and Salzberg, S.L. (2002). Computational gene finding in plants. *Plant Molec. Biol.* **48**(1-2), 39-48.

38. Burge, C. B. and Karlin, S. (1998) Finding the genes in genomic DNA. *Curr. Opin. Struct. Biol.* **8,** 346-354.

Appendix A: A sample output of HMMTOP for a *N. crassa* protein, NCU10034.1

```
Protein: NCU10034.1 predicted protein (4678 - 5483)
Length:  214
N-terminus: OUT
Number of transmembrane helices: 3
Transmembrane helices: 17-41 106-125 156-180

Total entropy of the model:  17.0073
Entropy of the best path:  17.0092

The best path:

seq  MPSLLVVIFV IELFVQLVNT IGAATINNLL WRIALSLPLP LSAQFAAQRK   50
pred Ooooooooooo oooooohhhh hhhhhhhhhh hhhhhhhhhh hiiiiiiiii

seq  KQKEYLAIRR ELNATSSQDE FAKWARLRRQ HDKLLEDLEK RKKELDAAKT  100
pred iiiiiiIIII IIIIIIIIII IIIIIIIIII IIIIIIIIII iiiiiiiiii

seq  KFDRTLTTVR VVATRGLQWF LPFWYSREPM FWLPYGWFPY YVEWFASFPR  150
pred iiiiiHHHHH HHHHHHHHHH HHHHHooooo oooooooooo oooooooooo

seq  APLGSVSIVV WQWACTGVIK LVIETVMAVV GLIVAARQKQ QEKQKAKQAV  200
pred ooooohhhhh hhhhhhhhhh hhhhhhhhhh iiiiiiiiii iiiiiIIIII

seq  PAAGGGDSKA EEAK 214
pred IIIIIIIIII IIII
```

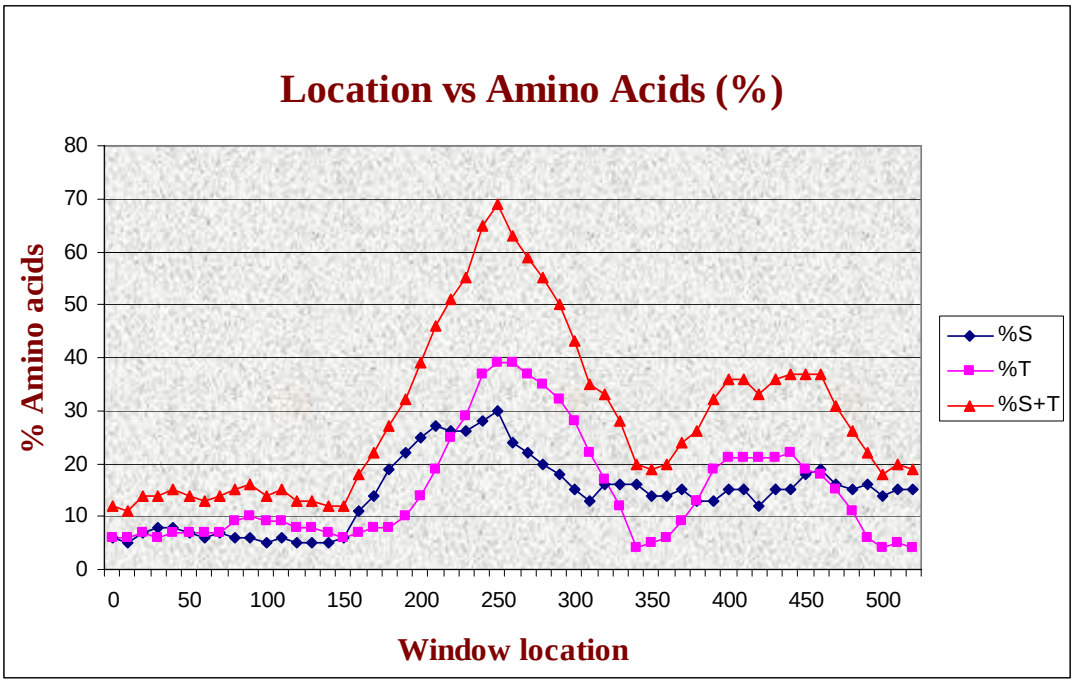
The predicted transmembrane regions are denoted by "H". For the above example, the program predicted three transmembrane segments.

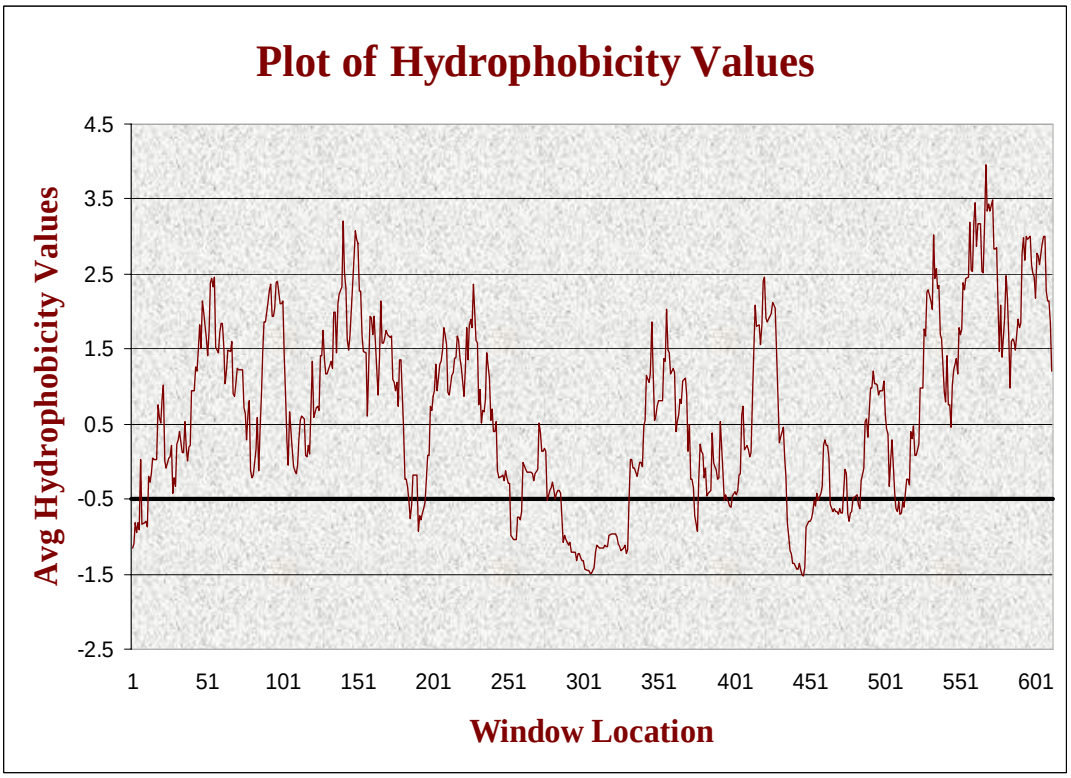
Appendix B: A sample output of TMHMM for a *N.crassa* protein, NCU10034.1

```
# NCU10034.1 Length: 214
# NCU10034.1 Number of predicted TMHs:  3
# NCU10034.1 Exp number of AAs in TMHs: 60.37802
# NCU10034.1 Exp number, first 60 AAs:  22.80493
# NCU10034.1 Total prob of N-in:        0.21769
# NCU10034.1 POSSIBLE N-term signal sequence
NCU10034.1      TMHMM2.0        outside      1    3
NCU10034.1      TMHMM2.0        TMhelix      4    26
NCU10034.1      TMHMM2.0        inside       27   130
NCU10034.1      TMHMM2.0        TMhelix      131  148
NCU10034.1      TMHMM2.0        outside      149  162
NCU10034.1      TMHMM2.0        TMhelix      163  185
NCU10034.1      TMHMM2.0        inside       186  214
```

The predicted transmembrane segments are denoted by TMhelix. For the above example, the program predicted three transmembrane segments.

Appendix C: An example plot of Ser/Thr % and hydrophobicity along the amino acid sequence of a known cell wall protein from *N. crassa*, NCU00039.1.





Appendix D: A sample GLIMMER output using *N. crassa* cDNA dataset for training the model and one of *N. crassa* contigs as input

```
GC Proportion = 35.4%
Minimum gene length = 30
Minimum overlap length = 30
Minimum overlap percent = 10.0%
Threshold score = 90
Use independent scores = True
Ignore independent score on orfs longer than 502
Use strict independent model = True
Use first start codon = True

             Orf     Gene                Lengths     Gene  |-- Frame Scores -  Indep
  ID#  Fr    Start    Start      End      Orf  Gene    Score   F1 F2 F3 R1 R2 R3  Score
       R2      176       89       60      117    30      0     0  0  _  2  0  _    97      0  -1.704
       F1      364      391      423       60    33      0     0  _  _  _  _  0    99      0  -1.625
       R1      759      753      697       63    57      0     _ 23  0  0  _  0    76      0  -1.624
       R3      769      679      494      276   186      0     _  _  _  _  _  0    99      0  -1.484
    1  F1     1762     1789     2034      273   246     93    93  _  _  _  _  _     6    177  -1.411
       F3     1935     1977     2048      114    72      0     _  _  0 28  _  _    71      0  -1.522
       R1     2106     2085     1975      132   111     26     _  _  _ 26  _  _    73     54  -1.419
       F2     2015     2054     2155      141   102      2     _  2  _  _  _  _    97      2  -1.423
       R3     2197     2197     2153       45    45      0     9  _  6 29 53  0     0      0  -1.502
       F3     2124     2136     2198       75    63     13     7  _ 13  _ 77  _     0     19  -1.418
       F2     2159     2219     2314      156    96      0     _  0  0  _  _  _    99      2  -1.563
       F3     2202     2217     2384      183   168      0     _  _  0  _  _  _    99     12  -1.586
End = 2279

   Original Genes =     1
  Potential Genes =     1
       Avg Olaps =   0.0
Potential Changes =     0
Potential Rejects =     0
     Sure Rejects =     0

   Original Genes =     1
  Potential Genes =     1
       Avg Olaps =   0.0
Potential Changes =     0
Potential Rejects =     0
     Sure Rejects =     0


Putative Genes:
    1     1789     2034  [+1 L= 246 r=-1.411]
```

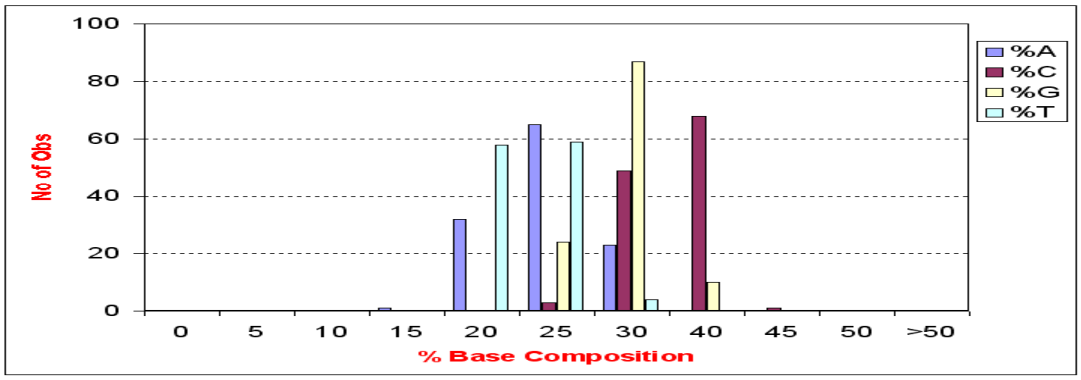
Predicted ORF*:

```
>Nucleotide sequence 10_1
ttgacgacaagacgacatttggtcggcacagaggagatggtgagacaagcggaacatgctca
tgtggtcaaggatgagtttccccctcacttcgccaaagaccaggcatgcacatgtcatgcgc
cacacttgacaagagatgataaaatggcttcgcttcgccgccaggttccgcagtcgcgttca
ccgtgcgaagaattcccacagcttcagcaacctttgtaggtatcctcagaccccaacttt
```

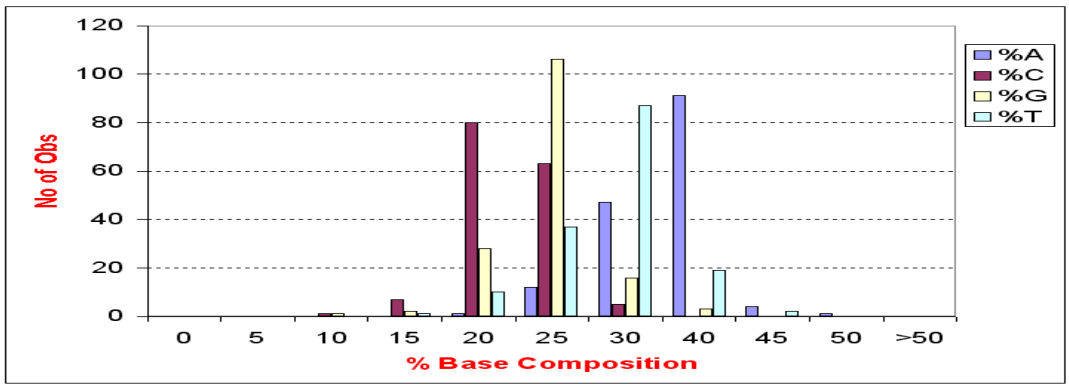
*The stop codon is not included in the output. GLIMMER prediction allows three possible start codons: ATG, TTG or GTG.

Appendix E: Frequency distribution of base composition based on cDNA training sets

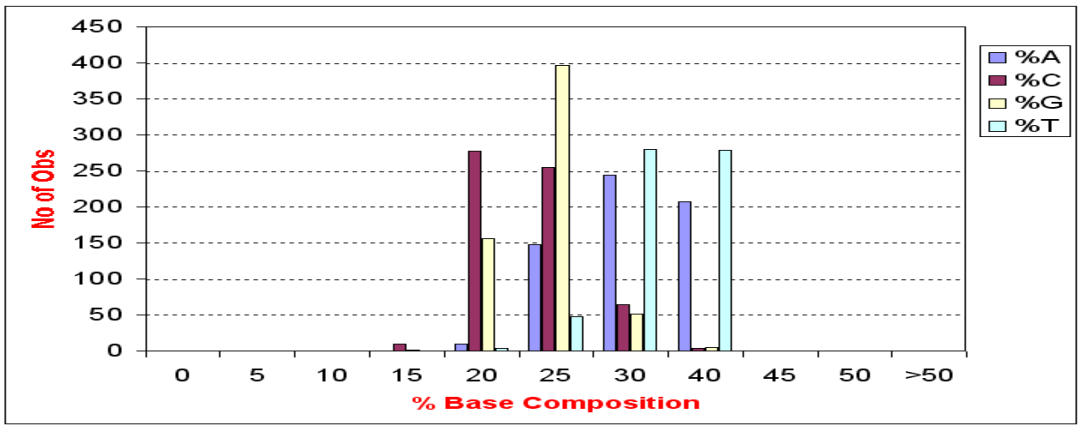
a. *Neurospora crassa*
(Average base compositions A: 22.03, C: 30.94, G: 26.77 and T: 20.24)



b. *Saccharomyces cerevisiae*
(Average base compositions A: 31.45, C: 19.52, G: 22.29 and T: 26.73)

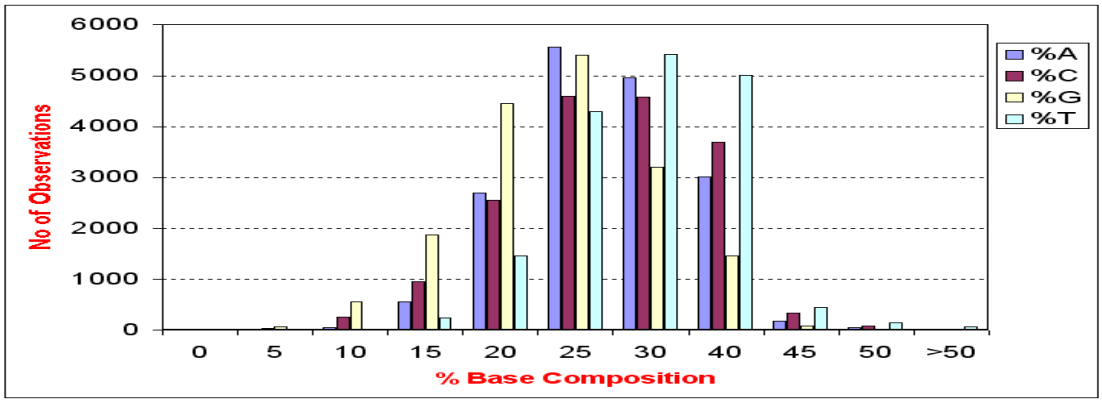


c. *Schizosaccharomyces pombe*
(Average base compositions A: 28.17, C: 20.74, G: 21.72 and T: 29.36)

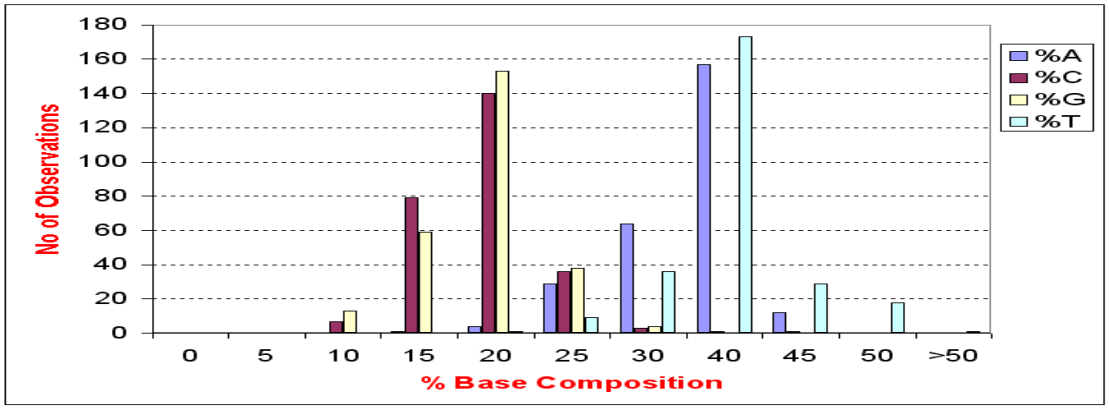


Appendix F: Frequency distribution of base composition from the intron datasets

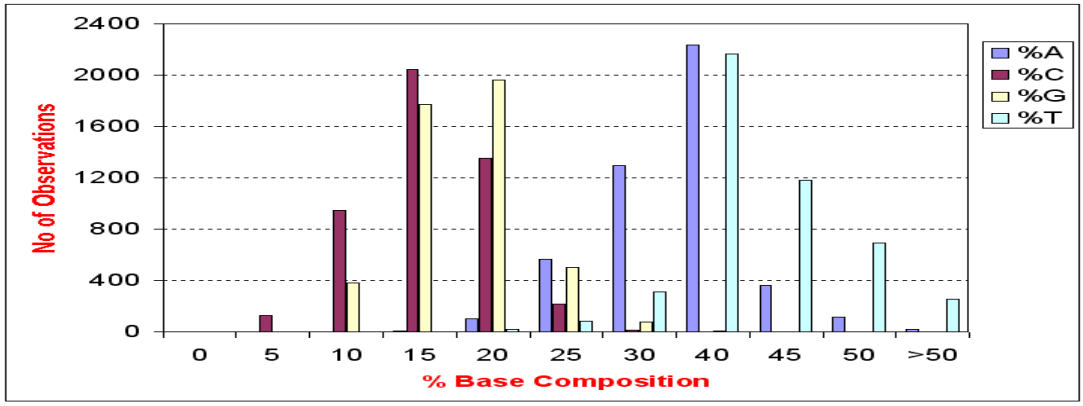
a. ***Neurospora crassa***
(Average base compositions A: 25.19, C: 25.37, G: 21.63 and T: 27.80)



b. ***Saccharomyces cerevisiae***
(Average base compositions A: 31.46, C: 16.73, G: 16.77 and T: 35.02)

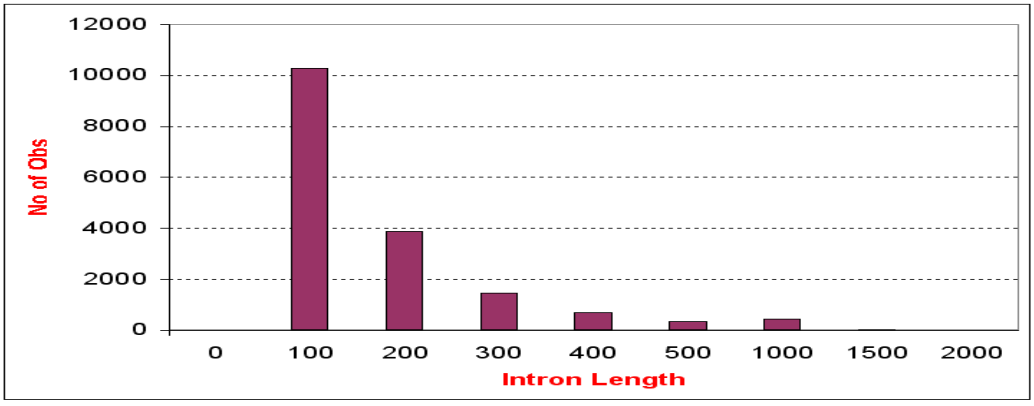


c. ***Schizosaccharomyces pombe***
(Average base compositions A: 31.71, C: 13.25, G: 15.60 and T: 39.39)

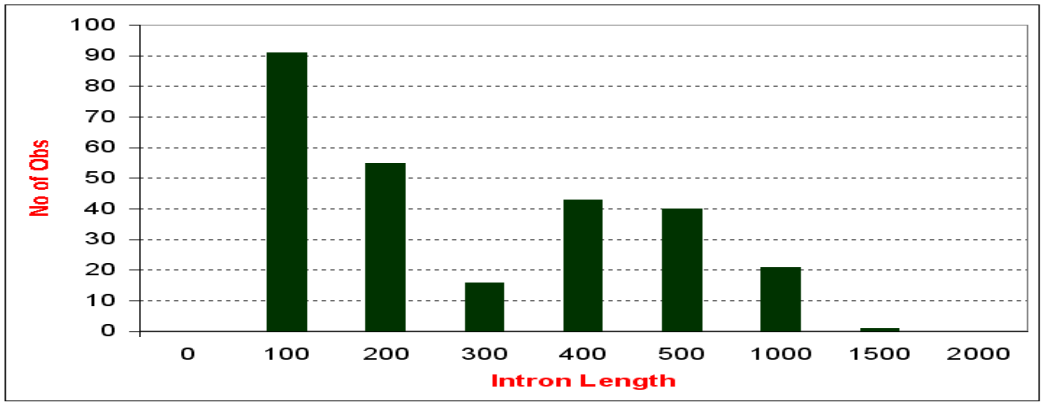


Appendix G: Frequency distribution of intron lengths (bp).

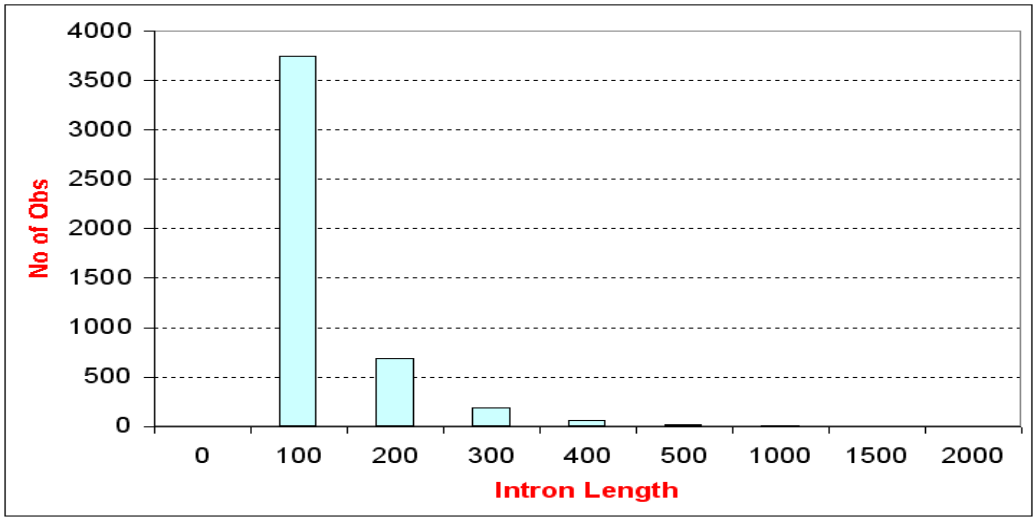
a. ***Neurospora crassa***



b. ***Saccharomyces cerevisiae***

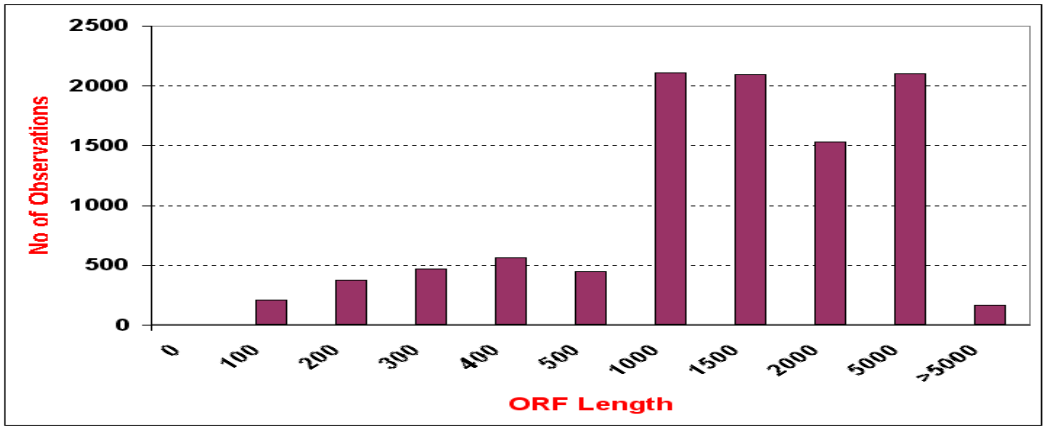


c. ***Schizosaccharomyces pombe***

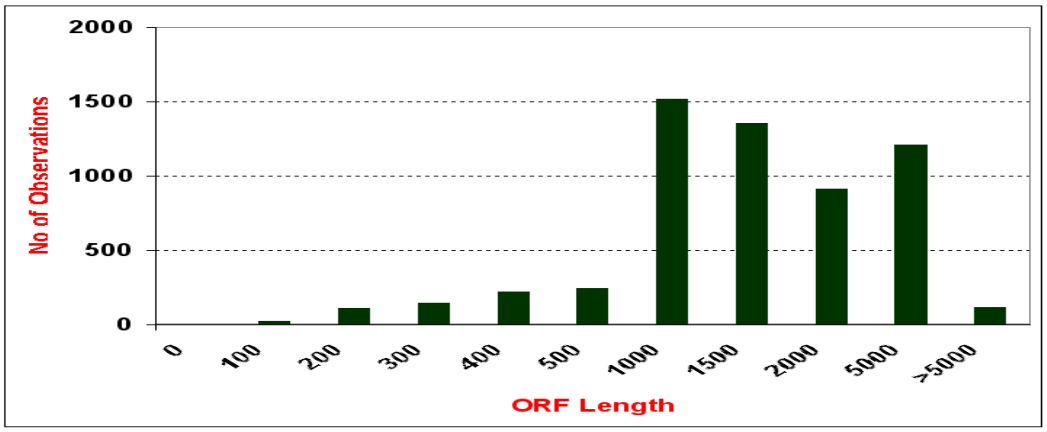


Appendix H: Frequency distribution of coding sequence (CDS) lengths (bp)

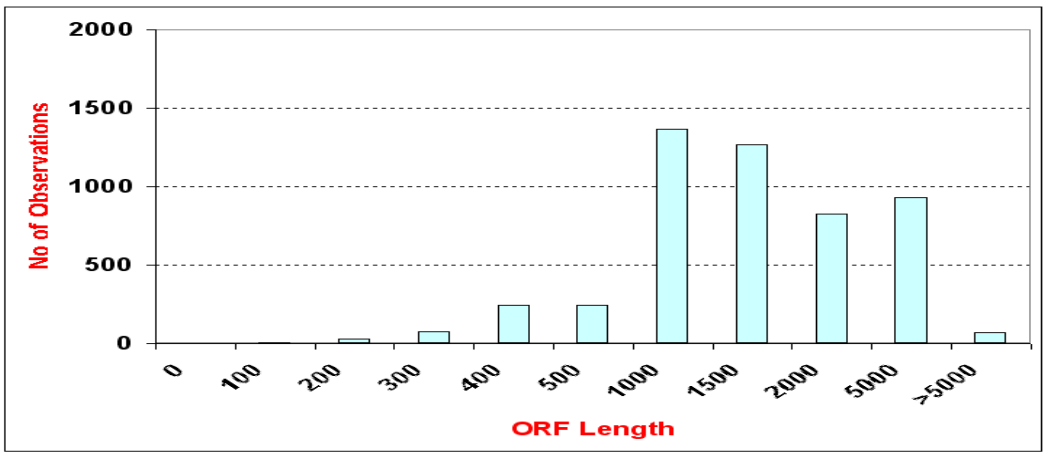
a. ***Neurospora crassa***



b. ***Saccharomyces cerevisiae***



c. ***Schizosaccharomyces pombe***



Appendix I: Trinucleotide frequencies observed in ORF coding sequences*

a) ***Neurospora crassa***

	A	**C**	**G**	**T**	
A	1.22 (177,635)	1.62 (235,719)	1.84 (267,911)	0.59 (86,440)	**A**
	1.63 (237,460)	1.94 (281,297)	1.91 (277,459)	1.62 (234,820)	**C**
	2.21 (321,732)	1.56 (227,627)	1.95 (283,084)	1.63 (236,489)	**G**
	0.81 (118,406)	1.16 (169,151)	1.04 (151,193)	0.89 (130,262)	**T**
C	2.42 (352,416)	2.32 (337,333)	2.15 (312,805)	0.88 (128,586)	**A**
	1.64 (238,970)	2.04 (296,408)	1.96 (285,422)	2.12 (308,748)	**C**
	1.90 (276,396)	2.06 (299,591)	1.75 (254,078)	1.66 (241,872)	**G**
	1.56 (227,638)	1.81 (262,276)	1.22 (177,150)	1.55 (225,675)	**T**
G	1.85 (269,779)	1.81 (263,501)	2.09 (304,655)	0.68 (98,814)	**A**
	1.89 (275,736)	2.31 (336,317)	2.15 (312,047)	1.62 (235,567)	**C**
	2.22 (322,229)	1.73 (250,888)	1.46 (212,858)	1.26 (184,382)	**G**
	1.54 (224,207)	1.74 (254,928)	1.44 (209,138)	1.15 (167,406)	**T**
T	0.40 (58,338)	1.78 (258,867)	1.45 (210,750)	0.52 (76,113)	**A**
	1.11 (161,628)	1.94 (281,586)	1.58 (230,076)	1.59 (231,794)	**C**
	0.42 (62,267)	1.73 (251,349)	1.98 (288,678)	1.48 (215,449)	**G**
	0.74 (107,720)	1.51 (219,127)	1.02 (148,688)	1.02 (149,697)	**T**

b) ***Saccharomyces cerevisiae***

	A	**C**	**G**	**T**	
A	4.16 (365,600)	2.00 (175,713)	2.59 (228,211)	2.03 (178,461)	**A**
	2.00 (175,755)	1.19 (105,345)	1.18 (104,040)	1.72 (150,551)	**C**
	2.75 (241,387)	1.01 (88,781)	1.34 (117,997)	2.11 (184,937)	**G**
	2.84 (250,131)	1.61 (141,132)	1.39 (122,011)	2.78 (244,629)	**T**
C	2.76 (242,681)	1.65 (144,686)	1.07 (94,177)	1.41 (123,663)	**A**
	1.11 (97,083)	0.67 (59,377)	0.56 (49,060)	1.03 (90,402)	**C**
	1.41 (123,747)	0.61 (53,517)	0.59 (51,768)	1.31 (114,788)	**G**
	1.64 (144,106)	1.01 (88,734)	0.75 (65,843)	1.59 (140,381)	**T**
G	2.93 (257,154)	1.30 (114,226)	1.49 (130,040)	1.16 (101,797)	**A**
	1.18 (103,642)	0.82 (72,383)	0.84 (74,197)	0.93 (81,270)	**C**
	1.30 (114,259)	0.53 (45,897)	0.71 (62,763)	1.06 (93,392)	**G**
	2.01 (176,571)	1.19 (104,285)	1.31 (114,938)	1.67 (146,414)	**T**
T	1.94 (170,200)	1.97 (172,992)	2.29 (200,936)	2.01 (176,906)	**A**
	1.53 (134,491)	1.24 (109,209)	1.24 (109,494)	1.91 (167,714)	**C**
	1.07 (94,209)	0.83 (72,653)	1.71 (149,410)	2.13 (186,804)	**G**
	2.07 (181,927)	1.54 (135,083)	1.37 (120,081)	2.81 (246,957)	**T**

c) *Schizosaccharomyces pombe*

	A	**C**	**G**	**T**	
A	3.90 (282,692) 1.73 (125,252) 2.34 (169,727) 2.71 (196,323)	1.57 (113,751) 1.00 (72,732) 0.89 (64,994) 1.61 (116,922)	1.89 (137,282) 1.27 (92,523) 1.09 (79,345) 1.39 (100,617)	1.73 (124,913) 1.65 (119,829) 2.16 (156,183) 2.96 (214,441)	**A** **C** **G** **T**
C	2.45 (170,034) 0.85 (61,940) 1.13 (81,646) 1.58 (11,4979)	1.22 (88,263) 0.68 (49,855) 0.64 (46,637) 1.24 (90,048)	1.08 (78,611) 0.58 (42,063) 0.55 (39,818) 1.01 (72,900)	1.38 (100,049) 1.29 (92,968) 1.37 (99,582) 2.26 (163,879)	**A** **C** **G** **T**
G	2.59 (187,785) 0.96 (69,593) 1.18 (85,315) 1.88 (136,353)	1.23 (89,205) 0.75 (54,104) 0.57 (44,173) 1.55 (112,348)	1.45 (105,131) 0.73 (52,969) 0.57 (41,299) 1.19 (86,545)	1.20 (81,788) 1.02 (73,999) 1.04 (75,377) 1.96 (141,788)	**A** **C** **G** **T**
T	1.88 (136,336) 1.54 (111,621) 1.02 (74,102) 2.25 (162,854)	1.89 (137,378) 1.36 (98,121) 1.12 (80,590) 1.89 (137,163)	2.19 (158,904) 1.51 (109,275) 1.73 (125,484) 1.63 (118,286)	2.38 (172,765) 2.29 (166,532) 2.49 (180,811) 3.73 (270,022)	**A** **C** **G** **T**

***Frequencies are given in percentages and the numbers in parentheses are the numbers of each nucleotide pairs.**

Appendix J: Trinucleotide frequencies observed from intron sequences.

a) ***Neurospora crassa***

	A	**C**	**G**	**T**	
A	1.87 (42,685) 1.87 (42,601) 1.66 (37,844) 1.39 (31,905)	2.08 (47,609) 1.77 (40,416) 1.12 (25,646) 1.65 (37,853)	1.37 (31,317) 1.25 (28,638) 1.15 (26,234) 1.31 (29,850)	1.37 (31,260) 1.63 (37,250) 1.69 (38,652) 1.46 (33,489)	**A** **C** **G** **T**
C	1.93 (44,014) 1.72 (39,347) 1.69 (38,712) 1.88 (42,969)	1.95 (44,644) 1.96 (44,692) 1.12 (25,493) 1.92 (43,908)	1.24 (28,284) 1.09 (25,068) 0.94 (21,429) 1.13 (25,712)	1.46 (33,249) 1.91 (43,591) 1.66 (37,995) 2.24 (51,270)	**A** **C** **G** **T**
G	1.54 (34,954) 1.52 (34,666) 1.36 (31,202) 1.41 (32,268)	1.33 (30,425) 1.26 (28,701) 0.91 (20,727) 1.56 (35,669)	1.33 (30,359) 1.11 (25,390) 1.10 (25,317) 1.19 (27,203)	1.58 (36,155) 1.46 (33,368) 1.44 (33,018) 1.67 (38,133)	**A** **C** **G** **T**
T	1.46 (33,382) 1.53 (34,910) 1.11 (25,394) 1.46 (33,509)	1.85 (42,364) 1.96 (44,928) 1.25 (28,627) 2.13 (48,675)	1.88 (43,130) 1.59 (36,426) 1.54 (35,289) 1.78 (40,796)	1.16 (26,531) 2.21 (50,385) 2.01 (45,976) 2.72 (62,014)	**A** **C** **G** **T**

b) ***Saccharomyces cerevisiae***

	A	**C**	**G**	**T**	
A	4.13 (884) 2.15 (460) 2.04 (437) 3.16 (676)	1.98 (425) 0.99 (212) 0.86 (184) 1.87 (401)	1.90 (407) 0.92 (196) 0.91 (194) 1.21 (324)	3.04 (652) 1.33 (285) 2.00 (430) 3.50 (750)	**A** **C** **G** **T**
C	2.00 (430) 0.86 (184) 1.07 (230) 1.62 (348)	1.13 (242) 0.47 (101) 0.47 (101) 0.78 (169)	0.85 (182) 0.38 (82) 0.40 (86) 0.81 (174)	1.64 (353) 0.93 (199) 0.96 (207) 2.02 (433)	**A** **C** **G** **T**
G	2.32 (496) 0.77 (166) 1.08 (233) 1.65 (354)	1.04 (224) 0.44 (96) 0.46 (100) 0.96 (206)	1.24 (266) 0.47 (102) 0.51 (110) 0.61 (130)	2.33 (500) 0.56 (121) 0.96 (205) 1.75 (374)	**A** **C** **G** **T**
T	2.75 (589) 1.73 (370) 1.36 (292) 2.76 (592)	1.78 (383) 0.97 (209) 0.75 (161) 1.76 (379)	2.31 (494) 1.14 (244) 0.95 (204) 2.75 (590)	2.89 (621) 2.13 (456) 2.02 (433) 5.98 (1281)	**A** **C** **G** **T**

c) *Schizosaccharomyces pombe*

	A	C	G	T	
A	4.55 (208,779)	1.69 (77,790)	1.81 (82,787)	2.51 (114,831)	**A**
	1.89 (86,988)	0.98 (44,763)	1.25 (57,210)	1.64 (75,223)	**C**
	2.19 (100,579)	0.87 (39,971)	1.03 (47,046)	1.81 (82,682)	**G**
	3.26 (149,306)	1.58 (72,258)	1.56 (71,747)	3.30 (151,178)	**T**
C	2.35 (107,667)	1.23 (56,620)	0.98 (45,080)	1.30 (59,718)	**A**
	0.89 (40,863)	0.53 (24,251)	0.51 (23,466)	1.04 (48,067)	**C**
	1.06 (48,809)	0.49 (22,483)	0.49 (22,542)	1.05 (48,127)	**G**
	1.82 (83,359)	1.03 (47,472)	0.89 (40,874)	2.22 (101,802)	**T**
G	2.33 (106,819)	1.25 (57,183)	1.18 (54,374)	1.56 (71,505)	**A**
	0.87 (39,922)	0.59 (27,164)	0.59 (27,427)	0.86 (39,569)	**C**
	1.08 (49,451)	0.50 (23,038)	0.53 (24,480)	0.88 (40,697)	**G**
	1.66 (76,113)	1.26 (57,913)	0.97 (44,827)	1.89 (86,846)	**T**
T	2.70 (123,822)	1.95 (89,441)	1.97 (90,114)	2.72 (124,646)	**A**
	1.49 (68,080)	1.17 (53,760)	1.25 (57,529)	2.31 (105,926)	**C**
	1.29 (59,293)	0.96 (44,405)	1.24 (56,617)	2.35 (107,713)	**G**
	2.51 (115,064)	1.79 (82,073)	1.71 (78,556)	4.63 (212,186)	**T**

***Frequencies are given in percentages and the numbers in parentheses are the numbers of each nucleotide pairs.**

Appendix K: The universal genetic code table

TTT F Phe	TTC F Phe	TTA L Leu	TTG L Leu
TCT S Ser	TCC S Ser	TCA S Ser	TCG S Ser
TAT Y Tyr	TAC Y Tyr	TAA * Ter	TAG * Ter
TGT C Cys	TGC C Cys	TGA * Ter	TGG W Trp
CTT L Leu	CTC L Leu	CTA L Leu	CTG L Leu
CCT P Pro	CCC P Pro	CCA P Pro	CCG P Pro
CAT H His	CAC H His	CAA Q Gln	CAG Q Gln
CGT R Arg	CGC R Arg	CGA R Arg	CGG R Arg
ATT I Ile	ATC I Ile	ATA I Ile	ATG M Met
ACT T Thr	ACC T Thr	ACA T Thr	ACG T Thr
AAT N Asn	AAC N Asn	AAA K Lys	AAG K Lys
AGT S Ser	AGC S Ser	AGA R Arg	AGG R Arg
GTT V Val	GTC V Val	GTA V Val	GTG V Val
GCT A Ala	GCC A Ala	GCA A Ala	GCG A Ala
GAT D Asp	GAC D Asp	GAA E Glu	GAG E Glu
GGT G Gly	GGC G Gly	GGA G Gly	GGG G Gly

The table lists the three letter representation of the amino acid coded by each codon. The one letter codes as well as name of the amino acid are shown next to letter representation. "Ter" is for the stop codon.