Mining Genomes of Filamentous Ascomycetes for Phylogenetic Markers

by

Chiu-Hua Vincent Huang

A Thesis
presented to
The University of Guelph

In partial fulfilment of requirements
for the degree of
Master of Science
in
Environmental Biology

Guelph, Ontario, Canada
© Chiu-Hua Vincent Huang, June 2012
MINING GENOMES OF FILAMENTOUS ASCOMYCETES FOR PHYLOGENETIC MARKERS

Chiu-Hua Vincent Huang
University of Guelph, 2012

Advisory Committee:
Dr. T. Hsiang
Dr. G. Moreno-Hagelsieb

Sequencing technologies have improved significantly in the past 10 years and the staggering number of genome sequences available has led to a migration from single-gene phylogenetics to multigene phylogenetics. A protocol was developed here to compare fungal genomes through BLAST to determine which BLAST statistics may best represent phylogenetic information. The results suggested that levels of sequence identity, relative to the query length, may be useful for predicting whether a gene will yield a well-resolved and consistent tree. Moreover, it was found that about 40% of the genes in a typical filamentous fungal genome may lead to a well-resolved and concordant tree topology that also matched an 18S rDNA derived topology; but for consistent results, multigene trees with a minimum of five genes should be used. An additional script to rapidly identify regions within genes that can be easily amplified was then developed and tested on eight genes. The genes were successfully amplified and several resultant amplicon trees matched the 18S rDNA topology.
Acknowledgements

I would like to thank Dr. Tom Hsiang for giving me this opportunity to obtain my MSc. Without his guidance, patience, knowledge, and ability to put up with me, I probably would have given up on my thesis a long time ago. I am also grateful for the insights and knowledge of the other member of my advisory committee, Dr. Gabriel Moreno-Hagelsieb. He provided me the motivation to learn more programming languages and improve my scripting knowledge. I also appreciated the photomicroscopy devices lent to us by Dr. George Barron, the new perspectives on evolution from Dr. Jack Trevors, and of course, their amusing banter. I am also extremely thankful for Sharcnet and Westgrid putting up with my outrageous computational demands.

I am appreciative of all of the advice provided to me by my labmates, Linda Jewell, Amy Shi, and Brady Nash. I would also like to extend my thanks to all of my friends for staying sane with me as we meander through our careers in science: Eric Chan, Derek Chau, Mathew Chui, Lincoln Lam, Veny Li, Andrew Ma, Karen Po, Dorothy Truong, and Leo Wan.

I am very grateful to my family for always being there and their continuous support. Finally, I would like to thank Alice Chen. Her love and endless support has allowed me to see this thesis to the end.
## Table of Contents

Acknowledgements ........................................................................................................................ iii

Table of Contents ........................................................................................................................................ iv

List of Tables ............................................................................................................................................... viii

List of Figures ............................................................................................................................................... x

List of Appendices ....................................................................................................................................... xiii

List of Abbreviations and Acronyms ........................................................................................................ xv

Chapter 1: Literature Review ..................................................................................................................... 1

1.1 Introduction ........................................................................................................................................... 1

1.1.1 History of Phylogenetics ............................................................................................................. 1

1.1.2 Molecular Characters .................................................................................................................. 2

1.1.3 Ribosomal DNA .......................................................................................................................... 4

1.1.4 Currently Used Genes in Phylogenetics .................................................................................. 5

1.2 Tree-Building Algorithms ................................................................................................................... 6

1.2.1 Maximum Parsimony ................................................................................................................. 6

1.2.2 Distance Methods ........................................................................................................................ 8

1.2.3 Statistical Methods ...................................................................................................................... 9

1.2.3.1 Nucleotide Substitution Models ......................................................................................... 10

1.2.4 Bootstrapping ............................................................................................................................. 11

1.2.5 Other Statistical Support ............................................................................................................ 12

1.2.6 Choice of Tree Algorithm .......................................................................................................... 13

1.3 Phylogenetic Analysis ........................................................................................................................ 14

1.3.1 Choice of Genes in Phylogenetics .............................................................................................. 14
Chapter 2: Development of a Comparative Fungal Genomics Method to Identify Phylogenetically Informative Genes

2.1 Introduction ................................................................................................................. 29

2.1.1 Background............................................................................................................ 29

2.1.2 Data Selection ....................................................................................................... 30

2.1.3 Choice of Phylogenetic Method ........................................................................... 32

2.1.4 Objectives ............................................................................................................. 34

2.2 Methods ...................................................................................................................... 35

2.2.1 Genome Selection and BLAST Databases ............................................................... 35

2.2.2 Comparison of Neurospora crassa against 48 Genomes ....................................... 36

2.2.3 Initial Gene Selection for Single Gene Trees ........................................................ 37

2.2.4 Identifying the Reference Tree Based on the Most Common Topology ............... 38

2.2.5 Gauging Resolution of Trees ................................................................................. 38

2.2.6 Comparisons of BLAST Statistics ........................................................................ 39

2.2.7 Mean Identity Recalculation with Eight Query Genomes ...................................... 40

2.2.8 Comparison of Commonly-Used Genes to the Reference Topology .................... 41
2.3 Results and Discussion

2.3.1 Selection of Genomes

2.3.2 Parsing BLAST Results

2.3.3 Single-Gene Tree Generation, Tree Topologies, and Resolution Evaluation

2.3.4 Selection of BLAST Statistics to Estimate Phylogenetic Information

2.3.5 Comparison of Mean Identities Calculated with Multiple Genomes

2.3.6 Potential Sources of Variations

2.3.7 Single-Gene Trees and the Reference Tree

Chapter 3: Comparisons of Superalignment Trees to Single-Gene Trees

3.1 Background

3.1.1 Introduction

3.1.2 Multigene Trees

3.1.3 Objectives

3.2 Methods

3.2.1 Random Gene Selection for Multigene Trees

3.2.2 Superalignment Tree Construction

3.2.3 Giant Superalignment Tree Construction

3.3 Results and Discussion

3.3.1 Superalignments Trees from Random Sets of Homologous Genes

3.3.2 Comparisons of 100-Gene Superalignment Trees

3.3.3 Superalignment Trees and Single-Gene Trees

3.3.4 Comparisons of Trees to 18S

3.3.5 What is the Optimal Number of Genes for a Multigene Tree?
Chapter 4: Selecting and Testing Potentially Phylogenetically Informative Genes

4.1 Introduction

4.1.1 Background

4.1.2 Objectives

4.2 Methods

4.2.1 Gene Screening and Primer Design

4.2.2 Fungal Cultures and DNA Extraction

4.2.3 PCR Amplification and Sequencing

4.2.4 Phylogenetic Tree Construction

4.3 Results

4.3.1 Primer Screening Script

4.3.2 Gene Selection and Screening

4.3.3 Primer Test and Phylogenetic Analyses

4.4 Discussion

4.4.1 Range of Mean Identity Selected for Priming Region Screening

4.4.2 Mean Identity and Screening Process

4.4.3 Mean Identity and Ease of Primer Design

4.4.4 Comparisons of Trees Generated using Amplicons and Whole Genes

4.4.5 Comparison of Primer Development Method to Recent Studies

4.5 Conclusions

Chapter 5: General Discussion

References

Appendices
List of Tables

Table 1.1. Number of sequences found on the GenBank non-redundant database (NR) for each of the specified genomic regions in Fungi and in Ascomycetes. Search was performed in April 2012................................................................. 24

Table 1.2. Number of studies published citing the different tree-building algorithms within the last 20 years as listed in the Web of Science (http://apps.webofknowledge.com/). Some studies may have used multiple algorithms, thus appearing in more than one query. ......................... 25

Table 1.3. Commonly used nucleotide substitution models, their relative base frequencies and rates of substitutions. Frequencies of nucleotides in the models are represented by F. (Page and Holmes 1998)......................................................................................................................... 26

Table 1.4. Different versions of the BLAST algorithm have been developed for different query and database types. Translations from nucleotides into amino acid sequences are performed in all six frames. (NCBI 2010).................................................................................................................. 27

Table 2.1. Sequenced filamentous ascomycete genomes used in this study with its taxonomic class and order, file size, number of genes, date downloaded, source, and version number. ...... 58

Table 2.2. Gene identifier, species, and class of the 45 fungal sequences used to generate the 18S ribosomal DNA tree. Sequences were obtained from Genbank (http://www.ncbi.nlm.nih.gov/genbank/). .................................................................................... 59

Table 2.3. Number of genes out of 165 genes sharing the same Neighbor-Joining (NJ) tree topologies based on manual evaluations of trees constructed from alignments of 49 sequences using ClustalX 2.0 and MEGA 5 with 1,000 bootstrap replicates. Trees generated using MEGA used a maximum composite likelihood model. Multiple sequence alignments were performed using MUSCLE. Numbers of classes in the tree topologies are also indicated. The unique and unresolved category contained tree topologies that either showed a large polytomy or only occurred once out of all 165 genes examined. The remaining topologies that occurred more than once but not as often as topologies 1, 2, and 3 are in the ‘others’ category. ......................... 60

Table 3.1. Parameters used in the Maximum Likelihood (ML) analyses for the superalignment generated from the eight genomes mean identity bins under the General Time Reversible plus gamma (GTR+Γ) model as estimated by RAxML. ................................................................. 81

Table 3.2. The number of superalignment Maximum Likelihood (ML) trees that matched the reference T1 (18S) topology. A total of 60 trees were constructed using up to 15 genes randomly selected from the set of 3,584 Neurospora crassa genes that have a homolog in the 48 other filamentous ascomycete genomes. All ML trees have 49 species and were constructed in RAxML with 1000 BS replicates under the General Time Reversible plus gamma (GTR+Γ) model and a random seed of 123. ................................................................. 83
Table 3.3. Number of trees sharing the same tree topology based on manual evaluations of phylogenetic trees. Single-gene trees were constructed using ClustalX with 1000 bootstrap replicates from multiple sequence alignments generated through MUSCLE followed by manual trimming in Bioedit. Maximum Likelihood (ML) trees were constructed using RAxML under the General Time Reversible plus gamma (GTR+Γ) with 1000 bootstrap replicates and a random seed of 123. Sequence alignments for the ML trees were performed using MAFFT-EINSI and then trimmed with TrimAl using a gap threshold of 0.75. The unique and unresolved category contained tree topologies that either showed a large polytomy or only occurred once out of all genes examined. The remaining topologies that occurred more than once but not as often as topologies 1, 2, and 3 were in the ‘others’ category. ................................................................. 84

Table 4.1. Forward and reverse oligonucleotide primers designed for the eight genes using the BuildOligos.pl script. Fragment lengths were calculated relative to the *Neurospora crassa* sequence data. Primers were initially tested with an annealing temperature of 55ºC and based on results of the gel electrophoresis; the annealing temperature was adjusted accordingly. The indicated annealing temperatures are the final optimized annealing temperatures used for the PCR. ........................................................................................................................................... 103

Table 4.2. Out of the original 165 genes used to determine a measure of phylogenetic potential in Chapter 2, five genes from each of the five percent bins between 50% and 90% were randomly selected to be screened by a primer-development script. Of these 40 genes, 21 genes passed the threshold requirements of the screening script BuildOligos.pl, and a single gene from each bin was randomly selected for the actual primer development. ................................................................. 104

Table 4.3. Eight sets of oligonucleotides were tested by performing PCR on each of the class representatives in the table. If the gene amplification with the primer pairs was successful, one amplicon from each of the four classes was submitted for sequencing. ................................................. 105
List of Figures

Figure 1.1. The eukaryotic ribosomal DNA (rDNA) gene cassette showing the relative positions of 18S small subunit (SSU), 5.8S, 28S large subunit (LSU), and the internal transcribed spacers (ITS1 and ITS2). Figure was adapted from Hillis and Dixon (1991). .......................................................... 28

Figure 2.1. Examples of resolution score assignments based on 4 classes distributed amongst two nodes. If the tree topology matches the reference topology (T1), then a value of ‘1’ is assigned as the resolution score. For every clade that is not resolved like in T1, a score of 1/4 is deducted. For example, if there are two disrupted clades because a species was placed into a clade that does not agree with T1 (D2 topology), then the tree is assigned a value of ‘2/4’. If a species was placed outside of all the clades including the one it was placed within in T1 (D1 topology), only a single clade is disrupted and the tree is assigned a value of ‘3/4’. If there is only one major node, and the tree shows a star topology (R0), the tree is assigned a value of ‘0’. ...................... 61

Figure 2.2. An example of the mean identity recalculations based on the example BLAST result found in Appendix 2.2. A query sequence 568 residues in length was compared using tBLASTn against the predicted gene database of the genome, and in the top hit returned, there were three HSPs in the subject sequence. The HSPs returned three identity values: 84%, 72%, and 58%. Because the HSPs are not reflective of the overall identity between the query and subject and cannot be compared between BLAST jobs as the lengths of the HSPs may not be constant across comparisons, a Perl script using the MapTiling module was used to stitch the HSPs together and recalculate the identity. The script adds up the number of identical residues found in the HSPs, account for any overlaps that may have occurred, and then calculate the identity relative to the length of the query sequence. In this example, the three HSPs do not overlap and the sum of their identities was 345. Dividing the sum of the identity against the query length of 568 yielded an identity value of 60.7%. .............................................................................................. 62

Figure 2.3. Flowchart of the steps in the comparative genomics method used in this study to identify phylogenetically informative genes. ................................................................. 63

Figure 2.4. Unrooted single-gene distance tree generated with the Neighbor-Joining algorithm implemented in ClustalX 2.0 using 49 filamentous ascomycetes. The class level topology of this tree, named T1, was the most common topology appearing 67 times out of the 165 genes examined. Resolution scores were assigned to each of the 165 single-gene trees based on similarities or deviations from the class-level topology as depicted in T1. The Sordariomycetes and Leotiomycetes were sister groups while the Dothideomycetes and Eurotiomycetes were sister groups. Class nodes and orders nodes are marked on the tree. Scale bar represents the number of nucleotide substitutions per site. ................................................................. 64

Figure 2.5. Unrooted single-gene distance tree generated with the Neighbor-Joining algorithm implemented in ClustalX 2.0 using 49 filamentous ascomycetes. The class level topology of this tree, named T2, was the second most common topology appearing 18 times out of the 165 genes examined. In the T2 topology, the Leotiomycetes have been placed within the Eurotiomycetes. However, in this specific tree, the Leotiomycetes split the Eurotiales and the Onygenales and the node has been indicated with a circle. Not all trees sharing this topology have the circled node in
the same place as some trees placed the node within the Eurotiales. Scale bar indicates the number of substitutions per site. .................................................................

Figure 2.6. The single-genome mean identity score was calculated by comparing each of the 9,839 Neurospora crassa genes against 48 other filamentous ascomycete genomes separately in tBLASTn. Subsequently, 165 genes were chosen for phylogenetic analyses. This histogram shows the relative distribution of the 165 genes among five resolution states within 20 equally divided mean percent identity bins between 0%, in which no sites were identical, to 100% where all sites were identical (0-5%...95-100%). Resolution states of the selected genes correspond with the number of resolved classes out of the four expected classes of T1 and are represented as different shadings. The greatest amount of resolution is four resolved classes out of the four expected classes. The numbers of genes in each of the mean identity bins are indicated above the graph. ............................................................................................................................................

Figure 2.7. Comparison of the distribution of genes among mean identity bins as increasingly more query genomes were used to calculate the mean identity of each gene. One to eight genomes were used to calculate the mean identity. Mean identities were retrieved by parsing BLAST outputs with a BioPerl script and the Data Analysis package in Excel 2007 was used to calculate the histogram values. The bin width for the histograms was 4% mean identity. The single genome mean identity bins show a bimodal distribution with a peak around a mean identity of 14, and a smaller peak at around 34-38. The addition of a second genome pulled the left side of the distribution in and the peak is shifted to a mean identity of 34. As increasingly more genomes were included in the calculation for mean identity, the distribution becomes right skewed around a mean of 34............................................................................................................................................

Figure 2.8. The eight-genome mean identity score was calculated by comparing each of the 9,839 Neurospora crassa genes and their homologs in seven other genomes against the corresponding 48 other filamentous ascomycete genomes separately in tBLASTn. The eight-genome mean identity values were then assigned to the 165 genes that were previously chosen for phylogenetic analyses. This histogram shows the relative distribution of the 165 genes among five resolution states within 20 equally divided mean percent identity bins between 0 and 100%. Resolution states of the selected genes correspond with the number of resolved classes out of the four expected classes of T1 and are represented as different shadings. The greatest amount of resolution is four resolved classes out of the four expected classes. The numbers of genes in each of the mean identity bins are indicated above the graph..............................................................

Figure 3.1. Flowchart of the steps used in this study to generate and compare superalignment trees made with various numbers of genes. ........................................................................................................

Figure 4.1. Three topologies were identified from the eight Neighbor-Joining trees that were generated from the alignments of the amplicons using ClustalX v2.1 with 1000 bootstrap replicates. For each gene, the four amplicons were aligned using EINSI in MAFFT v6 and manually trimmed in BioEdit 7.0.5.3. A) Five of the eight trees showed the reference T1 topology as determined in Chapter 2. This topology occurred the most frequently out of the single-gene NJ trees (67/165) and was used as the reference topology. B) Two of the eight trees showed the T3 topology. This topology was the third most common topology out of the 165
single-gene NJ trees. It grouped Eurotiomycetes and Leotiomycetes together and
Dothideomycetes and Sordariomycetes together. C) One of the eight NJ trees generated showed
this topology, which occurred only eight times out of the 165 single-gene NJ trees. In this
topology, Leotiomycetes is grouped with Dothideomycetes while Eurotiomycetes and
Sordariomycetes are grouped together............................................................................................................. 106

Figure 4.2. The resulting topologies of the eight genes from the Neurospora crassa genome that
were selected for primer design based on the primer screening process and their single-gene tree
topology as determined in Chapter 2. The single-gene trees from Chapter 2 were generated using
49 sequences, while the amplicon trees were generated using only four sequences. The list has
been sorted according to ascending mean identity. Out of the eight amplicon trees, five yielded
evolutionary relationships that matched T1. The four classes are represented by the first letter of
their names (E = Eurotiomycetes; D = Dothideomycetes; S = Sordariomycetes; and L =
Leotiomycetes)............................................................................................................................................. 107
List of Appendices

Appendix 2.1. This bash script was used on Sharcnet to generate the file used by mpiBLAST to find the location of the database files and also generate the mpiBLAST databases in the folder. mpiBLAST database files were split into 30 pieces and allows BLAST to execute on 30 cores. Fasta files used to generate the databases were stored in /work/$USER/hound/mpiBLAST/database. This script requires an installation of mpiBLAST. ................................................................. 127

Appendix 2.2. This is an example BLAST output from MPIBLAST. .................................................. 128

Appendix 2.3. This Perl script was used to parse text formatted BLAST output files into BLAST statistics of interest in tab-delimited format. ................................................................. 132

Appendix 2.4. This Perl script was used to condense all the parsed BLAST files into a single table to make viewing easier. Downstream calculations such as means and geometric means were performed using the output of this script. ................................................................. 134

Appendix 2.5. This script was used in part to compile the mean identity across several genomes for Neurospora crassa. .................................................................................................................. 136

Appendix 3.1. Top 25 genes from Neurospora crassa (version 4 from Ensemble) with the highest 8-genome mean identity in the filamentous ascomycetes. Mean identities were calculated using 8 genomes as compared to 48 other filamentous ascomycete genomes using tBLASTn. Identity values are relative to the length of the query sequence. ................................................................. 137

Appendix 3.2. This Perl script was used to concatenate the alignments of genes into superalignments containing 5, 10, 15, 20, and 25 genes. ................................................................. 138

Appendix 3.3. This Perl script was found online and adapted to convert all Fasta files with a specific extension in the folder into a relaxed Phylip format (no ten character limit for names). .................................................................................................................. 140

Appendix 3.4. This Perl script was used to concatenate the alignments of genes into 100 gene superalignments. .................................................................................................................. 141

Appendix 3.5. This Perl script was used to concatenate all alignments in a directory into a single superalignment. .................................................................................................................. 143

Appendix 3.6. This script was used to randomly select genes for the control superalignment trees. The script reads a list of gene, and randomly selects a specified number of genes without replacement. The number of independent replicates can also be specified. .................................................. 145

Appendix 3.7. An example of a multiple sequence alignment (A) before being trimmed by TrimAl v1.2 with a gap threshold of 0.75 and (B) after. ................................................................. 146
Appendix 3.8. This script was used to randomly select a set of genes and outputs the mean of mean identities. The script reads a list of gene, and randomly selects a specified number of genes without replacement. The mean of mean identities of the selected genes is then calculated and outputted. The number of independent replicates can also be specified. ................................. 149

Appendix 3.9. Computational Constraints Encountered in Chapter 3 ........................................ 151

Appendix 4.1. This sample Perl script was used to identify potential priming sites in multiple sequence alignments. This script requires the TrimAL program. .............................................. 153

Appendix 4.2. Forward and reverse priming regions for the *N. crassa* gene NC001938 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below. ........................................................................................................................................ 158

Appendix 4.3. Forward and reverse priming regions for the *N. crassa* gene NC003210 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below........................................................................................................................................ 159

Appendix 4.4. Forward and reverse priming regions for the *N. crassa* gene NC004139 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below........................................................................................................................................ 160

Appendix 4.5. Forward and reverse priming regions for the *N. crassa* gene NC004477 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below................................................................. 161

Appendix 4.6. Forward and reverse priming regions for the *N. crassa* gene NC006543 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below................................................................. 162

Appendix 4.7. Forward and reverse priming regions for the *N. crassa* gene NC006868 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below................................................................. 163

Appendix 4.8. Forward and reverse priming regions for the *N. crassa* gene NC007341 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below........................................................................................................................................ 164

Appendix 4.9. Forward and reverse priming regions for the *N. crassa* gene NC008393 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below........................................................................................................................................ 165
List of Abbreviations and Acronyms

rDNA: ribosomal DNA
ITS: internal transcribed spacers
EF1-α: Elongation Factor 1 Alpha
RPB1: RNA polymerase 2, largest subunit
RPB2: RNA polymerase 2, second largest subunit
MP: Maximum parsimony
NJ: Neighbour Joining
ML: Maximum likelihood
pp: posterior probability
HPC: High Performance Computing
JC: Jukes-Cantor
LBA: Long Branch Attraction
NCBI: National Center for Biotechnology Information
BLAST: Basic Local Alignment Search Tool
PSI-BLAST: Position-Specific Iterative BLAST
PHI-BLAST: Pattern-Hit Initiated BLAST
E-value: Expectation Value
OTU: Operational Taxonomic Unit
SHARCNET: Shared Hierarchical Academic Research Computing Network
MPI: Message Passing Interface
β-tub: Beta Tubulin
PCR: Polymerase Chain Reaction
FunyBASE: Fungal Phylogenomic Database
bp: base pair
PDA: Potato Dextrose Agar
TBE: Tris-Borate-EDTA
v/v: volume per volume
w/v: weight per volume
Ti: transitions
Tv: transversions
Chapter 1: Literature Review

1.1 Introduction

1.1.1 History of Phylogenetics

In the past, the study of relationships between organisms was separated into at least two major schools of thought: cladistics, in which organisms are grouped together based on shared derived characteristics while taking evolution into consideration (Hennig 1966); and phenetics, by which organisms are grouped based on their degree of relatedness inferred from differences without assumptions regarding evolution and homology (Sneath and Sokal 1973). Cladistics has been considered to be more reliable than phenetics for inferring evolutionary relationships between organisms, and has been the dominant ideology behind modern phylogenetic studies. (Livezey 2011; Lukhtanov 2010; Williams and Ebach 2009) However, phenetic methods such as distance tree algorithms are being increasingly used in evolutionary studies because of their ease of use and relative low computational demands.

Studies of evolutionary relationships generally employed methods that involved physical and tangible traits (Giribet et al. 2010). For example, the appearance of a specimen, or additional characteristics such as scent, texture, and taste. Often, these observations were related to specific morphological characters of the organisms of interest (Giribet et al. 2010). By identifying similar characteristics between organisms, an estimate of their evolutionary relationships could be established (Delsuc et al. 2005; Hennig 1966; Page and Holmes 1998). Based on these estimates, the genealogy or tree of the order in which organisms evolved and speciated over time could be inferred and examined in the field known as phylogenetic systematics (Hennig 1966; Page and
Holmes 1998). For example, the presence of a backbone is a trait shared by mammals, while bees and ants can be inferred to be related as they have three pairs of legs.

Morphological characters are useful at higher taxonomic levels, but at lower taxonomic levels in some kingdoms, there are fewer differences in morphology and they may not be as distinctive. For example, distinguishing different species or subspecies of fungi can be difficult because they may look very similar, or because the ranges of variation overlap considerably. This morphological problem is further exacerbated by the limited number of distinctive morphological features for certain taxa (Hillis and Dixon 1991). One solution is to descend to the microscopic level and examine the traits found there, but doing so only adds a limited set of morphological information for taxa such as fungi which show limited morphological differentiation.

1.1.2 Molecular Characters

For some taxa, morphological characteristics have largely been replaced by molecular data such as DNA and protein sequences (Hillis and Dixon 1991). Over evolutionary time, morphological changes that accumulated are expected to be reflected in the DNA sequence, the ‘blueprint’ of the organism, but changes in the DNA may not always appear as morphological differences. Thus, there may be numerous molecular differences, limited only by the number of sequence positions that can diverge across organisms.

In eukaryotes, DNA can be found coiled into chromosomes, which can be further divided into non-coding genes, protein-coding genes, regulatory regions, intergenic regions and structural regions (Alberts et al. 2002). A particular region can be used in phylogenetics if the same region can be identified across different organisms of interest, and depending on how conserved some regions are, they may be more useful than others for different taxonomic levels.
Amino acid or protein sequences are another set of molecular data useful for phylogenetics (Page and Holmes 1998). These are translated from protein-coding genes and may provide more phylogenetic information than DNA sequences alone. An advantage protein sequences have over nucleotide sequences is that there are 20 possible amino acids per position as opposed to four bases per position in DNA (Bergsten 2005; Page and Holmes 1998). Assuming equal probability for each amino-acid, two proteins will only have 1 in 20 chance of having identical amino acids at the same sites as compared to 1 in 4 chance for nucleotides. As additional sites are considered, identical matches by chance between two proteins diminish more rapidly than for nucleotides. Given a stretch of three sites, proteins have \(20^3 = 8000\) possible permutations versus the \(4^3 = 64\) in nucleotides. Therefore, because identical matches are less likely to occur by chance in protein sequences than in nucleotides, they are more likely to be a true match allowing for more accurate phylogenetic analyses. This can potentially affect the choice of data used as there may be fewer informative sites in amino acid sequences (Townsend et al. 2008a), and for closely related taxa, nucleotides made be needed for differentiation.

Another issue that may affect the decision between using nucleotide sequences or protein sequences for phylogenetics analyses is the degeneracy of the third nucleotide position. The 20 amino acids are coded by triplets of nucleotides called codons and because there are 64 possible codons, there are more codons than there are amino acids. Therefore, there are several codons for the same amino acid. In a stretch of protein-coding DNA, the third position in the codon has a greater chance of accumulating mutations than the second position, which in turn has a higher chance than the first position. The nucleotide changes that might occur in the protein-coding regions may not result in a change in the protein sequence because of codon redundancy (i.e. multiple nucleotide triplets code for the same amino acid especially ones differing in the third
position) and are effectively silent as they do not affect the function of protein (Page and Holmes 1998). Thus, the relatively slower rate of evolution and 20 potential residues allow comparisons using protein sequences to be more objective during analyses and less prone to error due to random noise (Townsend et al. 2008a). As a result, protein sequences are generally more useful for phylogenetic analyses at higher taxonomic levels such as phyla, classes, and orders than their nucleotide counterparts. However, some protein-coding nucleotides are more informative than the protein sequences because of their relatively faster rate of evolution, and there are three times more nucleotides than there are amino acids when translated although they have more stochastic noise (Townsend et al. 2008a). In these cases, nucleotides may still outperform proteins at higher taxonomic levels simply because there are more informative sites. For example, Murphy et al. (2001) found that they were unable to completely resolve basal vertebrate relationships with 11 combined protein sequences, but were able to do so using a dataset consisting of concatenated nucleotide sequences.

1.1.3 Ribosomal DNA

One of the most commonly used genomic regions for phylogenetics and species identification is ribosomal DNA (rDNA) (Hillis and Dixon 1991). Ribosomal DNA, when transcribed, produces ribosomal RNA, which is part of the cellular machinery responsible for translating messenger RNAs into amino acids. Because this function is necessary for life, rDNA can be found in all living organisms (Hillis and Dixon 1991). Another reason for the popularity of rDNA for phylogenetic studies is its ease of amplification due to its occurrence as multiple copies in tandem repeats throughout the genome. The number of rDNA arrays per genome has been shown to be between 39 and 19,300 in animals, 150 to 26,048 in plants (Prokopowich et al. 2003), and 50 to 150 in ascomycetous fungi (Ganley and Kobayashi 2007).
In eukaryotes, rDNA can be classified into several types (Hillis and Dixon 1991): the large subunit 28S, the small subunit 18S, 5.8S, and the internal transcribed spacers 1 and 2 (ITS) (Figure 1.1). Unlike most other genes, different parts of rDNA can be used for resolving phylogenies at different taxonomic levels. For example, 5.8S rDNA is highly conserved within kingdoms and thus might only be useful to differentiate between kingdoms. ITS on the other hand is highly variable and is used frequently for species level phylogenies and recently, has been designated the DNA barcode for fungal species (Doolittle 1999; Hillis and Dixon 1991; Schoch et al. 2012). The use of 18S rDNA has a long history in molecular phylogenetics, and it was the first region used to separate Bacteria from Archaea (Woese et al. 1990). A gene tree is only an estimate of the true species tree because it is impossible to observe evolution that occurred in the past. Thus, to examine and validate the extensive use of 18S rDNA in phylogenetics, further examination is needed.

1.1.4 Currently Used Genes in Phylogenetics

Although select parts of the rDNA region are popular for phylogenetic studies because of their ease of amplification (Schoch et al. 2012), they might not be able to fully clarify the evolutionary relationships of some taxonomic groups because there may not be enough information at deeper taxonomic levels (Berbee and Taylor 2001). Thus, well-conserved protein-coding genes have also been used for phylogenetic resolution. Several commonly used genes in fungal systematics include: translation elongation factor 1 alpha (EF1-α) (Roger et al. 1999), which is involved in mRNA translation; RNA polymerase II largest subunit (RPB1) and RNA polymerase II second largest subunit (RPB2) (Liu et al. 1999), which are needed for mRNA transcription; and beta tubulin (Thon and Royse 1999), a cytoskeletal scaffolding protein. These essential genes have been used in conjunction with rDNA to resolve the phylogeny of various
clades in the ascomycetes with strong statistical support (James et al. 2006; Schoch et al. 2009a; Schoch et al. 2009b; Spatafora et al. 2006). As seen in Table 1.1, many sequences for each of the commonly used genes are available on GenBank, but the most abundant appears to be the rDNA region with almost ten times more than the next most uploaded gene.

There are still issues associated with the use of protein-coding genes for phylogenetic studies. For example, the few genes that are currently used for phylogenetics often yield conflicting single-gene tree topologies. Given that the true species tree is not known, it is difficult to separate the genes that carry the true evolutionary signals for the species tree from the genes that may have confounding signals arising from other forces and events. Thus, different studies have been conducted to determine what may be representative of the true species tree, and what other genes can be used for phylogenetics. For example, Townsend et al. (2008) developed a method to quantify the phylogenetic signals in a gene, and based on this information, phylogenetically informative genes can be selected. Aguileta et al. (2008) combined genomic information to produce a tree, and suggested that individual genes that yielded trees matching their genome tree could be used for phylogenetics. In a recent study, 13 genes useful for yeast phylogenetics were identified through a computational pipeline that compared distance trees generated from individual genes in genomes to a reference genome tree of the yeasts (Ramazzotti et al. 2012).

1.2 Tree-Building Algorithms

1.2.1 Maximum Parsimony

Many algorithms can be used to generate a tree from phylogenetic datasets and each has strengths and weaknesses. Maximum parsimony (MP), originally one of the more common methods used for phylogenetics, is based on the same principle as Occam’s razor (Forster 2000;
Nakhleh et al. 2005; Swofford and Sullivan 2003). Occam’s razor states that the simplest solution is most likely the correct solution. Thus, identifying the tree with the shortest number of steps that can explain the dataset will most likely yield the correct tree (Page and Holmes 1998; Swofford and Sullivan 2003).

An exhaustive MP search is possible for datasets containing few taxa or with a limited number of characters (Page and Holmes 1998). However, as more taxa are added, the tree search time increases rapidly as it considers all possible trees for the given data set. The number of possible trees can be calculated using $N_{\text{rooted}}=\frac{(2n-3)!}{(2^{n-2})(n-2)!}$ for rooted trees and $N_{\text{u}}=\frac{(2n-5)!}{(2^{n-3})(n-3)!}$ for unrooted trees, where ‘$n$’ is equal to the number of taxa in the dataset (Felsenstein 2004). For example, with 20 taxa, there are $2.21\times10^{20}$ possible unrooted trees. If it takes two seconds to evaluate each tree, that means it will require about $1.4\times10^{13}$ years to finish. Alternative MP methods have been developed to avoid the issues of exhaustive searches by using heuristic algorithms (Moore et al. 1973; Swofford and Sullivan 2003; Takahashi and Nei 2000). In addition, there may be multiple ‘best’ possible trees and, therefore, either additional evidence should be gathered to identify the tree that is most likely to be true or a consensus tree should be used (Page and Holmes 1998).

MP differs considerably from other methods in that it does not consider every single character in the dataset (Page and Holmes 1998). Characters that are conserved across all taxa being examined do not offer any information on how the taxa are split and are not used in the analyses. Likewise, characters in which a change has occurred in a single taxon, autapomorphies, are also not useful. The remaining characters are the ones where there are different character states in at least two taxa that can be used to group taxa into clades (Page and Holmes 1998). Grouping organisms based on these shared derived traits (synapomorphies) is the hallmark of a
cladistics method. Unlike the other methods where all sites are evaluated and used for tree generation, MP only considers the informative characters, which may leave the bulk of the dataset unused (Page and Holmes 1998).

1.2.2 Distance Methods

For phylogenetic analyses, it is also possible to convert character matrices or multiple sequence alignments into numerical values to represent the genetic distance between taxa (Delsuc et al. 2005). The greater the distance between a pair of taxa, the more evolutionarily distinct they are from each other and by minimizing the distances between a group of taxa, a phylogeny can be constructed (Saitou and Nei 1987). In datasets that contain closely related taxa, distance methods such as Neighbour-Joining (NJ) can show strong agreement with other tree-building methods (Kuhner and Felsenstein 1994; Takahashi and Nei 2000). When examining higher taxonomic levels or older lineages, relationships may not be properly resolved or inferred by distance methods (Holder and Lewis 2003).

Distance methods are generally faster than other methods given the same dataset (Gabaldon 2005; Holder and Lewis 2003). Speedups are achieved by converting the data into numerical values but at the cost of losing additional and potentially useful data such as derived character states (apomorphies) and ancestral character states (plesiomorphies) (Page and Holmes 1998). The conversion to a numerical value also prevents differentiation of homologous or coincidentally matching (homoplasious) characters, and does not take into account back mutations and multiple mutations at the same sites (Holder and Lewis 2003). This discarding of evolutionary information is a classical example of the phenetic school of thought in which groups of organisms are established based on their numerical differences rather than their similarities. However, this phenetic method has found use in modern phylogenetics for
generating quick phylogenetic tests that can give an idea of what the tree may look like and is also used by more computationally intensive and statistical algorithms as initial starting trees (Holder and Lewis 2003).

### 1.2.3 Statistical Methods

Statistical methods are often recommended for performing phylogenetic analyses as they are considered more accurate than either distance and parsimony (Holder and Lewis 2003; Whelan et al. 2001). Two statistical methods are currently dominant in phylogenetics: Maximum Likelihood (ML) and Bayesian (Table 1.2). Current ML programs utilize a heuristic approach by generating a tree and evaluating how well the data fits the tree given an evolutionary model (Page and Holmes 1998). Subsequently, the tree is altered and the data is evaluated again. This process reiterates until the highest scoring tree, given the model, is found. The Bayesian analysis differs from ML in the way it estimates probability, and what the probability values mean (Holder and Lewis 2003; Huelsenbeck et al. 2001). When starting a Bayesian analysis, all possible trees are usually given the same *a priori* probability unless there is previous knowledge of what the tree may look like. A random tree is selected from the search space and the *posterior* probability (PP) of the tree given the data and evolutionary model is then calculated (Holder and Lewis 2003; Huelsenbeck et al. 2001). The tree is altered and re-evaluated in terms of PP (Holder and Lewis 2003; Huelsenbeck et al. 2001). The PP contrasts with likelihood values in that PP is the probability of the model given the data and is also calculated from likelihood, whereas likelihood represent the probability of the data given the model.

Statistical methods are more computationally intensive, and consequently more time consuming than MP and distance methods (Holder and Lewis 2003). Evaluating likelihood and probability values requires more computational power than calculating sums, but in recent years,
statistical methods are becoming increasingly popular because of improvements in computer technology. Analyses that once took months to complete can now be performed in several days. Runtime can be further reduced by utilizing specialized high performance computing (HPC) clusters.

1.2.3.1 Nucleotide Substitution Models

In order for some tree-building methods to be accurate, substitution models or ‘evolutionary’ models must be considered (Page and Holmes 1998). In the most basic model, rates of mutations between nucleotides are often assumed to be the same. For example, the probability of an A to be replaced by a T is considered equal to that of an A being replaced by a G (Page and Holmes 1998). This has been demonstrated to be biologically unrealistic (Page and Holmes 1998). Rates of mutation among the four bases are different due to the chemical differences between the bases. The four bases can be classified by their chemical structure into purines (A, G), and pyrimidines (C, T). Mutations that cause a purine to mutate into another purine or a pyrimidine into a pyrimidine are called transitions while mutations that exchange a purine with a pyrimidine or vice versa are called transversions (Collins and Jukes 1994). Furthermore, although there are four ways transversions can occur, and only two ways for transitions to occur, transversions have been shown to occur less often than transitions (Collins and Jukes 1994; Yang and Nielsen 2000).

In evolutionary models, base compositions are assumed to remain constant over time. This assumption may not be completely accurate, but allows for simplifications of the phylogenetic algorithms. In a strand of DNA, bases may not be divided equally into A, C, T, and G’s. For example, in heat tolerant organisms, GC content tend to be higher because these bases form one more hydrogen bond than ATs, and thus are more stable under extreme conditions
If base compositions are not taken into consideration, phylogenetic analyses may group taxa with similar GC content together. Moreover, by combining mutations rates and base compositions, the number of mutations a site may have accumulated can be estimated. This addresses the problem of back mutations and multiple mutations at the same site (Page and Holmes 1998).

There are several commonly used substitution models (Table 1.3). The simplest model with the fewest parameters is the Jukes-Cantor (JC) model which assumes equal frequencies of bases and identical substitution rates for all possible substitutions (Jukes and Cantor 1969). This simplified model allowed for easier calculations without the need for computers. However, biological systems are rarely that simple, and the nucleotide model with the greatest biological significance is most likely the General Time-Reversible model (Page and Holmes 1998). When performing ML or Bayesian analyses, using an incorrect or less than optimal substitution model may lead to false phylogenetic inferences. Therefore, it is often recommended to perform a model test to determine the best substitution model to use for the ML or Bayesian analyses (Posada 2008; Posada and Crandall 2001).

1.2.4 Bootstrapping

A popular method of adding statistical support to phylogenetic trees is to perform non-parametric bootstrapping. When bootstrapping is performed on a character matrix or sequence alignment, replicates of the dataset are generated by sampling with replacement (Felsenstein 1988). If the dataset is intended for distance methods, replicates are then converted to a distance matrix. From each bootstrap replicate, a tree is generated and used to evaluate the branch scores on the original tree. Bootstrap support scores represent the number of times the same branch (or the same tip taxa) appeared in all bootstrap replicates (Felsenstein 1988). For example, a branch
with a BS score of 0.5 means that the branch appeared in half of the bootstrap replicates. The higher the bootstrap value, the more statistical support the node has; however, a branch with high bootstrap score does not indicate the branch itself is real in terms of the true species tree, but is well supported in terms of the data.

1.2.5 Other Statistical Support

Aside from bootstrap replicates, which has become one of the most popular methods for estimating branch support (Archibald et al. 2003), there are other methods that can be used to add statistical support to the trees such as jackknife replicates and Bremer support. Similar to bootstrap replicates, jackknifing generates smaller replicates from the original dataset by sampling without replacement and is useful for examining the variance in the dataset whereas bootstrap generates replicates that are the same size through sampling with replacement. Bremer support or decay indices refer to the number of changes in the dataset that is required for a branch to be lost from the consensus tree (Bremer 1994). The larger the decay index, the stronger the data supports the branch (Bremer 1994).

Unlike other commonly used tree-building algorithms, one of the advantages of Bayesian analyses is that it can generate its own ‘support’ values in the form of PP. When compared to bootstraps, PP appears to be more ‘liberal’ within the estimates while the bootstrap seems more ‘conservative’, which may lead to incorrect inferences (Alfaro et al. 2003). In contrast to bootstrap support, estimates of PP for branches take into consideration the evolutionary model, the estimated parameters, the data, and the prior probabilities of each tree in the search space, and thus, may not be directly comparable to bootstrap values (Archibald et al. 2003).
1.2.6 Choice of Tree Algorithm

In the last 20 years, rigorous tree-building algorithms have been developed to better utilize the rapidly growing computing power, but one of the major areas of contention in phylogenetics remains the choice of tree-building algorithm. Several methods have been outlined in the previous sections but it is unclear whether there is a single best method. Although distance methods, such as NJ, are fast and often thought to be less rigorous than statistical methods, they have been shown to yield equivalent results compared to likelihood and parsimony in specific cases. For example, Tateno et al. (1994) demonstrated that all NJ, MP, and ML, have the same statistical efficiency when substitution rates vary with site in datasets containing more than 1,000 nucleotides with a sequence divergence of around 5%. Moreover, Hillis et al. (1994) emphasized the importance of determining whether the evolutionary rates are appropriate for the analyses as their results showed that distance methods, ML, and MP will perform equally well if the substitution rates are appropriate.

MP was found to be highly susceptible to long branch attraction (LBA), in which rapidly evolving taxa are grouped closer than they should be due to more matches occurring by chance (Felsenstein 1978). LBA was found to occur more frequently than previously thought (Bergsten 2005), and as a result, methods to avoid this bias were developed. To prevent problems caused by LBA, rapidly evolving taxa should be omitted, more taxa should be sampled to break up the long branches, or methods should be used that are less susceptible to LBA such as ML (Bergsten 2005).

Aside from being computationally intensive and slow, ML has been shown to have difficulties dealing with site-specific evolutionary rate variations, heterotachy (Kolaczkowski and Thornton 2004). In these same cases, MP has been shown to be better for inferring trees
(Kolaczkowski and Thornton 2004). However, in more realistic simulations, both heterotachy and between-site rate variations are expected and therefore, ML will still outperform MP due to its decreases susceptibility to LBA (Philippe et al. 2005b).

There does not appear to be a single best or most preferred tree-building algorithm for phylogenetics. Simulations may not be sufficient to fully represent evolution and all its complexities, and it is not possible to definitively prove which method is the best, but it is possible to identify the most popular method. As shown in Table 1.2, distance, MP, and ML methods were almost equal in popularity between 1992 and 2002. However, in the last 10 years, there has been an overall increase in the use of phylogenetic methods, but the increase for distance methods was less than that for MP which was less than that for ML or Bayesian. The Bayesian method, which is the most computationally intensive among these, has seen the greatest increase in popularity in the last 10 years.

1.3 Phylogenetic Analysis

1.3.1 Choice of Genes in Phylogenetics

Sometimes, a single gene is not sufficient to produce a well-resolved phylogenetic tree with robust bootstrap support (Rokas et al. 2003). Incomplete lineage sorting may occur because the gene being used lacks sufficient variation to resolve the relationships at that taxonomic level (Maddison and Knowles 2006; Rannala and Yang 2008). For example, a highly conserved gene may not be good for resolving species-level phylogenies because it lacks variation between closely related species. However, the same gene might be able to resolve higher-level taxa such as kingdom, phylum or order because it has diverged sufficiently to resolve relationships but is still sufficiently conserved for more confidence in alignments. Conversely, when a poorly
conserved gene is used, the resultant phylogenetic tree also may not produce a robustly supported tree, or infer the correct relationships due to poorly aligned regions and multiple mutations at the same sites (Maddison and Knowles 2006; Page and Holmes 1998; Sanderson and Shaffer 2002).

Ideally, genes that should be used for phylogenetics are the ones that fall between the range of being ‘too conserved’ to resolve and ‘too variable’ for alignment depending on the taxonomic level under study (Rokas et al. 2002). However, methods to rapidly sort through genome data have not been well developed or are not easily accessible. Development of methods to rapidly mine genes for particular purposes will be a good step towards effective utilization of the growing amount of sequenced genomic data.

1.3.2 Phylogenomics

When a single gene is not enough, the next step is to consider phylogenomics instead of phylogenetics. Unlike molecular phylogenetics in which a single gene is used to determine the relationships between organisms, phylogenomics refers to examining multiple genes at the same time and other genome characters such as gene order and other secondary genome structures (Delsuc et al. 2005).

Analyzing multiple genes confers several benefits over single gene phylogenetics. It is possible to obtain a more strongly supported and accurate tree estimate with multiple genes (Ropiquet et al. 2009). In addition, by combining many genes, the combined phylogenetic signal may possibly overpower the noise from stochastic mutations (Rokas et al. 2003; Ropiquet et al. 2009). This can increase the reliability of the relationships inferred by the tree. However, there are some disadvantages to phylogenomics. The time and computational power required even for a single gene can be considerable depending on the number of taxa, size of the gene, and
algorithm of choice. Due to computational and time constraints, a balance must be achieved between the number of taxa and amount of data to analyze (Delsuc et al. 2005).

Another factor that must be considered is the number of sequenced genomes in the field of study. Many economically and medically important fungi have already been sequenced but the same cannot be said for insects, mammals or plants (Hammesfahr et al. 2011). Having the genome available allows genes to be selected rapidly for phylogenomics. However, if the genome has not been sequenced, then the meticulous process of synthesizing oligonucleotides and amplification would have to be performed in order to retrieve the genes of interest. Fortunately, sequencing technology has been improving at a dramatic rate. Third generation sequencing platforms have started appearing on the market, and the current ‘next generation’ platforms are still improving (Glenn 2011; SanMiguel et al. 2011). In the next few years, the sequence repositories are expected to be inundated with genomic data (Baker 2010) and large-scale phylogenomics studies will most likely replace the current small-scale phylogenetics (Rannala and Yang 2008). It may become more cost effective to obtain the rough assembly of an entire genome rather than attempting to fish out a single gene sequence that is poorly conserved using sequences from related species.

1.4 BLAST

The National Center for Biotechnology Information (NCBI) curates a gene sequence database called GenBank. The program that accompanies GenBank, which allows researchers to quickly identify sequences of interest is called the Basic Local Alignment Search Tool (BLAST) (Altschul et al. 1990). BLAST uses heuristics to quickly find high-scoring local pairwise alignments. To accomplish this, BLAST splits a query sequence into smaller ‘words’ based on the user’s preference. Larger ‘word’ sizes increases stringency while lower word sizes increases
sensitivity. These ‘words’ are matched with the database and when found, the alignment/extension step is initiated to compare the query sequence against the database hit (Altschul et al. 1990). Once the number of consecutive mismatches reaches a negative score limit, the extension step stops and the statistics of the match is calculated (Altschul et al. 1990; Altschul et al. 1997). When the entire database has been searched, it will output all hits that returned statistics above the user’s specified threshold (Altschul et al. 1990).

Unlike the Smith-Waterman algorithm, which is able to identify the best possible match within the database, the BLAST algorithm is not guaranteed to identify the best match and trades some sensitivity for increased efficiency and less computation requirement. This is especially important considering the rapidly growing size of the databases available now.

Over the years, NCBI improved upon the original BLAST program that was released in 1997. Since then, they have introduced the different algorithms such as MegaBLAST, PSI- and PHI-BLAST, gapped-BLAST, and added the statistical expect-value (Camacho et al. 2009). In the latest iteration, BLAST+, the code was rewritten to be more efficient and modular allowing for future ideas and algorithms to be implemented more easily (Camacho et al. 2009). One new feature BLAST+ has is the ability to split long query sequences into smaller pieces for processing, which can significantly decrease search time (Camacho et al. 2009).

There are several alternatives to BLAST. The FASTA package contains several programs that can be used to identify homologous sequences of interest including one that uses the Smith-Waterman algorithm (Pearson and Lipman 1988). In addition, WU-BLAST is another program similar to BLAST but was developed to allow for greater controls over the search parameters (Altschul and Gish 1996). Both alternatives have been demonstrated to outperform BLAST in
identifying divergent homologous sequences with respect to error rates (Brenner et al. 1998). However, BLAST is relatively fast and useful for rapid searches (Brenner et al. 1998).

1.4.1 Flavours of BLAST

There are several flavours of BLAST, each useful for different purposes (Table 1.4). For example, genes may be highly conserved at the amino acid level but not at the nucleotide level. Thus, a search conducted using the nucleotide sequence may not identify the same hits as a search performed using the corresponding amino acid sequence. By using BLASTx, which translates the nucleotides into the six frames of amino acids prior to BLASTing the amino acid databases, better scoring hits may be identified. If the nucleotide sequence of a specific taxon is desired from the database, but conservation is at the amino acid level, it is possible to search using the tBLASTx version. tBLASTx translates both query, and nucleotide databases into amino acids before searching (NCBI 2010). Because translation of both query and databases are performed in all six frames resulting in 36 comparisons for each query sequence and database sequence pair, this dramatically increases the search time. Position specific iterated BLAST (PSI-BLAST) and pattern hit initiated BLAST (PHI-BLAST) are specialized versions of protein BLAST that allows users to find protein motifs (Altschul et al. 1997). PSI-BLAST allows users to scan BLAST results and identify the regions of interests. This process is reiterated several times until the motif and database hits are identified (Altschul et al. 2009; Altschul et al. 1997). PHI-BLAST is similar to PSI-BLAST but users can submit a specific motif to identify in the database.
1.4.2 BLAST Statistics

When BLAST completes, it returns the set of database hits found, and their accompanying BLAST statistics. For each hit, there is a raw alignment score, the recalculated bit score, and an expectation value (e-value) (NCBI 2010). The raw alignment score shows how well the query and subject sequence match but are not comparable between BLAST jobs so it is normalized into the bit score according to the identity matrix used. E-values are further processed from the bit score while taking into account the size of the database queried, and represent the odds of obtaining a better hit in the same database by random chance alone (NCBI 2010). Thus, the closer the e-value is to zero, the more significant the BLAST hit (NCBI 2010).

BLAST also returns an identity value for each hit but the reported identity values are relative to the size of the aligned region that BLAST finds. Moreover, the identity scores has been shown to not be as useful as the other BLAST statistics for inferring homology (Brenner et al. 1998). In contrast, e-values are often used and recommended for inferring homology (Brenner et al. 1998). E-values are also comparable between different BLAST jobs because it adjusts for sequence length, identity matrix used, and database size (NCBI 2010). As a result, many studies that require homology to be inferred used e-values cutoff to separate and identify the homologous BLAST hits from the homoplasious matches. This can lead to potential problems depending on the nature of the study. To identify potential homologs between genomes, an e-value cutoff of 1E-5 was previously used by Hsiang and Baillie (2005), but in other studies, more stringent e-value cutoffs as low as 1E-30 have been used to indicate orthology (Mao et al. 2006). Depending on rates of evolution, or divergence times, it may be possible to misidentify genes as homologous or miss genes that are true homologs.
1.5 The Filamentous Ascomycetes

The Pezizomycotina, which is commonly referred to as the filamentous ascomycetes, is the largest monophyletic group in the largest fungal Phylum, the Ascomycetes (Alexopoulos et al. 1996; Spatafora et al. 2006). This subphylum of the Ascomycetes is characterized by a dominant filamentous growth phase while its sexual state produces ascospores within enclosed structures call asci (Alexopoulos et al. 1996; Schoch et al. 2009b; Spatafora et al. 2006).

Ascomycetes and Basidiomycetes last shared a common ancestor between 750 million years ago (Berbee and Taylor 2010) to 1.2 billion years ago (Blair Hedges and Kumar 2003). Within the Ascomycetes, the yeasts and the filamentous ascomycetes was estimated to have diverged between 400 million (Berbee and Taylor 2010) to one billion years ago (Blair Hedges and Kumar 2003), while the last common ancestor for the filamentous ascomycetes existed about 300 million (Berbee and Taylor 2010) to 670 million years ago (Blair Hedges and Kumar 2003).

Many filamentous ascomycetes and their anamorphs play important medical, ecological and economical roles. There are species that have been found to cause opportunistic infections in humans, whereas others are sources of antibiotics such as the cephalosporin and penicillin (Bowman et al. 1992; Ligon 2004; Macdonald et al. 1963). In addition, many Pezizomycotina species are also economically important crop pathogens that cost the agriculture and forestry industries billions of dollars every year and put many humans at risk of starvation. For example, the cost of fungicides that specifically target Botrytis was estimated to be around €500 million in 2001 (Dean et al. 2012), while the estimated annual loss of rice, the staple diet for over half of the world’s population (Khush 2005), due to Magnaporthe oryzae could feed over 60 million people (Dean et al. 2012; Zeigler et al. 1994).
1.5.1 Sequenced Fungal Genomes

Among the eukaryotes, the greatest numbers of sequenced genomes have been from the fungal kingdom (Haridas et al. 2011; Hsiang and Baillie 2004, 2005). Fungi, especially the yeasts, have relatively small genomes among all eukaryotes, which make them attractive for sequencing (Haridas et al. 2011; Hsiang and Baillie 2004, 2005). Despite the focus on sequencing human and animal yeast pathogens by the medical community, many filamentous ascomycetes have already been sequenced (Hammesfahr et al. 2011) and more will continue to be sequenced as the costs of sequencing continues to fall while sequencing technology continues to improve (Baker 2010). This growing number of fungal genomes is a potential source of data for comparative genomic research.

According to Hammesfahr et al. (2011), 232 Ascomycete genomes have been sequenced at the time of their study, which is at least 100 more than the Chordates, the second most sequenced Eukaryotic group. Additional information on sequenced genomes, genomes being sequenced, and assembly statistics can be accessed through diARK 2.0 (Hammesfahr et al. 2011), GOLD (Liolios et al. 2010), NCBI genome (Sayers et al. 2011), the National Human Genome Research Institute (http://www.genome.gov/10001691), and the International Sequencing Consortium (http://www.intlgenome.org/).

Many of the currently available fungal genomes were sequenced by large research groups such as the Broad Institute of MIT and Harvard, the Department of Energy Joint Genome Institute, and the John Craig Venter Institute. However, given the falling cost of sequencing, and the development of open source tools for *de novo* sequence assembly, increasingly more assembled genomes may be expected from small research labs. A flowchart of the steps involved in sequencing and assembling a small eukaryotic genome can be found in Haridas et al. (2011).
1.6 Hypotheses

Using the available genomes from the Pezizomycotina, this study seeks to develop a comparative genomic method based on BLAST and sequence divergence to identify phylogenetically informative genes, to determine if multigene trees are better than single-gene trees for resolving phylogenetic trees, and to find out if the identified genes can be used in genomes that have not been sequenced. Thus, the following hypotheses were examined in this study:

1) BLAST statistics such as e-values, bit scores, and identity can be used to represent sequence divergence, and subsequently identify genes that are phylogenetically informative for the filamentous ascomycetes at the class level.

2) A plurality of individual genes randomly selected from the filamentous ascomycete genomes will yield a single phylogenetic tree showing evolutionary relationships that match the currently accepted tree of the filamentous ascomycetes.

3) Superalignment trees of the filamentous ascomycetes are more consistent than single-gene trees at yielding the same tree topology when at least five genes are used.

In Chapter 2, comparative genomic methods to explore phylogenetic informative genes were examined. The following objectives were addressed in this chapter: (i) explore different BLAST statistics to identify a measure of phylogenetic information; and (ii) determine if individual genes can reveal well-resolved trees of the filamentous ascomycetes with strong branch support.

In Chapter 3, superalignments of the filamentous ascomycetes were constructed from sets of homologous genes to explore the effectiveness of superalignment trees compared to single gene trees. In addition, the role of 18S rDNA as an accurate representation of the true species
tree was addressed. For this Chapter, the following objectives were addressed: (i) construct superalignment trees based on increasing numbers of genes; (ii) compare the topologies of the superalignment trees to the single-gene trees; and (iii) determine the number of genes required for strongly-supported and well-resolved superalignment trees.

In Chapter 4, primers were developed from examples of genes showing particular mean identity values to examine their phylogenetic potential. The following objectives were addressed: (i) select a random sample of genes that contain the optimal amount of phylogenetic information for the filamentous ascomycetes; and (ii) determine if primers can be designed and used on unsequenced filamentous ascomycete genomes.
Table 1.1. Number of sequences found on the GenBank non-redundant database (NR) for each of the specified genomic regions in Fungi and in Ascomycetes. Search was performed in April 2012.

<table>
<thead>
<tr>
<th>Gene</th>
<th>Fungal Sequences</th>
<th>Ascomycete Sequences</th>
</tr>
</thead>
<tbody>
<tr>
<td>18S rDNA</td>
<td>228,334</td>
<td>91,588</td>
</tr>
<tr>
<td>RNA polymerase II largest subunit</td>
<td>6,838</td>
<td>4,782</td>
</tr>
<tr>
<td>RNA polymerase II second largest subunit</td>
<td>10,911</td>
<td>7,481</td>
</tr>
<tr>
<td>Translation elongation factor 1-alpha</td>
<td>16,940</td>
<td>14,604</td>
</tr>
<tr>
<td>Beta-tubulin</td>
<td>26,534</td>
<td>24,124</td>
</tr>
</tbody>
</table>
Table 1.2. Number of studies published citing the different tree-building algorithms within the last 20 years as listed in the Web of Science (http://apps.webofknowledge.com/). Some studies may have used multiple algorithms, thus appearing in more than one query.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Bayesian</td>
<td>135</td>
<td>14,139</td>
</tr>
<tr>
<td>Maximum Likelihood</td>
<td>1,728</td>
<td>13,061</td>
</tr>
<tr>
<td>Maximum Parsimony</td>
<td>1,655</td>
<td>9,100</td>
</tr>
<tr>
<td>Neighbor Joining</td>
<td>1,154</td>
<td>2,105</td>
</tr>
<tr>
<td>UPGMA</td>
<td>305</td>
<td>569</td>
</tr>
<tr>
<td>Minimum Evolution</td>
<td>116</td>
<td>183</td>
</tr>
<tr>
<td>TOTAL</td>
<td>5,093</td>
<td>39,157</td>
</tr>
</tbody>
</table>
Table 1.3. Commonly used nucleotide substitution models, their relative base frequencies and rates of substitutions. Frequencies of nucleotides in the models are represented by F. (Page and Holmes 1998)

<table>
<thead>
<tr>
<th>Substitution Model</th>
<th>Base Frequencies</th>
<th>Substitution Rates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jukes-Cantor (JC)</td>
<td>( F_A = F_C = F_G = F_T )</td>
<td>Transversions (Tv) = Transitions (Tn)</td>
</tr>
<tr>
<td>Kimura 2 Parameter (K2P)</td>
<td>( F_A = F_C = F_G = F_T )</td>
<td>Tv ( \neq ) Tn</td>
</tr>
<tr>
<td>Felsenstein (F81)</td>
<td>( F_A \neq F_C \neq F_G \neq F_T )</td>
<td>Tv = Tn</td>
</tr>
<tr>
<td>Hasegawa et al (HKY85)</td>
<td>( F_A \neq F_C \neq F_G \neq F_T )</td>
<td>Tv ( \neq ) Tn</td>
</tr>
<tr>
<td>General Time-Reversible (GTR)</td>
<td>( F_A \neq F_C \neq F_G \neq F_T )</td>
<td>Unequal substitution rates throughout</td>
</tr>
</tbody>
</table>
Table 1.4. Different versions of the BLAST algorithm have been developed for different query and database types. Translations from nucleotides into amino acid sequences are performed in all six frames. (NCBI 2010)

<table>
<thead>
<tr>
<th>Program</th>
<th>Query</th>
<th>Database</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLASTn</td>
<td>Nucleotide</td>
<td>Nucleotide</td>
<td>Nucleotide against nucleotide</td>
</tr>
<tr>
<td>BLASTp</td>
<td>Protein</td>
<td>Protein</td>
<td>Protein against protein</td>
</tr>
<tr>
<td>BLASTx</td>
<td>Nucleotide</td>
<td>Protein</td>
<td>Translated nucleotide against protein</td>
</tr>
<tr>
<td>tBLASTn</td>
<td>Protein</td>
<td>Nucleotide</td>
<td>Protein against translated nucleotide</td>
</tr>
<tr>
<td>tBLASTx</td>
<td>Nucleotide</td>
<td>Nucleotide</td>
<td>Translated nucleotide against translated nucleotide</td>
</tr>
<tr>
<td>PSI-BLAST</td>
<td>Protein</td>
<td>Protein</td>
<td>Identifies protein motif and searches through database</td>
</tr>
<tr>
<td>PHI-BLAST</td>
<td>Protein</td>
<td>Protein</td>
<td>Searches for user-defined protein motif</td>
</tr>
</tbody>
</table>
Figure 1.1. The eukaryotic ribosomal DNA (rDNA) gene cassette showing the relative positions of 18S small subunit (SSU), 5.8S, 28S large subunit (LSU), and the internal transcribed spacers (ITS1 and ITS2). Figure was adapted from Hillis and Dixon (1991).
Chapter 2: Development of a Comparative Fungal Genomics Method to Identify Phylogenetically Informative Genes

2.1 Introduction

2.1.1 Background

The study of evolutionary relationships has advanced from a relatively simple categorization of shared derived traits to a variety of complex methods such as those involving probabilistic estimations (Delsuc et al. 2005; Page and Holmes 1998; Rannala and Yang 2008). More recently developed phylogenetic approaches consider many factors, and try to minimize potential sources of errors. Factors that are considered controversial and subjected to criticism include the choice of gene and data type for the phylogenetic study and the method used to generate a dendrogram from the data (Rannala and Yang 2008). A difficulty often encountered in determining the best method for phylogenetics is the inability to confirm the true species tree, which lies in the evolutionary past.

There is currently an abundance of whole genome sequences available for study, with an exponential increase with each passing year (Benson et al. 2008; Hammesfahr et al. 2011). The rapidly growing amount of sequenced genome data can be attributed to the gradual replacement of traditional Sanger sequencing by the more recently developed next generation sequencers (Hammesfahr et al. 2011). Next generation sequencers, such as the Illumina-Solexa and Roche-454, can sequence an entire genome with considerably less investment in time and money, but with greater genome coverage than Sanger sequencing (Miller et al. 2010), which partially compensates for the much shorter read sequence lengths (Haridas et al. 2011). Furthermore, there is a massive output of data from single sequencing runs, such as the flow cell in Illumina GAIIx
machine which can produce over 50 Gb of data in eight lanes, or the newer HiSeq 2000 which can produce over 200 Gb of data in 16 lanes in a single run (Karow 2010). Multiple small genomes can be sequenced in parallel within a single run by multiplexing or subdividing single lanes (Haridas et al. 2011). Recent attempts at analyzing large amounts of sequencing data have yielded new insights into evolutionary relationships (Delsuc et al. 2005). However, despite the phylogenetic potential in the abundance of data, better methods may need to be developed to provide a more reliable and accurate estimation of the true species tree (Delsuc et al. 2005; Rannala and Yang 2008).

2.1.2 Data Selection

In the past, gene selection for use in phylogenetics was often based on practicality and its frequency of use (Aguileta et al. 2008). For example, ribosomal RNA genes (or rDNA) were popular in phylogenetics because there are multiple copies in the genome, and they experience concerted evolution, which keeps the copies consistent, and hence the multiple copies of the same genes allows them to be more easily amplified from even poor quality DNA (Hillis and Dixon 1991). As a demonstration of the popularity of ribosomal genes, a search for fungal sequences (Table 1.1) on Genbank (http://www.ncbi.nlm.nih.gov/genbank/) found 228,334 sequences for 18S rDNA, which was significantly higher than the 10,911 for RNA polymerase II second largest subunit (RPB2) sequences, 16,942 translation elongation factor 1-alpha, and 26,534 for beta-tubulin.

Although phylogenetic studies using ribosomal RNA genes were able to address questions about evolutionary history such as the relationship between Metazoa and Fungi (Roger and Hug 2006; Wainright et al. 1993), and the separation of Archaea and Bacteria (Hillis and Dixon 1991; Woese et al. 1990), little further research has been done to determine if these genes
reflect the true species tree. In fact, more recent phylogenetic studies using protein-coding genes revealed results that conflict with the trees derived from ribosomal gene sequences (Roger and Hug 2006). For example, Microsporidia was placed basal to most other Eukaryotes according to ribosomal genes (Keeling and McFadden 1998). However, protein-coding genes placed Microsporidia as a sister group to Fungi (Baldauf et al. 2000; Hirt et al. 1999; Keeling and McFadden 1998). In more recent studies with concatenated sequence data, Microsporidia is now considered to be a basal taxon in the fungal kingdom, but its exact relationship with the other fungal taxa are still unclear (Corradi and Keeling 2009).

Recent in-depth studies have been conducted to establish more rigorous criteria when selecting genes for phylogenetics. For example, different methods have been developed to attempt to identify true orthologs using Basic Local Alignment Search Tool (BLAST) (Altschul et al. 1990) results from genome sequence data. This theoretically has reduced the problem of paralogs confounding the phylogenetic analyses (Aguileta et al. 2008; Dujon et al. 2004; Moreno-Hagelsieb and Latimer 2008). In addition, statistical methods for estimating the amount of phylogenetic information contained within a gene sequence have also been developed (Goldman 1998; Townsend 2007).

Rokas et al. (2002) found a relationship between the amount of sequence divergence in a gene and its ability to separate taxa into their respective groups at specific taxonomic levels (Rokas et al. 2002). This makes intuitive sense as highly conserved genes lack sufficient information to separate different species, whereas poorly conserved genes may be too variable for phylum or class level relationships. Conserved sites that do not change across target lineages are useful for aligning the sequences, but offer little information regarding the evolutionary relationships between the organisms. Variable sites can have a phylogenetic signal but depending
on the tree-building algorithm, they may or may not be useful (Page and Holmes 1998). For example, in the parsimony method, at least two taxa must share the same trait in order for it to be considered phylogenetically informative and thereby phylogenetically useful, whereas in distance methods, all variations are used to build the tree (Page and Holmes 1998). It might be possible to develop a rapid method based on the levels of sequence divergence to identify phylogenetically useful genes.

2.1.3 Choice of Phylogenetic Method

In phylogenetics, there are many methods to generate a tree from the dataset. Moreover, there are different ways to manipulate the data to avoid problems such as composition bias, long-branch attraction (LBA), or account for multiple mutations at the same site (Felsenstein 1978; Gatesy et al. 2007; Hedtke et al. 2006; Page and Holmes 1998). Given the number of methods available and their advantages and disadvantages, disagreements may occur over the best method for data analyses, especially in cases where different analyses of the same dataset yield inconsistent results and conclusions (e.g. Rokas et al. 2003).

With the advent of genome sequencing, another potential area of contention has arisen. Given the growing amount of sequence data available, it has become relatively easy to select genes of interest for phylogenetic purposes. However, it is not certain how many genes need to be used. Computational constraints impose an upper limit on the size of the dataset, but selection of the optimal number of genes for phylogenetic analyses needs further study.

Rokas et al. (2003) used genome data from eight yeast species, and found that multigene phylogenies outperformed single-gene phylogenies in generating tree topologies with bootstrap support values greater than 95%. Analyses with single genes often result in several different, but equally optimal, phylogenies, and therefore may lead to potentially inconsistent evolutionary
relationships. Using different methods, Rokas et al. (2003) showed that their concatenated gene sets consistently gave the same tree, and they were able to reject all alternative topologies suggested by the single-gene trees using Templeton’s test (p<0.001). Moreover, based on their 106-gene tree of eight yeast species, Rokas et al. (2003) concluded that at least 20 genes might be necessary for constructing a well-supported multigene phylogeny.

Although their approach was novel and provided insight into the number of genes needed for consistent phylogenies, their methodologies and subsequent conclusions have faced criticisms. Hedtke et al. (2005) pointed out that high bootstrap values do not necessarily mean that the tree is correct. In addition, because Rokas et al. (2003) were examining many phylogenies from a large number of genes, they were more likely to be affected by method inconsistency, which is the failure of different methods to arrive at the same or correct tree despite the dataset becoming infinitely large (Felsenstein 1978; Hedtke et al. 2006).

One source of inconsistency in the dataset from Rokas et al. (2003) was the presence of LBA as a result of extremely long branches at the basal nodes and the outgroup (Gatesy et al. 2007; Hedtke et al. 2006). When the three basal taxa (out of eight) were removed from all 106 single-gene trees used in Rokas et al. (2003), the topologies of the resulting trees containing the remaining five taxa were identical (Gatesy et al. 2007). Moreover, Phillips et al. (2004) showed that using a different method to construct a tree based on the Rokas et al. (2003) dataset led to an alternate tree with 100% bootstrap support at all nodes. Phillips et al. (2004) identified a compositional bias that led to the alternate tree and concluded that although sampling error was greatly reduced using the method outlined by Rokas et al. (2003), systemic errors may still have been present and should be filtered out.
When attempting a multigene phylogenetic approach, recommendations have been suggested based on the results from the Rokas et al. (2003) study. For example, increased taxa sampling could have reduced the effects of LBAs (Hedtke et al. 2006). In avoiding LBAs, fewer genes may be needed to generate a well-supported tree (Gatesy et al. 2007; Hedtke et al. 2006). At that time, Rokas et al. (2003) were limited to the eight sequenced fungal genomes, but now, many more genomes have been sequenced and are publicly available. By having a larger dataset of diverse taxa with more representatives within each of the major clades or groupings, the effects of LBAs can be mitigated as OTUs (operational taxonomic units) with long branches can be broken up with sister taxa (Felsenstein 1978).

The studies showed that combining many genes may lead to well-resolved trees with strong bootstrap support, but if systematic errors are properly accounted for, fewer genes may be required. In addition, as previously stated, genes with specific levels of sequence divergence may be useful for specific taxonomic levels. When combined, a possible interpretation of these studies is that although a concatenated gene set can lead to a well-resolved tree, there is the possibility that some genes alone may be able to yield a well-resolved single-gene tree at specific taxonomic levels with strong bootstrap support. By carefully taking into consideration the evolutionary history of the genes, and sampling a sufficient number of taxa, it may be possible to identify individual genes that are reflective of the true species tree and are potentially useful for phylogenetics.

2.1.4 Objectives

The primary objective of this study was to develop a comparative fungal genomics method using BLAST and its statistics to assess the possibility of predicting, based on sequence divergence, whether a single gene with homologs present in 49 filamentous ascomycetes might
be useful for inferring a well-resolved tree with strong branch support. In this study, a subsample of available genes forming almost 200 single-gene trees were categorized into topological groups based on how many of the four fungal classes (Eurotiomycetes, Leotiomycetes, Dothideomycetes, and Sordariomycetes) were represented and resolved. This allowed testing of whether the most common topology was concordant with the currently accepted tree depicting the filamentous ascomycetes.

Hypothesis: Sequence divergence estimated using BLAST statistics, can be used to identify genes that are phylogenetically informative for the filamentous ascomycetes at the class level.

Hypothesis: Individual genes selected from the filamentous ascomycete genomes can be used to infer trees showing evolutionary relationships that match the currently accepted tree of the filamentous ascomycetes.

2.2 Methods

2.2.1 Genome Selection and BLAST Databases

The predicted protein-coding nucleotide sequences and amino acid sequences of 49 sequenced and annotated filamentous ascomycetes (Table 2.1) were downloaded in FASTA format. These files were copied to the Requin computing cluster, a part of the Shared Hierarchical Academic Research Computing Network (Sharcnet, Ontario) accessible at www.sharcnet.ca. The message passing interface (MPI) version of BLAST was used to take advantage of the computing resources available on Sharcnet, especially multiple cores. Aside from being able to use multiple cores, MPI-BLAST is functionally identical to StandAlone BLAST from NCBI (National Center for Biotechnology Information, http://www.ncbi.nlm.nih.gov/). Using the mpiformatdb.sh script that comes with mpiBLAST v1.5.0 (Darling et al. 2003), which is equivalent to the more common formatdb in StandAlone
BLAST, each of the 49 genomes was converted into a multiple-piece mpiBLAST database, to match the number of cores available. Example of scripts used to generate BLAST databases from genomic sequence data using mpiformatdb.sh in MPI-BLAST or formatdb in StandAlone BLAST are shown in Appendix 2.1.

2.2.2 Comparison of *Neurospora crassa* against 48 Genomes

*Neurospora crassa* was selected as the reference genome (version 4 from Ensemble, 21 April 2010). Each of its 9,839 annotated protein sequences was individually compared against the DNA sequences of the 48 other filamentous ascomycete genomes (Table 2.1) using tBLASTn (default options) in mpiBLAST with an e-value cutoff of 1E-5. An example blast job is shown in Appendix 2.2. Simmons et al. (2002) and Townsend et al. (2008) showed that although comparisons may be more objective at the amino acid level and true matches might be more easily recognized, there may be more phylogenetic information at the nucleotide level. Thus, tBLASTn was used to compare the protein query sequence to the translated nucleotide sequence. Preliminary comparisons were made against the assembled genome sequences since it is possible that some proteins may have not been called during the gene prediction and annotation process (Hsiang and Baillie 2005), but comparisons showed that relatively few genes fit such criteria. As a result, final BLAST comparisons (tBLASTn) were performed against the coding nucleotide sequences instead of the genomic sequences as it allowed for rapid retrieval of the sequence data from the databases.

For a match between a query sequence and a subject sequence, BLAST will typically output raw alignment scores, bit scores, expectation values (e-value), and identity scores. The raw alignment score shows how well the query and subject sequence match given an identity matrix. Because raw scores may not be comparable if the identity matrices used are not the same,
it is then converted to the bit score, which takes into account the identity matrix used and is therefore, more comparable across BLAST jobs. The e-value is calculated from the bit score adjusted for the size of both the query sequence and the database to give a probabilistic value of how significant the match is. The closer the e-value is to zero, the greater the significance and the less likely a similar or better match can be found by random chance in a database of the same size. The identity score is the percent of identical residues in the matching regions between the query and subject sequence.

Since BLAST v2.2.13 will sometimes fragment the query sequence and show multiple HSPs separated by low-scoring regions and thus, there might be multiple BLAST scores between the same query / subject pair. To make comparisons across entire gene lengths, a script was used to stitch and tile the HSP-fragments together and account for any overlaps before returning a single identity value for a query / subject pair. This Perl script (Appendix 2.3) made use of the BioPerl::MapTiling module (Stajich et al. 2002). In this study, we used an identity value relative to the entire query length because it is a constant value when a single gene is compared against 48 different genomes. Using another Perl script (Appendix 2.4), each of the 48 parsed BLAST results were condensed into a single table. For each N. crassa gene, an arithmetic mean bit score, an arithmetic mean identity value, and a geometric mean e-value were calculated based on the 48 comparisons.

### 2.2.3 Initial Gene Selection for Single Gene Trees

The 9839 genes in the N. crassa genome were sorted according to their geometric mean e-value based on comparison to 48 other filamentous ascomycete genomes. Genes that were present in at least 40 out of the 49 genomes were retained for the analyses. Based on the sorted list of N. crassa genes, 165 were then systematically selected (every 36th gene) for phylogenetic
analyses. For each 165 selected *N. crassa* gene, the highest matching nucleotide sequences were retrieved from the other 48 genomes using Stand Alone BLAST fastacmd on the original database (Altschul et al. 2009). Multiple sequence alignments were performed using Muscle v3.6 (Edgar 2004), trimmed in BioEdit v7.0.5.3 (Hall 1999), and subsequently used to generate unrooted NJ trees in ClustalW v2.0 (Larkin et al. 2007) with 1000 bootstrap replicates. To confirm the tree results, a second set of NJ trees were generated using MEGA v5.05 (Tamura et al. 2011) from each of the sequence alignments with 1000 bootstrap replicates and with the maximum composite likelihood method.

2.2.4 Identifying the Reference Tree Based on the Most Common Topology

The topologies of the 165 unrooted NJ trees were evaluated at the fungal class and fungal order taxonomic levels (Table 2.1). The number of trees sharing the same topology at the class level (i.e. Eurotiomycetes, Dothideomycetes, Sordariomycetes, and Leotiomycetes) was tabulated, and named T1, T2, T3, etc in order of frequency. The topology shared by the largest number of trees (T1) was used as the reference for comparing the topologies of the remaining trees, with the assumption that the most common topology may be the most representative of the true species tree.

2.2.5 Gauging Resolution of Trees

A resolution value for each of the 165 single-gene trees based on 49 genomes was assigned by comparison to the reference topology T1 (Figure 2.1), the most commonly found topology. If all species were placed into the expected four fungal classes, the tree was assigned a resolution value of ‘1’. If the species were not placed into the expected four classes, the resolution scores were assigned based on their congruence with the major nodes of T1. For
example, in alternate topologies, for each fungal class that appeared with the same species as in T1, the tree was assigned a value of 1/4. If a taxon was placed into a class that was not in agreement with T1, both classes were considered disrupted and 2/4 was deducted from its resolution score. If a single species was placed outside of its expected class, but not within another class, it was assigned a score of 3/4 because the other three classes had the expected placement. If none of the taxa tested were placed into their expected classes, the tree was given a value of ‘0’ for no resolution. Ordinal level relationships were not considered because in three of the four classes, there were only two orders. In the Sordariomycetes, there were more than two orders, but most of them only had one representative genome.

2.2.6 Comparisons of BLAST Statistics

The mean BLAST output statistics, such as the bit scores, e-values, and identity, were examined to identify which values might better reflect the suitability of a gene for producing a well-resolved NJ tree with robust bootstrap support. Genes were initially sorted by each particular BLAST statistic to assess whether a correlative pattern with the resolution values derived above was apparent.

Based on their mean identity value, each of the 165 systematically selected *N. crassa* genes tested for resolution was placed into a histogram with twenty equally divided mean identity bins between 0-100% (0-5%, 5-10%...95-100%). Resolution scores assigned to the genes ranged from 0 to 4 depending on the similarity of their inferred NJ trees to the reference topology at the class level. On the histogram, resolution scores were represented as different shadings.
2.2.7 Mean Identity Recalculation with Eight Query Genomes

Because of possible deficiencies in the original method for mean identity calculations which was based on the comparison of a single genome (N. crassa) to 48 other genomes, mean identities were recalculated by selecting two representative genomes from each of four fungal classes for comparison against the 48 other genomes. We selected Aspergillus nidulans and Coccidioides immitis from the Eurotiomycetes; Botrytis cinerea and Sclerotinia sclerotiorum from the Leotiomycetes; Pyrenophora tritici-repentis and Mycosphaerella graminicola from the Dothideomycetes; and N. crassa and Fusarium graminearum from the Sordariomycetes. The amino acid sequences of these eight genomes were compared using tBLASTn against the nucleotide sequences of the other 48 genomes separately, and parsed as described above. For each of the eight genomes, this yielded a database of each gene with its identity values for all other 48 compared genomes. A mean identity value was then calculated for each gene in each of the eight genomes.

To assess whether the distribution of the mean identities might change with each additional query genome, a script (Appendix 2.5) was used to help match each gene and its calculated mean identity value from the other seven query genomes to the homologous N. crassa gene, so that all homologs across the genomes could be represented by a single common gene identifier (e.g. NC00001). This resulted in a single table with the N. crassa gene identifier, the identifier for the homologous genes in the other seven query genomes if present, and the accompanying calculated mean identity value for each gene identifier. Only N. crassa genes with matching homologs in all 48 of the other genomes were retained resulting in 6,075 out of the original 9,839 genes.
An overall mean identity value was calculated for each *N. crassa* gene identifier using two query genomes, then three, four, and up to eight, yielding up to seven overall mean identity values for each gene. When combined with the original *N. crassa* single genome mean identity value, there were up to eight values for each gene. To determine if the number of query genomes affected the mean identity, the distribution of the identity values were examined by generating a histogram for each of the eight sets of values. Based on their overall mean identity value, each *N. crassa* gene was placed into a histogram with 25 equally divided mean identity bins between 0-100% (0-4%, 4-8%...96-100%). Rather than visualizing the eight histograms separately, the values used to generate the eight histograms were combined into a single table to produce a multiplot line graph using Excel 2007 (Figure 2.7) as this was more suitable for visual comparisons.

The 165 genes that were initially selected by mean e-value in section 2.1.3 were then re-assigned to mean identity bins based on the overall mean identity as calculated by the comparisons of 8 query genomes against all other genomes rather than just the single *N. crassa* comparisons. Another histogram was then constructed following methods in section 2.2.6 but using the 8-query mean identity instead of the single-genome mean identity.

### 2.2.8 Comparison of Commonly-Used Genes to the Reference Topology

Single-gene trees were generated from four genes commonly used in fungal phylogenetics to determine if they match the reference tree topology. Translation elongation factor one alpha (EF1-α), RNA polymerase II second largest subunit (RPB2), beta tubulin (β-tub), and 18S rDNA were selected for additional phylogenetic analyses. For RPB2, EF1-α, and β-tub, the sequence retrieval and NJ tree generation were performed as above from the 49 genomes. Because repetitive sequences are often difficult to assemble, 18S rDNA sequences
(which are part of the multicopy ribosomal cassette (Hillis and Dixon 1991)), were not found in the 49 filamentous ascomycete genomes. Thus, 18S rDNA sequences were obtained for 45 different filamentous ascomycetes from Genbank (Table 2.2; not all the same species as the 49, but with similar representation by Ascomycota class: 21 Eurotiomycetes, 16 Sordariomycetes, 5 Dothideomycetes, and 3 Leotiomycetes) and used to generate a NJ tree in ClustalW and a maximum likelihood (ML) tree in Phylip 3.2 (Felsenstein 1989), both with 1000 bootstrap replicates.

2.3 Results and Discussion

A flowchart outlining the steps to the method used in this study is depicted in Figure 2.3.

2.3.1 Selection of Genomes

Among the eukaryotes, the most commonly sequenced genomes are from the fungal kingdom, specifically the fungal division Ascomycota (Hammesfahr et al. 2011). Not all available Ascomycete genomes were used in this study, especially the abundant yeast genomes. Although previous studies have been criticized for their small taxa sampling size as that may understandably lead to possibly incorrect but strongly supported inferences (Hedtke et al. 2005), we chose to avoid yeast genomes as their reduced genome sizes (~5000 genes) might reduce the number of available genes in the common pool. For example Hsiang & Baillie (2005) found that the common core between the Baker's yeast *Saccharomyces cerevisiae* and at least 12 out of 14 other fungal genomes contained only 3340 genes whereas we found here that the common pool within at least 40 of 49 filamentous ascomycetes tested was 6,075 genes. Another potential problem is that yeasts are a more ancient and diverse lineage that diverged from the filamentous ascomycetes around a billion years ago (Blair Hedges and Kumar 2003; Heckman et al. 2001).
Many confounding differences may have arisen since then. Hence, we examined the sequences of 49 filamentous ascomycetes which had a last common ancestor 670 million years ago (Blair Hedges and Kumar 2003).

The 49 filamentous ascomycete genomes used in this study were the annotated ones available before 10 Jan. 2011. These were classified in four classes (Dothideomycetes, Eurotiomycetes, Leotiomycetes, and Sordariomycetes) and eleven orders (Table 2.1). Of the four classes, Leotiomycetes contained the fewest number of genomes and was represented by three genomes with two in the Helotiales and the remaining in the Erysiphales. Few Leotiomycetes have actually been sequenced and annotated (three genomes as of 10 Jan. 2011). This may at least in part be because some Leotiomycetes plant pathogens are obligate biotrophs (e.g. powdery mildews and possibly Rhystisma spp.), which makes them difficult to culture on artificial media, hence making it difficult to obtain pure DNA of single genotypes (Spanu et al. 2010).

Out of the four filamentous ascomycete classes, the Eurotiomycetes contain the greatest number of sequenced genomes followed by the Sordariomycetes. During this study, we found 22 Eurotiomycetes and 18 Sordariomycetes sequenced genomes. The sequenced Eurotiomycetes genomes belong to two orders, Eurotiales and Onygenales. Many Eurotiales are commonly found in laboratory and household environments, and will grow readily on artificial media. These include several species from the medically and economically important genera, Penicillium and Aspergillus. Similarly, many of the sequenced genomes from the Onygenales are medically important dermatophytes that can cause opportunistic infections in humans and animals.

The 18 sequenced Sordariomycete genomes were classified in six orders: Diaporthales, Hypocreales, Magn leporthales, Ophiostomatales, Phyllachorales, and Sordariales. At the start of
the study, there were only single representatives available from Diaporthales, Magnaporthales, and Ophiostomatales while Phyllachorales had two. Hypocreales and Sordariales had more sequenced genomes than the other orders with seven and six sequenced genomes respectively. Similar to the Eurotiomycetes, many Sordariomycetes will grow readily on prepared media. Furthermore, a large number of the sequenced Sordariomycetes are also economically important plant pathogens.

The remaining filamentous ascomycete class which has the second least abundant sequenced genomes found during this study was Dothideomycetes. The six Dothideomycetes genomes used were found in two orders, with two from Capnodiales, and four from Pleosporales. All six genomes are economically important plant pathogens and can also be readily grown on artificial media. However, fewer Dothideomycetes have been sequenced compared to both the Eurotiomycetes and the Sordariomycetes, perhaps because there are more medically important pathogens in the latter two groups.

Genome sequencing targets in the filamentous ascomycetes seem to have been disproportionately focused on the Sordariomycetes and the Eurotiomycetes rather than the Leotiomycetes and the Dothideomycetes. As outlined by the Fungal Genome Initiative in their white paper published in 2003 and 2004, the focus of fungal genome sequencing was placed primarily on human pathogenic fungi and their close relatives, and then secondarily on plant pathogens. As sequencing technologies improved and became less expensive over time, more fungal genomes were sequenced because of their small size among eukaryotes (Hsiang and Baillie 2004), and for general biological curiosity rather than economic importance. Cuomo et al. (2010) stated that in the past, genome sequencing efforts were by large specialized centers, but future individual investigators will be encouraged to sequence their fungal genomes of interest.
2.3.2 Parsing BLAST Results

Each of the 9,839 *N. crassa* genes was compared to those of the 48 other genomes separately using tBLASTn. An example of a BLAST output is shown in Appendix 2.2. For mean e-values, calculations were performed on a geometric scale instead of an arithmetic scale, while the arithmetic mean was used for bit scores and identities. An example of the mean identity calculation process is shown in Figure 2.2 in which the top match of a query sequence with a length of 568 amino acids yielded three HSPs with identity values of 84%, 72%, and 58%. Following the MapTiling stitching and recalculation process, the recalculated mean identity was 60.7%.

2.3.3 Single-Gene Tree Generation, Tree Topologies, and Resolution Evaluation

The first step in this study was to identify a reference tree topology against which the rest of the trees could be compared. Because the true species tree is not known (and might never be known), the criteria for selecting a reference tree for comparison purposes was subjective. In this study, we decided against generating a tree from a concatenation of all the selected genes because there may be stochastic and systematic errors (Phillips et al. 2004). Instead, we began by generating single-gene trees derived from comparisons among 49 genomes, and then examining them for common topological motifs.

To select the genes for the phylogenetic analyses, the list of the 9,839 *N. crassa* genes was sorted by geometric mean e-value. Then the genes that did not have a match in at least 40 genomes were filtered out leaving 6,075 genes. Next, 165 *N. crassa* genes were systematically selected from the list of 6,075 genes. The sequences of the selected 165 genes and their homologs from the 48 other genomes were retrieved from the FASTA files and aligned using Muscle. Ends and gaps of the alignments were manually trimmed in Bioedit. Subsequently, NJ
trees for each of the 165 alignments were generated using ClustalX with 1000 bootstrap replicates.

Out of the 165 single-gene NJ trees generated using ClustalW, the largest concordant group at the fungal class level contained 67 genes and was thus used as the reference T1 for comparisons with the remaining alternate topologies (Table 2.3). The next two largest groups, T2 and T3, contained 18 and 15 genes, respectively (Table 2.3). From the remaining trees, 38 trees were completely unresolved, or were partially resolved with topologies that appeared only once out of the 165 single-gene trees. The leftover 27 trees had topologies that occurred at least twice but not as often as eight.

With T1, all species were resolved into the four fungal classes with two major nodes: Dothideomycetes with Eurotiomycetes, and Leotiomycetes with Sordariomycetes (Figure 2.4). For other single-gene trees, a resolution value was assigned based on comparison to the T1 topology (0 for all four orders not resolved to 1 for all four orders resolved).

Eighteen genes shared the second most common topology, in which the clade of the three Leotiomycetes was nested within the 22 Eurotiomycetes. For these 18 trees, the placement of Leotiomycetes was not consistent, since in some trees, it split the Onygenales and Eurotiales (Figure 2.5), while in others, the split was uneven and the Leotiomycetes were placed within the Eurotiales. In these trees, the Eurotiomycetes were rendered paraphyletic by the Leotiomycetes but because evaluations were performed at the class level, the slightly different topologies were treated as one general topology.

The third most common topology occurred in 15 genes, and it placed the Leotiomycetes as a sister to the Eurotiomycetes. Although all the species were placed into their proper class, the relationships between classes were not the same as T1. In T1, the Eurotiomycetes and
Dothideomycetes grouped together while the Leotiomycetes and the Sordariomycetes were grouped together. In contrast, T3 showed the Eurotiomycetes grouped with the Leotiomycetes while the Dothideomycetes grouped together with the Sordariomycetes.

In comparison, the NJ trees generated using MEGA yielded similar results but with different numbers. T1 was still the most frequently observed topology with 49 trees, which differed from the 65 found when using ClustalW. Moreover, there were 6 and 13 trees showing T2 and T3 respectively.

Ordinal level relationships were not considered in the resolution values because in three of the four classes, there were only two orders, which is not sufficient for evaluating relationships as there is only one possible way for them to group together. In the Sordariomycetes, there were more than two orders, but most of them only had one representative genome. In addition, more often than not, if a species was not placed into its correct taxonomic order, it was also not placed with other members of its class.

2.3.4 Selection of BLAST Statistics to Estimate Phylogenetic Information

From parsing the BLAST result for each gene to genome comparison, we obtained the bit scores, e-value, and identity relative to query length for all 9,839 N. crassa genes against the genes of 48 other fungal genomes. The means of each of the three BLAST assessment parameters were considered for estimating the potential of a gene for resolving relationships at specific taxonomic levels.

Throughout the course of the testing, several problems were encountered regarding the use of e-values. First, there is an artificial lower limit of $10^{-180}$ imposed by the BLAST program. Anything lower is assigned a value of ‘0’. This is intuitive because e-values represent the probability of finding a better BLAST hit in the same database by chance alone. As the e-value
becomes infinitesimally small, the chance is essentially zero so the system assigns it a value of zero. The other problem we found with e-values is the programmed ‘bias’ against shorter query sequences. E-values are calculated from bit scores, which are refined versions of the raw alignment score. Given the same parameters, a longer gene with the same raw score, and bit score as a shorter gene will have a higher e-value. Thus, short and conserved genes may have greater percentage of conservation when compared to longer genes, but the conservation is not reflected as a more significant e-value. This is not a problem with the BLAST algorithm because it is valid to assume that a smaller query sequence will have a higher probability of being matched by chance than a much longer sequence. However, these particular BLAST statistics may not be suitable for estimating the amount of sequence divergence a gene has as it is possible for the lower values of the shorter genes to confound the results or produce false trends.

Another potential problem may arise when long query sequences with multiple domains are used. A single HSP may be found between the query and subject sequences over the entire length of the query despite the low levels of conservation between domains. Based on such HSP, BLAST may then output a large bit score and a significant e-value. However, since only a small portion of the gene is conserved between the two sequences, the overall level of identity may be very low despite the significant BLAST scores. Thus, bit scores and e-values may not be fully representative of the level of sequence divergence between a query and subject sequence. Moreover, the correlation scores between resolution ability of genes and mean bit scores, mean e-values, and mean identity were initially evaluated in this study, but because several underlying assumptions of correlation analyses (e.g. data were nonlinear and not normally distributed) were violated, correlation analyses was discontinued.
Out of the three BLAST statistics we compared, we found that mean nucleotide identity may be more reliable than e-values or bit scores in representing the level of sequence divergence and estimating the potential of a gene for producing a well-resolved tree as compared to the reference T1 at the class level. To better visualize the data, a histogram was generated using the mean identity values and the resolution scores for the filamentous ascomycetes. Each of the selected 165 genes was placed into a histogram with 20 equally divided mean identity bins between 0-99% (0-5%, 5-9%...95-99%) based on their mean identity as calculated by comparing the *N. crassa* sequence against the other 48 genomes. The five different possible resolution states assigned to the 165 genes were depicted on the histogram as different shadings (Figure 2.6).

As seen in Figure 2.6, genes with low mean identity tended not to resolve the filamentous ascomycetes at the class level. As the mean identity increased to 60%, genes were more likely to produce a well-resolved tree. However, as the mean identity of the gene approached 100%, fewer genes were fully resolving the trees at the class level. Our results suggest that it may be possible to predict if a gene is able to resolve a tree with robust bootstrap values at the class level but more representative taxa at this level need to be studied. In addition, these results corroborated the findings of Rokas et al. (2002) in which the amount of sequence divergence is related to tree resolution at different taxonomic levels.

In this study, we used mean sequence identity relative to the query as the measure of sequence divergence. As sequence divergence decreases, genes have more conserved sites and are less likely to accumulate multiple mutations at the same site. This may lead to an improvement in tree resolution. However, as sequence divergence decreases further, the amount of shared variations between taxa decreases. Without variations, genes will not be able to yield well-resolved trees and this corresponded with the decrease in tree resolution that was observed
(Figure 2.6). If resolution of orders were considered instead of classes, the histogram could be expected to shift leftwards because an increased amount of variation may be needed for the taxonomic levels lower than class, such as orders, families and genera.

2.3.5 Comparison of Mean Identities Calculated with Multiple Genomes

Throughout phylogenetic analyses, several genes appeared to be fairly conserved at the nucleotide level but their calculated mean identity values were lower than what was expected for the amount of sequence similarity. We examined the sequence alignments and found that for some genes, the *N. crassa* sequence matched well with the other *Neurospora* sequences (*N. discreta* and *N. tetrasperma*), but poorly with the 46 sequences from the other genomes, despite the other 46 showing high similarity to each other. This suggests that a single query genome may not be sufficient for calculating an accurate mean identity value for one gene across all 49 genomes.

To calculate an identity value for a gene with a higher degree of accuracy, the mean identity should be calculated from pairwise comparisons of each of the 49 genomes. However, this would have been computationally prohibitive and time consuming because there were close to 10,000 genes in each of the 49 genomes analyzed. In addition, the sequenced genomes were not equally distributed across classes and this might skew values toward over-represented clades. For example, out of the 49 genomes used in this study, 22 were from the Eurotiomycetes but only 3 were from Leotiomycetes. To avoid bias from over-represented classes, two genomes from each of the four classes were selected for the tests against the other 48 genomes. Thus, rather than comparing all 49 genomes against each other, mean identities were calculated using one to eight query genomes against the other 48 genomes to determine if the number of query genomes affected the mean identity distribution.
The same method used to calculate the single-genome mean identity values for the *N. crassa* genes was applied to each of the other seven selected query genomes. Each gene from the query genome was compared against the genes in the 48 other genomes using tBLASTn. The BLAST jobs were then parsed to yield the gene identifiers and the identity values with matching genes in the other 48 genomes, which were used to calculate the mean identity for the gene. Using a script, each gene and its mean identity from the seven other query genomes were matched to the homologous *N. crassa* gene. Thus, all homologs across the eight query genomes could be referred to by a *N. crassa* gene identifier. This process resulted in a table with the *N. crassa* gene identifier, the identifier for the homologous genes in the other seven query genomes (if present) and up to eight mean identity values. *N. crassa* genes that were missing a matching homolog in more than eight other genomes were discarded. From the original 9,839 genes, only 6,075 genes remained.

Overall mean identity values were calculated for each *N. crassa* gene identifier using two query genomes, then three, and up to eight. Including the original *N. crassa* mean identity, there were up to eight overall mean identity values for each gene. To examine how overall mean identity of sets of homologous genes might be affected by the number of combined genomes, distributions of the genes according to each of the eight overall mean identity values were generated and examined using a multiplot line graph (Figure 2.7).

The addition of a second genome to the mean identity calculations caused the distribution to change from slightly bimodal to unimodal. With each additional genome up to four, the mode of the distribution stopped changing and the tails were drawn in towards the mode. Another genome from each class was selected and added to the calculations to see if the distribution would change further. Between four to eight genomes, the distribution appeared to have
stabilized as little change occurred with each additional genome. Thus, additional genomes beyond the eight were not expected to alter the distribution and no further additions were made. The original single genome *N. crassa* mean identity values were then replaced with the more consistent and possibly more accurate eight genome identities.

Because the mean identity of the 165 genes used in the phylogenetic analyses may have changed when they were replaced with the eight genome mean identity, the histogram in Figure 2.6 was regenerated using the new values. As before, each of the 165 genes was placed into a histogram with 20 equally divided mean identity bins between 0-99% (0-5%, 5-9%...95-99%) based on their eight genome mean identity values. The five different possible resolution states assigned to the 165 genes were depicted on the histogram as different shadings (Figure 2.8). The differences between the two histograms are discussed in section 2.3.7.

### 2.3.6 Potential Sources of Variations

In many of the alternative topologies that appeared, such as T2, T3 and the less common topologies, the major difference was the placement of the Leotiomycetes. Because there were only three sequenced genomes representing two different orders in the Leotiomycetes, it may have been more susceptible to systematic errors such as LBAs. Another potentially confounding problem is the highly reduced number of genes in some genomes. For example, the Leotiomycetes *B. graminis* has been estimated to have around 5,800 (Spanu et al. 2010), which contrasts with the average of 10,032 genes (Table 2.1) in the Eurotiomycetes genomes used in this study. This may have reduced the number of genes available for the phylogenetic comparisons. If more Leotiomycetes genomes were sequenced and available for this study, we suspect that more single-gene trees would share a topology with the T1 reference tree.
During a gene duplication event, taxa may obtain two copies of the same gene. Because the organism only requires a single functional copy of the gene, the other copy is free to mutate and often does (Page and Holmes 1998). However, in recent gene duplication events, the copy may not have mutated beyond recognition. As a result, the presence of these paralogous genes may also be detected by BLAST. Although the top hit found by BLAST was used for the downstream phylogenetic analyses, it is possible that the top hit in the genome database was not the ortholog but the paralog and therefore, not the correct homolog to use. This would have lead to trees with reduced resolution scores as a result of a few misplaced taxa. In this study, because 165 genes were systematically selected from the genome, a few of the genes may have paralogs in other genomes and indeed, some trees do show unusual relationships between more distantly related taxa.

2.3.7 Single-Gene Trees and the Reference Tree

In studies that examine the performance of genes that may be useful for phylogenetics, a reference tree is often generated or selected from previous research for comparison purposes. As previously mentioned, methods to generate a reference tree can be subjective because the true species tree cannot be determined with absolute certainty and the conclusions may change depending on the reference tree chosen. In this study, the tree topology that occurred most often among the single-gene NJ trees was selected as the reference. In contrast, several studies chose to recognize a superalignment tree generated using many genes as the most reflective of the true species tree and thus, as a reference tree (Aguileta et al. 2008; Rokas et al. 2003). Superalignments confer several advantages over conventional single gene trees, but their reliability may be heavily dependent on the gene selection method and genes used, since it has been shown that neglecting differences in the evolutionary histories of the genes selected can
lead to poorly-supported trees despite the large number of genes involved in the tree (Kuramae et al. 2007).

The approach we used offers a different perspective on what may be representative of the true tree. As shown in Table 2.3, the number of single-gene trees sharing the reference T1 was much higher than what would be expected by random chance alone. When T1 was compared to phylogenetic trees of the ascomycetes in the literature, it matched the topology of an unrooted version of a tree constructed from six genes which included the 18S rDNA small subunit (Schoch et al. 2009b). To confirm this observation, we retrieved 18S sequences from 46 filamentous ascomycetes and used the resultant alignment to generate a NJ tree and a maximum likelihood tree. We chose 18S because like other regions of the rDNA, it has been used extensively for phylogenetics, especially in fungal systematics. We compared the resulting unrooted topology of the 18S tree and found that it was congruent with T1 at the class level. This shows that a plurality of the 165 single gene NJ trees was concordant with both the currently accepted evolutionary relationship of the filamentous ascomycetes, and also an 18S-based tree.

The reference T1 topology was also compared against three other commonly used genes in phylogenetics, EF1-α, RPB2, and β-tub. The calculated mean identities were 83.7, 71.9, and 85.4 respectively (N. crassa gene identifiers: NC001234; NC001409; and NC003616), which placed them among the more highly conserved genes in N. crassa. β-tub was within the top 25 genes with the highest eight-genome mean identity out of all the genes in the N. crassa genome. Because these three genes are well-conserved, it may have been easier to design oligonucleotides to allow for PCR amplification across large taxonomic groups, which may explain their popularity. However, surprisingly, none of the three trees of the 49 filamentous ascomycetes exactly matched the T1 topology: EF1-α placed the three Leotiomycetes within the
Dothideomycetes, while RPB2 yielded the T3 topology and β-tub revealed a T2 topology but may have been suffering from LBAs as some branches were unusually long. Despite their inconsistency with 18S and T1, the three genes may have some utility when combined to generate multigene trees as some studies have done (Schoch et al. 2009a; Schoch et al. 2009b). One possibility is that the Leotiomycetes may be skewing the topologies because this class only has three sequenced representatives, and even on GenBank, full length or near full length coding sequences of these three genes are not available for a diverse sampling of this class. Complete genome sequencing of more Leotiomycetes is needed. Alternatively, it is also possible that these three genes may be too conserved and lack sufficient variation to resolve class level relationships.

The results from this study suggest that a single gene alone may be sufficient for producing a well-resolved tree without having to use multiple genes for superalignments. In addition, the results also support the conclusion that the sequence divergence of a gene is related to its ability to infer a well-resolved tree at a specific taxonomic level (Rokas et al. 2002). Thus, for single-gene trees, rather than choosing a random gene for phylogenetic analyses, a selection based on level of sequence divergence may be more likely to produce a well-resolved and accurate tree for a specific taxonomic level. In Figure 2.8, which shows the distribution of resolution with the eight genome mean identities instead of the original, single-genome mean identities, rising mean identity corresponded with an increasing chance for a gene to yield a well-resolved tree at the class level. The highest peak occurred at 45-50% mean identity suggesting that genes with mean identities that fall in this range are more likely to be phylogenetically useful at the class level. However, there were two other peaks, one at 30-35% and the other at
75-80%. Both these peaks contained single-gene trees that fully resolved, but not with the T1 topology.

In conclusion, out of the single-gene trees, 88/165 of genes tested yielded a resolved tree with all taxa in their proper class, while 67/88 or 67/165 (40%) genes had a tree topology that matched T1. This suggests that sampling a random gene from the Ascomycota genome that is also present in another 48 diverse Ascomycota genomes at the class level will only give about a 1/3 chance of producing a well resolved tree. In contrast, by targeting a gene or genes at specific levels of sequence divergence, the chance of obtaining a well-resolved tree may increase. For example, between 45-60% mean identity, 33/47 (70.2%) genes yielded a well-resolved tree while 30/47 (63.8%) yielded a well-resolved tree that also matched 18S and supports the currently accepted tree of the Ascomycetes. Genes at different levels of sequence divergence will have different chances of yielding a well-resolved tree that also matched 18S. For example, at 20-30% mean identity, 4/20 (20%) genes resolved but none matched 18S; at 80-90% mean identity, 7/22 gene resolved (31.8%) but only 4 matched 18S (4/22, 18.2%). Although our results corroborated the findings from Rokas et al. (2003) where they showed that some genes may give some conflicting tree topologies, we also found that a plurality of fungal genes alone can produce a well-resolved tree for the filamentous ascomycetes that also match the currently accepted tree of Ascomycetes classes.

In this study, the results showed that it is possible to infer well-resolved trees from individual genes. However, it has not been determined if single-gene trees are more consistent and reliable than multigene trees. Additional research will have to be conducted to compare the performance of single-gene trees to multigene trees. Moreover, because of the limited number of available sequenced and annotated genomes at the start of this study, resolution of relationships
below the class level could not be examined. Since the amount of sequenced data is growing, this type of study can be conducted in the future to identify individual genes that may be phylogenetically informative at lower taxonomic levels such as orders, genera and species.
Table 2.1. Sequenced genomes used in this study with taxonomic class and order, file size, number of genes, date downloaded, source, and version.

† Joint Genome Institute (JGI), Broad Institute (Broad), National Center for Biotechnology Information (NCBI), Database of the Genomes Analyzed at NITE (DOGAN), Blumeria Sequencing Project (BluGen)

<table>
<thead>
<tr>
<th>Species</th>
<th>Class</th>
<th>Order</th>
<th>File (MB)</th>
<th>Genes</th>
<th>Downloaded</th>
<th>Source†</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mycosphaerella fijiensis</td>
<td>Dothideomycetes</td>
<td>Capnodiales</td>
<td>113</td>
<td>81,415</td>
<td>10 May 2010</td>
<td>JGI</td>
<td>1</td>
</tr>
<tr>
<td>Mycosphaerella graminicola</td>
<td>Dothideomycetes</td>
<td>Capnodiales</td>
<td>16</td>
<td>11,395</td>
<td>20 Apr 2008</td>
<td>JGI</td>
<td>1</td>
</tr>
<tr>
<td>Alternaria brassicicola</td>
<td>Dothideomycetes</td>
<td>Pleosporales</td>
<td>15</td>
<td>10,688</td>
<td>20 Jan 2010</td>
<td>JGI</td>
<td>1</td>
</tr>
<tr>
<td>Cochliobolus heterostrophus</td>
<td>Dothideomycetes</td>
<td>Pleosporales</td>
<td>178</td>
<td>111,246</td>
<td>20 Jan 2010</td>
<td>JGI</td>
<td>1</td>
</tr>
<tr>
<td>Pyrenophora tritic-repentis</td>
<td>Dothideomycetes</td>
<td>Pleosporales</td>
<td>20</td>
<td>12,171</td>
<td>26 Jan 2010</td>
<td>Broad</td>
<td>3</td>
</tr>
<tr>
<td>Sclerotinia nodorum</td>
<td>Dothideomycetes</td>
<td>Pleosporales</td>
<td>20</td>
<td>15,983</td>
<td>20 Jan 2010</td>
<td>JGI</td>
<td>1</td>
</tr>
<tr>
<td>Penicillium crysogenum</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>19</td>
<td>12,744</td>
<td>01 Sep 2010</td>
<td>NCBI</td>
<td>1</td>
</tr>
<tr>
<td>Penicillium marneffei</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>20</td>
<td>10,630</td>
<td>01 Sep 2010</td>
<td>NCBI</td>
<td>1</td>
</tr>
<tr>
<td>Talaromyces stipitatis</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>23</td>
<td>13,261</td>
<td>31 Aug 2010</td>
<td>NCBI</td>
<td>1</td>
</tr>
<tr>
<td>Aspergillus clavatus</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>15</td>
<td>9,379</td>
<td>10 May 2010</td>
<td>Ensembl</td>
<td>4</td>
</tr>
<tr>
<td>Aspergillus fischeri</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>16</td>
<td>10,728</td>
<td>01 Sep 2010</td>
<td>Ensembl</td>
<td>4</td>
</tr>
<tr>
<td>Aspergillus flavus</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>19</td>
<td>13,487</td>
<td>10 May 2010</td>
<td>Ensembl</td>
<td>4</td>
</tr>
<tr>
<td>Aspergillus fumigatus</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>15</td>
<td>9,846</td>
<td>10 May 2010</td>
<td>Ensembl</td>
<td>4</td>
</tr>
<tr>
<td>Aspergillus nidulans</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>18</td>
<td>10,597</td>
<td>10 May 2010</td>
<td>Ensembl</td>
<td>4</td>
</tr>
<tr>
<td>Aspergillus oryzae</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>18</td>
<td>11,197</td>
<td>10 May 2010</td>
<td>JGI</td>
<td>3</td>
</tr>
<tr>
<td>Aspergillus terreus</td>
<td>Eurotiomycetes</td>
<td>Eurotiolales</td>
<td>17</td>
<td>12,074</td>
<td>02 Mar 2008</td>
<td>DOGAN</td>
<td>1</td>
</tr>
<tr>
<td>Ajellomyces capsulatus</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>17</td>
<td>10,406</td>
<td>02 Sep 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Arthrodema benhamiae</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>12</td>
<td>7,980</td>
<td>01 Sep 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Coccomyces immitis</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>14</td>
<td>9,910</td>
<td>10 May 2010</td>
<td>Broad</td>
<td>3</td>
</tr>
<tr>
<td>Coccomyces posadassii</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>14</td>
<td>10,228</td>
<td>10 May 2010</td>
<td>Broad</td>
<td>2</td>
</tr>
<tr>
<td>Microsporum canis</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>14</td>
<td>8,765</td>
<td>10 May 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Microsporum gyipseum</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>14</td>
<td>8,876</td>
<td>20 Jan 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Trichophyton equinum</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>15</td>
<td>8,560</td>
<td>26 Jan 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Trichophyton rubrum</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>16</td>
<td>8,638</td>
<td>26 Jan 2010</td>
<td>Broad</td>
<td>2</td>
</tr>
<tr>
<td>Trichophyton tonsurans</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>15</td>
<td>8,230</td>
<td>26 Jan 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Trichophyton verrucosum</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>12</td>
<td>8,024</td>
<td>10 May 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Uncinocarpus reessii</td>
<td>Eurotiomycetes</td>
<td>Ongyenales</td>
<td>14</td>
<td>7,798</td>
<td>20 Apr 2008</td>
<td>Broad</td>
<td>2</td>
</tr>
<tr>
<td>Blumeria graminis</td>
<td>Leotiomycetes</td>
<td>Erysiphales</td>
<td>8.8</td>
<td>5,822</td>
<td>10 Jan 2011</td>
<td>BluGen</td>
<td>1</td>
</tr>
<tr>
<td>Botrytis cinerea</td>
<td>Leotiomycetes</td>
<td>Helotiales</td>
<td>21</td>
<td>16,448</td>
<td>19 Jan 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Sclerotinia sclerotorum</td>
<td>Leotiomycetes</td>
<td>Helotiales</td>
<td>20</td>
<td>14,522</td>
<td>26 Jan 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Cyphotrichia parasitica</td>
<td>Sordariomycetes</td>
<td>Diaporthales</td>
<td>17</td>
<td>11,609</td>
<td>20 Jan 2010</td>
<td>JGI</td>
<td>2</td>
</tr>
<tr>
<td>Fusarium graminearum</td>
<td>Sordariomycetes</td>
<td>Hypocreales</td>
<td>21</td>
<td>11,640</td>
<td>06 Apr 2009</td>
<td>Broad</td>
<td>3</td>
</tr>
<tr>
<td>Fusarium oxysporum</td>
<td>Sordariomycetes</td>
<td>Hypocreales</td>
<td>28</td>
<td>17,735</td>
<td>25 May 2010</td>
<td>Broad</td>
<td>2</td>
</tr>
<tr>
<td>Fusarium verticillioides</td>
<td>Sordariomycetes</td>
<td>Hypocreales</td>
<td>24</td>
<td>14,179</td>
<td>25 May 2010</td>
<td>Broad</td>
<td>3</td>
</tr>
<tr>
<td>Trichoderma atroviride</td>
<td>Sordariomycetes</td>
<td>Hypocreales</td>
<td>18</td>
<td>11,100</td>
<td>20 Jan 2010</td>
<td>JGI</td>
<td>4</td>
</tr>
<tr>
<td>Trichoderma reesei</td>
<td>Sordariomycetes</td>
<td>Hypocreales</td>
<td>15</td>
<td>9,129</td>
<td>07 Nov 2008</td>
<td>JGI</td>
<td>2</td>
</tr>
<tr>
<td>Trichoderma virens</td>
<td>Sordariomycetes</td>
<td>Hypocreales</td>
<td>131</td>
<td>85,377</td>
<td>20 Jan 2010</td>
<td>JGI</td>
<td>1</td>
</tr>
<tr>
<td>Nectria haematococca</td>
<td>Sordariomycetes</td>
<td>Hypocreales</td>
<td>25</td>
<td>15,707</td>
<td>19 Jan 2010</td>
<td>JGI</td>
<td>2</td>
</tr>
<tr>
<td>Magnaporthea oryzae</td>
<td>Sordariomycetes</td>
<td>Magnaporthales</td>
<td>17</td>
<td>11,054</td>
<td>10 May 2010</td>
<td>Broad</td>
<td>6</td>
</tr>
<tr>
<td>Grossmannia clavigera</td>
<td>Sordariomycetes</td>
<td>Ophiostomalatales</td>
<td>15</td>
<td>8,513</td>
<td>13 May 2010</td>
<td>NCBI</td>
<td>1</td>
</tr>
<tr>
<td>Verticillium albo-atrum</td>
<td>Sordariomycetes</td>
<td>Phyllocherales</td>
<td>17</td>
<td>10,221</td>
<td>26 Jan 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Verticillium dahliae</td>
<td>Sordariomycetes</td>
<td>Phyllocherales</td>
<td>18</td>
<td>10,535</td>
<td>26 Jan 2010</td>
<td>NCBI</td>
<td>1</td>
</tr>
<tr>
<td>Chaetomium globosum</td>
<td>Sordariomycetes</td>
<td>Sordariales</td>
<td>17</td>
<td>11,124</td>
<td>20 Jan 2010</td>
<td>Broad</td>
<td>1</td>
</tr>
<tr>
<td>Neurospora crassa</td>
<td>Sordariomycetes</td>
<td>Sordariales</td>
<td>16</td>
<td>9,857</td>
<td>10 May 2010</td>
<td>Ensembl</td>
<td>4</td>
</tr>
<tr>
<td>Neurospora discreta</td>
<td>Sordariomycetes</td>
<td>Sordariales</td>
<td>17</td>
<td>9,948</td>
<td>20 Jan 2010</td>
<td>JGI</td>
<td>1</td>
</tr>
<tr>
<td>Neurospora tetrasperma</td>
<td>Sordariomycetes</td>
<td>Sordariales</td>
<td>17</td>
<td>10,640</td>
<td>20 Jan 2010</td>
<td>JGI</td>
<td>1</td>
</tr>
<tr>
<td>Podospora anserina</td>
<td>Sordariomycetes</td>
<td>Sordariales</td>
<td>17</td>
<td>10,594</td>
<td>20 Apr 2008</td>
<td>JGI</td>
<td>5</td>
</tr>
<tr>
<td>Sporotrichum thermophile</td>
<td>Sordariomycetes</td>
<td>Sordariales</td>
<td>15</td>
<td>8,806</td>
<td>20 Jan 2010</td>
<td>JGI</td>
<td>2</td>
</tr>
</tbody>
</table>
Table 2.2. Gene identifier, species, and class of the 45 fungal sequences used to generate the 18S ribosomal DNA tree. Sequences were obtained from Genbank (http://www.ncbi.nlm.nih.gov/genbank/).

<table>
<thead>
<tr>
<th>Gene Identifier</th>
<th>Species</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>gi</td>
<td>451833</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>28415895</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>284158965</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>37778551</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>158516813</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>1419549</td>
<td>emb</td>
</tr>
<tr>
<td>gi</td>
<td>156857863</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>12862449</td>
<td>emb</td>
</tr>
<tr>
<td>gi</td>
<td>709802</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>945411</td>
<td>dbj</td>
</tr>
<tr>
<td>gi</td>
<td>7194836</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>115279558</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>297375444</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>13661089</td>
<td>dbj</td>
</tr>
<tr>
<td>gi</td>
<td>267451935</td>
<td>ref</td>
</tr>
<tr>
<td>gi</td>
<td>19908241</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>16189885</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>1209739</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>3105</td>
<td>emb</td>
</tr>
<tr>
<td>gi</td>
<td>35844215</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>5186</td>
<td>emb</td>
</tr>
<tr>
<td>gi</td>
<td>19908243</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>19908247</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>450860</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>975348</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>2588678</td>
<td>dbj</td>
</tr>
<tr>
<td>gi</td>
<td>383213015</td>
<td>dbj</td>
</tr>
<tr>
<td>gi</td>
<td>6172368</td>
<td>dbj</td>
</tr>
<tr>
<td>gi</td>
<td>45775624</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>167304</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>22340357</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>56201266</td>
<td>dbj</td>
</tr>
<tr>
<td>gi</td>
<td>12611655</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>331029790</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>385137410</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>13661089</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>1209739</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>3080</td>
<td>emb</td>
</tr>
<tr>
<td>gi</td>
<td>27448350</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>347013775</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>371496520</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>20207282</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>156857864</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>9581850</td>
<td>gb</td>
</tr>
<tr>
<td>gi</td>
<td>6172364</td>
<td>dbj</td>
</tr>
</tbody>
</table>
Table 2.3. Number of genes out of 165 genes sharing the same Neighbor-Joining (NJ) tree topologies based on manual evaluations of trees constructed from alignments of 49 sequences using ClustalX 2.0 and MEGA 5 with 1,000 bootstrap replicates. Trees generated using MEGA used a maximum composite likelihood model. Multiple sequence alignments were performed using MUSCLE. Numbers of classes in the tree topologies are also indicated. The unique and unresolved category contained tree topologies that either showed a large polytomy or only occurred once out of all 165 genes examined. The remaining topologies that occurred more than once but not as often as topologies 1, 2, and 3 are in the ‘others’ category.

<table>
<thead>
<tr>
<th>NJ Tree Topology</th>
<th>Number of Genes</th>
<th>Number of Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology 1 (T1)</td>
<td>67 Clustal 49 MEGA</td>
<td>4</td>
</tr>
<tr>
<td>Topology 2 (T2)</td>
<td>18 Clustal 6 MEGA</td>
<td>4</td>
</tr>
<tr>
<td>Topology 3 (T3)</td>
<td>15 Clustal 18 MEGA</td>
<td>4</td>
</tr>
<tr>
<td>Unique and Unresolved</td>
<td>38 Clustal 71 MEGA</td>
<td>N/A</td>
</tr>
<tr>
<td>Others</td>
<td>27 Clustal 21 MEGA</td>
<td>N/A</td>
</tr>
<tr>
<td>Total</td>
<td>165 Clustal 165 MEGA</td>
<td></td>
</tr>
</tbody>
</table>
Figure 2.1. Examples of resolution score assignments based on 4 classes distributed amongst two nodes. If the vtree topology matches the reference topology (T1), then a value of ‘1’ is assigned as the resolution score. For every clade that is not resolved like in T1, a score of 1/4 is deducted. For example, if there are two disrupted clades because a species was placed into a clade that does not agree with T1 (D2 topology), then the tree is assigned a value of ‘2/4’. If a species was placed outside of all the clades including the one it was placed within in T1 (D1 topology), only a single clade is disrupted and the tree is assigned a value of ‘3/4’. If there is only one major node, and the tree shows a star topology (R0), the tree is assigned a value of ‘0’.
Figure 2.2. An example of the mean identity recalculations based on the example BLAST result found in Appendix 2.2. A query sequence 568 residues in length was compared using tBLASTn against the predicted gene database of the genome, and in the top hit returned, there were three HSPs in the subject sequence. The HSPs returned three identity values: 84%, 72%, and 58%. Because the HSPs are not reflective of the overall identity between the query and subject and cannot be compared between BLAST jobs as the lengths of the HSPs may not be constant across comparisons, a Perl script using the MapTiling module was used to stitch the HSPs together and recalculate the identity. The script adds up the number of identical residues found in the HSPs, account for any overlaps that may have occurred, and then calculate the identity relative to the length of the query sequence. In this example, the three HSPs do not overlap and the sum of their identities was 345. Dividing the sum of the identity against the query length of 568 yielded an identity value of 60.7%.
Fungal Genomes
1. Downloaded 49 genome files.
2. Converted genome files to BLAST databases using mpiformatdb.sh.

BLAST Comparison
1. Queried *Neurospora crassa* against all other genomes using tBLASTn.
2. Parsed BLAST results using a Perl script (Appendix 2.3).
3. Calculated means of BLAST statistics (bits, e-value, identity) for each gene.

Gene Selection
1. Retained *N. crassa* genes present in at least 40 of 48 genomes with an e-value less than e^-5.
2. Systematically selected every 36th genes from the genome to produce a set of 165 genes to analyze.

Neighbor-Joining Tree Generation
1. Retrieved homologous sequences of selected genes from other genomes.
3. Manually trimmed alignments in BioEdit v7.0.5.3.
4. Generated a Neighbor-Joining tree in ClustalX v2.0 for each gene.

Comparisons of Trees to the Reference Tree
1. Identified most common tree topology (T1) for use as a reference.
2. Compared and scored the resolution of all trees according to the reference T1.

Selection of BLAST Statistic to Represent Phylogenetic Information
1. Compared resolution scores of whole set of gene trees (165) to BLAST means.
2. Generated histogram of resolution scores according to single-genome identity values.

Recalculation of Mean Identity with more Query Genomes
1. Selected eight genomes, two from each of the four classes, to recalculate mean identity.
2. Queried the eight selected genomes against all others using tBLASTn.
3.Parsed BLAST results using a Perl script (Appendix 2.3).
4. Calculated mean identities for each gene.
5. Generated histogram of resolution scores according to multi-genome mean identity scores.
6. Determined range of mean identity useful for phylogenetics at the class level.

Figure 2.3. Flowchart of the steps in the comparative genomics method used in this study to identify phylogenetically informative genes.
Figure 2.4. Unrooted single-gene distance tree generated with the Neighbor-Joining algorithm implemented in ClustalX 2.0 using 49 filamentous ascomycetes. The class level topology of this tree, named T1, was the most common topology appearing 67 times out of the 165 genes examined. Resolution scores were assigned to each of the 165 single-gene trees based on similarities or deviations from the class-level topology as depicted in T1. The Sordariomycetes and Leotiomycetes were sister groups while the Dothideomycetes and Eurotiomycetes were sister groups. Class nodes and orders nodes are marked on the tree. Scale bar represents the number of nucleotide substitutions per site.
Figure 2.5. Unrooted single-gene distance tree generated with the Neighbor-Joining algorithm implemented in ClustalX 2.0 using 49 filamentous ascomycetes. The class level topology of this tree, named T2, was the second most common topology appearing 18 times out of the 165 genes examined. In the T2 topology, the Leotiomycetes have been placed within the Eurotiomycetes. However, in this specific tree, the Leotiomycetes split the Eurotiales and the Onygenales and the node has been indicated with a circle. Not all trees sharing this topology have the circled node in the same place as some trees placed the node within the Eurotiales. Scale bar indicates the number of substitutions per site.
Figure 2.6. The single-genome mean identity score was calculated by comparing each of the 9,839 Neurospora crassa genes against 48 other filamentous ascomycete genomes separately in tBLASTn. Subsequently, 165 genes were chosen for phylogenetic analyses. This histogram shows the relative distribution of the 165 genes among five resolution states within 20 equally divided mean percent identity bins between 0%, in which no sites were identical, to 100% where all sites were identical (0-5%, 5-10%, 10-15%, 15-20%, 20-25%, 25-30%, 30-35%, 35-40%, 40-45%, 45-50%, 50-55%, 55-60%, 60-65%, 65-70%, 70-75%, 75-80%, 80-85%, 85-90%, 90-95%, 95-100%). Resolution states of the selected genes correspond with the number of resolved classes out of the four expected classes of T1 and are represented as different shadings. The greatest amount of resolution is four resolved classes out of the four expected classes. The numbers of genes in each of the mean identity bins are indicated above the graph.
Figure 2.7. Comparison of the distribution of genes among mean identity bins as increasingly more query genomes were used to calculate the mean identity of each gene. One to eight genomes were used to calculate the mean identity. Mean identities were retrieved by parsing BLAST outputs with a BioPerl script and the Data Analysis package in Excel 2007 was used to calculate the histogram values. The bin width for the histograms was 4% mean identity. The single genome mean identity bins show a bimodal distribution with a peak around a mean identity of 14, and a smaller peak at around 34-38. The addition of a second genome pulled the left side of the distribution in and the peak is shifted to a mean identity of 34. As increasingly more genomes were included in the calculation for mean identity, the distribution becomes right skewed around a mean of 34.
Figure 2.8. The eight-genome mean identity score was calculated by comparing each of the 9,839 Neurospora crassa genes and their homologs in seven other genomes against the corresponding 48 other filamentous ascomycete genomes separately in tBLASTn. The eight-genome mean identity values were then assigned to the 165 genes that were previously chosen for phylogenetic analyses. This histogram shows the relative distribution of the 165 genes among five resolution states within 20 equally divided mean percent identity bins between 0 and 100%. Resolution states of the selected genes correspond with the number of resolved classes out of the four expected classes of T1 and are represented as different shadings. The greatest amount of resolution is four resolved classes out of the four expected classes. The numbers of genes in each of the mean identity bins are indicated above the graph.
Chapter 3: Comparisons of Superalignment Trees to Single-Gene Trees

3.1 Background

3.1.1 Introduction

Researchers working on different taxonomic groups may prefer different types of characteristics, different genes, and often, different tree-building algorithms for the study of evolutionary relationships or the classification of organisms. There are both historical and practical reasons for the preference of different characteristics or different methods. Some fields have continued to focus heavily on morphological characteristics but others, such as mycology where relatively few distinguishing morphological features are available, have left behind the use of morphology except for descriptions of new species (Hawksworth et al. 2011). This should allow for new species that cannot be cultured to be described using genes and gene barcodes (Hawksworth et al. 2011).

One major change that has occurred within the previous few years is the rapidly falling cost of genome sequencing. Associated with the falling cost is the explosion in the amount of sequencing data being made available in online repositories such as GenBank (Benson et al. 2008). Molecular phylogenetics has traditionally focused on single-gene phylogenies, but many studies are now moving beyond single genes and are focusing on multiple genes. However, not all genes may be equally suitable for phylogenetics at specific taxonomic levels. In addition, it is not clear how many genes should be used in multigene phylogenetic analyses for greatest efficiency and optimal results. In the previous Chapter, a range of sequence divergences was found to be useful for phylogenetic analyses at the Class level in Fungi. This Chapter examines whether using more genes together can provide a more strongly supported phylogeny than a
single gene, and investigates the number of genes that are needed for strongly supported multigene phylogenies.

3.1.2 Multigene Trees

The use of multiple genes has several advantages over a single gene. The evidence of the evolutionary history of a single species is carried by the genes in the genome, but not all the genes are under the same selective pressures. Although the phylogenetic signals over time can become diluted and scattered by random mutations and large-scale genome events, a sufficiently large sample may allow the signals to come through since the combination of many genes in a single dataset may amplify the true underlying phylogenetic signals and overpower the stochastic noise in the dataset (Delsuc et al. 2005; Rokas et al. 2003). This may allow for a more reliable and statistically supported phylogeny to be generated, which can resolve nodes that a single gene may not be able to, and perhaps allow elucidation of relationships at deeper taxonomic levels (Barrett et al. 1991; Fitzpatrick et al. 2006). Increased gene sampling has been found to increase phylogenetic accuracy more than increased taxa sampling (Rokas and Carrol 2005). Problems associated with single-gene trees can also affect multigene datasets such as compositional bias, recombination, and horizontal gene transfer, but these can be avoided through a careful evaluation of the genes used in the analyses (Fitzpatrick et al. 2006; Rannala and Yang 2008).

The benefits conferred by additional genes in a dataset need to be balanced by the additional computational time and cost per gene added. Computing power is growing constantly and the average desktop computer is now sufficient for running many phylogenetic tasks, but high performance computing resources are still needed for large datasets, and not all research labs will have such access. Moreover, depending on the tree-building algorithm, more computing power and time than what is available or feasible may be required.
Rokas et al. (2003) suggested that a minimum of 20 genes was necessary for a well-resolved and supported tree. In response, follow-up studies were conducted and these showed that sampling more taxa may reduce long-branch attractions (LBAs) and consequently reduce the number of genes required for a well-supported tree (Gatesy et al. 2007; Hedtke et al. 2006). The results from Chapter 2 demonstrated that well-resolved trees for filamentous ascomycetes at the class level could be obtained from single gene analyses. However, only a third of the total number of genes showed a topology that matched the currently accepted phylogeny for filamentous ascomycetes at the class level with strong bootstrap support. By using a larger dataset and concatenating gene alignments, it may be possible to obtain better branch support, resolve more nodes in the tree, and consistently retrieve the same tree with different gene sets (Delsuc et al. 2005). Identifying an optimal or saturation number of genes required to generate a well-supported and accurate tree may allow for more efficient use of computational power.

There are two general methods for working with multiple genes in phylogenetics: superalignments and supertrees. The supertree methods combine several gene trees into a single large phylogeny (Rannala and Yang 2008). This is ideal when combining previously published trees with currently available gene data and may reduce the computational requirements (Rannala and Yang 2008; Ropiquet et al. 2009). Supertrees require trees to be converted into a matrix before being combined, and subsequently regenerated into a single tree. Alternatively, superalignments involve combining sequence data prior to generating the tree. The superalignment method is more commonly used because superalignments are just larger alignments of separate genomic regions, and tree-generating methods available right now can be used when working with a superalignment (Philippe et al. 2005a; Rannala and Yang 2008). Thus, superalignment methods were chosen for this study.
3.1.3 Objectives

The objective of this study was to compare the resolution of single-gene trees (Chapter 2) to superalignment trees generated from various numbers of homologous genes identified from available genomes of filamentous ascomycetes. This involved an evaluation of the number of concatenated genes required for a robust tree and an investigation of whether or not multigene trees are more reliable than single-gene trees in terms of arriving at the same tree topology or the reference topology (T1 in Chapter 2) of the filamentous ascomycetes. In addition, the congruency between the superalignment trees and the currently accepted phylogeny was examined.

3.2 Methods

The steps used in the methods have been outlined as a flowchart in Figure 3.1. Gene sets of various sizes were generated by systematically selecting *Neurospora crassa* genes from bins based on their single-genome mean identity, eight-genome mean identity, and at random. Genes and their homologous sequences in the 48 other filamentous ascomycetes were retrieved, aligned, and concatenated within their sets. Next, trees were generated and compared to the reference T1 topology determined in Chapter 2. The major steps are summarized (Figure 3.1), and each step of the flowchart has been detailed in the following sections.

3.2.1 Random Gene Selection for Multigene Trees

Previously in Chapter 2, *Neurospora crassa* genes were filtered by setting an e-value cutoff of 1E-5 when parsing the BLAST results, and retaining the genes that had a homolog in at least 40 of the 48 genomes being compared. This reduced the list of 9,839 genes down to 6,075 genes, which were then systematically sampled to produce the single-gene trees. In order to
reduce the number of blank sequences and improve the integrity of the multigene datasets, an additional filter was applied to remove the genes that did not have a homolog in all 49 genomes being examined. This yielded a pool of 3,584 genes.

From the pool of 3,584 genes, ten sets of 5, 10, and 15 genes were independently sampled without replacement. Based on preliminary results, ten sets of 2, 3, and 4 genes were also independently sampled without replacement.

### 3.2.2 Superalignment Tree Construction

For all selected genes, the fastacmd program from the NCBI BLAST v2.2.13 package (Darling et al. 2003) was used to retrieve the top-scoring homologous nucleotide sequences (Chapter 2) from all 49 filamentous ascomycete genomes. Sequences of each gene were copied to the Checkers computing cluster, a part of the Western Canada Research Grid (Westgrid) accessible at [www.westgrid.ca](http://www.westgrid.ca). The sequences were subsequently aligned on Checkers using the EINSI algorithm in MAFFT v6 (Katoh et al. 2009), and each alignment was trimmed in TrimAl v1.2 (Capella-Gutierrez et al. 2009) with a gap threshold of 0.75. In other studies using TrimAl, the gap thresholds tended to range between 0.75 and 0.80 (Amend et al. 2010; Huerta-Cepas et al. 2010; Schluter et al. 2011) and thus, 0.75 was used as the threshold. With a gap threshold of 0.75, TrimAl will remove a position in the alignments if more than 25% of the sequences have a gap in that position. Manual trimming of alignments is frequently found in the literature, and involves removing the unaligned ends of some sequences and the gaps within the sequences. Sometimes, these gaps result from incorrectly sequenced regions, or incorrectly annotated sequences. Removing the gaps and ends may improve the fidelity of the phylogenetic signals, but manual removal may be subjective. As a result, TrimAl was used because it can evaluate each
position in the alignment objectively, and remove the gap if it does not exceed the gap threshold limit.

In each of the gene sets (described above), a script (Appendix 3.2) was used to concatenate all sequence alignments into superalignments. The script outputted all concatenated superalignments in FASTA format. Subsequently, a maximum likelihood (ML) method was also used to generate a tree from each superalignment, using RAxML v7.2.8 Alpha (Stamatakis 2006) as compiled on Checkers. To analyze the data with RAxML, the superalignment FASTA files were converted to Phylip format (Felsenstein 1989) using a Perl script (Appendix 3.3). RAxML was then used to generate a ML tree from each superalignment Phylip file with 1000 rapid bootstrap replicates, a random search seed of 123, and using the GTR+Γ model. Each tree construction was performed using 20 cores and 10 GB of memory. Because of the large size of the superalignments, model tests were performed on only three randomly selected superalignments using Modeltest (Posada and Crandall 1998) in PAUP* (Swofford 2003) on a desktop computer with 3 GB of RAM running Windows XP.

3.2.3 Giant Superalignment Tree Construction

For the large superalignment trees, ten sets of 100 genes were independently and randomly sampled without replacement from the pool of 3,584 genes. For each selected gene, Fastacmd was used to retrieve the sequences from all 49 filamentous ascomycete genomes. Subsequently, a multiple sequence alignment was performed using MAFFT-EINSI followed by trimming in TrimAL with a gap threshold of 0.75. A Perl script was used to concatenate each set of 100 alignments into superalignments in no particular order (Appendix 3.2). Each superalignment was converted to Phylip format using a Perl script (Appendix 3.3) before being
analyzed with RAxML. An ML tree with 1,000 bootstrap replicates was then generated from each superalignment using RAxML under the GTR+Γ model and a random search seed of 123.

3.3 Results and Discussion

The steps used in the methods have been outlined as a flowchart in Figure 3.1, and each step of the flowchart is more fully discussed in the following sections.

3.3.1 Superalignments Trees from Random Sets of Homologous Genes

Unlike the pool of 6,075 genes used in Chapter 2 for the single-gene trees, a more stringently filtered selection of genes was used for the multigene trees to allow for potentially more accurate inferences. Genes that did not have a BLAST hit scoring higher than the e-value cutoff of 1E-5 in all 49 genomes being examined were removed. This resulted in a pool containing 3,584 genes.

From the 3,584 genes, ten samples of 5, 10, and 15 genes were independently and randomly selected without replacement. The sequences for each gene were retrieved from the BLAST databases, aligned and trimmed. Depending on how much excess information is present at the ends of the sequences, TrimAl may remove up to 75% of the alignment file. For alignments of well-annotated and highly conserved genes, there may be no changes in alignment after using TrimAl. Appendix 3.7 gives an example of an alignment before and after trimming with TrimAl. A Perl script (Appendix 3.2) was use to concatenate the sequence alignments of the selected genes in each sample into superalignments containing 5, 10, and 15 genes.

In total, there were 30 superalignments, and these were used to construct ML trees using RAxML with 1,000 bootstrap replicates under the GTR+Γ model. The evolutionary model parameters used by RAxML to generate the trees are detailed in Table 3.1. All 30 trees were
resolved with all taxa placed into the four expected classes (Eurotiomycetes, Dothideomycetes, Leotiomycetes, and Sordariomycetes), and were concordant with the T1 topology (Table 3.2). Unlike the single-gene trees, all superalignment trees, so far, resolved species into the four classes that were concordant with the T1 topology.

The preliminary results showed that multigene trees made with at least five genes can yield a topology that matched T1. However, the results did not show a lower limit for the number of genes that should be used. Thus, a second set of multigene trees were generated using fewer than 5 genes.

From the pool of 3,584 homologous filamentous ascomycete genes, ten sets of two, three, and four genes were independently and randomly sampled without replacement. Using the same procedure as mentioned above for the 5-, 10-, and 15-gene trees, multigene trees were generated from the two-, three-, and four-gene samples. The model parameters used by RAxML to generate the trees are shown in Table 3.1.

Thirty multigene trees were generated using superalignments containing fewer than five genes. With two and four genes, seven out of the ten trees showed the T1 topology, while three of the multigene trees generated from three genes showed T1 (Table 3.2). The results of the analyses of 60 multigene trees suggests that at least five genes may be required for producing well-resolved and consistent multigene trees when using homologous genes that are present in all genomes being tested. Interestingly, the lowest occurrence of multigene trees showing T1 occurred when three genes were used. This may indicate that not only were the phylogenetic signals increasing, but noise had somehow increased as well. However, once a fourth gene was added, the noise was relatively smaller than the phylogenetic signals, and then with the addition
of a fifth gene, the signal to noise ratio may have reached a sufficient level for more consistent results.

### 3.3.2 Comparisons of 100-Gene Superalignment Trees

Because the results with 5 to 15 genes showed that the topologies of the generated superalignment trees were the same, large superalignments with over 15 random genes may be redundant. However, it is possible that the trees generated with between 5 to 15 genes are not showing the true topology and instead, are showing the results of strong noise from different evolutionary histories. To confirm the T1 topology, very large superalignment trees were generated with 100 genes to thoroughly saturate the phylogenetic signals, and to see if the resultant topologies at the class level differed from superalignment trees generated using 5 to 15 genes.

All ten 100-gene trees were completed within 61 hours (20 processor cores, 20 GB memory). The ten 100-gene superalignment trees consistently showed the same topology and relationships at the class and order level with 100% bootstrap support at the major nodes, and this topology also matched T1 (Table 3.3). Some relationships between orders did not receive 100% bootstrap support but those usually involved orders with only one representative genome. For further discussion regarding the computational constraints and bottlenecks encountered throughout the superalignment ML tree generation process, the choices of evolutionary models, and the model tests, please see Appendix 3.9.

### 3.3.3 Superalignment Trees and Single-Gene Trees

One of the purposes of this part of the study was to determine whether multigene trees outperform single-gene trees in terms of resolving relationships between classes in the
filamentous ascomycetes. In Chapter 2, we demonstrated that trees generated from some individual genes with specific amounts of sequence divergence were sufficient for resolving such relationships. Although the single largest group of trees was concordant with a single topology, that group only represented a third of all genes tested.

In this Chapter, the majority of superalignment (multigene) trees and all multigene trees with more than five genes also agreed with the T1 reference tree. Because a larger percentage of trees matched the reference, we concluded that superalignment trees may be more reliable than single-gene trees in separating filamentous ascomycete species into their taxonomic classes if a sufficient number of genes were used. This corroborates the conclusion that having more genes in the dataset may yield a more reliable tree (Delsuc et al. 2005; Rokas and Carroll 2005).

### 3.3.4 Comparisons of Trees to 18S

The T1 topology that was used as a reference tree also matched the 18S rDNA tree, which has been used extensively for phylogenetics in animals and fungi, and is consistent with the current Ascomycetes tree of life (Schoch et al. 2009b). The results suggest that 18S rDNA alone may be sufficient for conducting phylogenetic studies at the class level and support its continued usage in phylogenetics of the filamentous ascomycetes at the class level. Although only a plurality of single-gene trees matched the 18S rDNA trees (40%, 67 trees out of 165, Table 3.3), all superalignment trees generated with more than 5 genes matched the 18S rDNA tree (Table 3.3). The results suggested that superalignment trees may be more reliable than single-gene trees given sufficient taxa and gene sampling when attempting to infer evolutionary relationships of the filamentous ascomycetes as they arrived at the same topology more frequently than the single-gene trees. In addition, the superalignment trees present a more compelling case because ML algorithms are often considered more robust than distance methods.
when an optimal model of evolution is used (Holder and Lewis 2003; Stamatakis et al. 2005; Whelan et al. 2001). More of the single-gene trees might have resolved and matched T1/18S if they had been generated using the more computationally intensive ML algorithm instead of the NJ implementation in ClustalW.

Based on the single-gene trees and multigene trees, 18S rDNA may be useful for inferring a tree of the filamentous ascomycetes at higher taxonomic levels such as orders, classes, and perhaps even phyla, which is in agreement with the conclusions made by Rokas et al. (2002). However, because there were not enough genomes to examine taxonomic relationships below the family level, it is not known if the use of 18S rDNA for inferring phylogenies below the family level will yield well-supported trees. A greater number of taxa for the lower taxonomic levels (family, genera) are needed to fully test this. It is possible that 18S rDNA alone may not be able to yield a well-resolved tree with high bootstrap values when many other species are considered and in these cases, combining 18S rDNA with a random sample of at least five genes that have a homolog in all the species being analyzed may further increase resolution and bootstrap support in a multigene tree.

3.3.5 What is the Optimal Number of Genes for a Multigene Tree?

The exact number of genes required for a well-resolved and strongly supported tree has been reported to be dependent on the project and the taxa being used (Gatesy et al. 2007; Hedtke et al. 2006). In addition, it has been previously shown that poor taxon sampling may lead to systematic errors (Hedtke et al. 2006). To avoid systematic errors that may arise from poor taxa sampling, we examined 49 filamentous ascomycetes in contrast to the eight yeast genomes in the Rokas et al. (2003) study (which was the number of genomes available at that time). In doing so,
it is possible that the effects of some LBAs were reduced, and consequently, reduced the need for increased gene sampling (Gatesy et al. 2007; Hedtke et al. 2006).

We previously found in Chapter 2 that around 40% of the genes in the *N. crassa* genome were able to yield well-resolved trees of filamentous ascomycete classes that matched the reference, and that multigene trees may not be necessary. Here, we tried to determine if multigene trees can more consistently yield the same resolution when compared to single-gene trees, and if so, how many genes may be required to mitigate the effects of stochastic noise. Although some individual genes may be sufficient for producing a well-resolved tree (Chapter 2), using multiple genes to construct a tree may be more reliable because a greater frequency of multigene trees can arrive at the same topology than the single-gene trees. The results also indicated that using 100 genes in a multigene tree may be excessive as the tree topologies at the class level were essentially the same when using 10, 15, and 100 genes. Moreover, a concatenation of five genes appeared to be sufficient for resolving class level relationships in the filamentous ascomycetes, but using fewer than five genes may lead to potentially erroneous inferences.
Table 3.1. Parameters used in the Maximum Likelihood (ML) analyses for the superalignment generated from the eight genomes mean identity bins under the General Time Reversible plus gamma (GTR+\Gamma) model as estimated by RAxML.

<table>
<thead>
<tr>
<th>Trees</th>
<th>Base Frequencies</th>
<th>Substitution Rates</th>
<th>Gamma Shape</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>2genesreplicate0</td>
<td>0.25</td>
<td>0.27</td>
<td>0.25</td>
</tr>
<tr>
<td>2genesreplicate1</td>
<td>0.27</td>
<td>0.26</td>
<td>0.25</td>
</tr>
<tr>
<td>2genesreplicate2</td>
<td>0.23</td>
<td>0.28</td>
<td>0.26</td>
</tr>
<tr>
<td>2genesreplicate3</td>
<td>0.26</td>
<td>0.28</td>
<td>0.26</td>
</tr>
<tr>
<td>2genesreplicate4</td>
<td>0.22</td>
<td>0.29</td>
<td>0.25</td>
</tr>
<tr>
<td>2genesreplicate5</td>
<td>0.25</td>
<td>0.26</td>
<td>0.27</td>
</tr>
<tr>
<td>2genesreplicate6</td>
<td>0.24</td>
<td>0.29</td>
<td>0.26</td>
</tr>
<tr>
<td>2genesreplicate7</td>
<td>0.24</td>
<td>0.30</td>
<td>0.26</td>
</tr>
<tr>
<td>2genesreplicate8</td>
<td>0.26</td>
<td>0.26</td>
<td>0.27</td>
</tr>
<tr>
<td>2genesreplicate9</td>
<td>0.25</td>
<td>0.27</td>
<td>0.27</td>
</tr>
<tr>
<td>3genesreplicate0</td>
<td>0.26</td>
<td>0.26</td>
<td>0.29</td>
</tr>
<tr>
<td>3genesreplicate1</td>
<td>0.26</td>
<td>0.26</td>
<td>0.27</td>
</tr>
<tr>
<td>3genesreplicate2</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
</tr>
<tr>
<td>3genesreplicate3</td>
<td>0.27</td>
<td>0.25</td>
<td>0.27</td>
</tr>
<tr>
<td>3genesreplicate4</td>
<td>0.22</td>
<td>0.29</td>
<td>0.25</td>
</tr>
<tr>
<td>3genesreplicate5</td>
<td>0.25</td>
<td>0.28</td>
<td>0.26</td>
</tr>
<tr>
<td>3genesreplicate6</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
</tr>
<tr>
<td>3genesreplicate7</td>
<td>0.23</td>
<td>0.29</td>
<td>0.28</td>
</tr>
<tr>
<td>3genesreplicate8</td>
<td>0.26</td>
<td>0.26</td>
<td>0.27</td>
</tr>
<tr>
<td>3genesreplicate9</td>
<td>0.23</td>
<td>0.28</td>
<td>0.27</td>
</tr>
<tr>
<td>4genesreplicate0</td>
<td>0.23</td>
<td>0.28</td>
<td>0.26</td>
</tr>
<tr>
<td>4genesreplicate1</td>
<td>0.23</td>
<td>0.29</td>
<td>0.27</td>
</tr>
<tr>
<td>4genesreplicate2</td>
<td>0.26</td>
<td>0.26</td>
<td>0.25</td>
</tr>
<tr>
<td>4genesreplicate3</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
</tr>
<tr>
<td>4genesreplicate4</td>
<td>0.23</td>
<td>0.27</td>
<td>0.26</td>
</tr>
<tr>
<td>4genesreplicate5</td>
<td>0.24</td>
<td>0.28</td>
<td>0.27</td>
</tr>
<tr>
<td>4genesreplicate6</td>
<td>0.24</td>
<td>0.29</td>
<td>0.26</td>
</tr>
<tr>
<td>4genesreplicate7</td>
<td>0.27</td>
<td>0.26</td>
<td>0.29</td>
</tr>
<tr>
<td>4genesreplicate8</td>
<td>0.23</td>
<td>0.28</td>
<td>0.27</td>
</tr>
<tr>
<td>4genesreplicate9</td>
<td>0.25</td>
<td>0.27</td>
<td>0.27</td>
</tr>
<tr>
<td>5genesreplicate0</td>
<td>0.25</td>
<td>0.26</td>
<td>0.28</td>
</tr>
<tr>
<td>5genesreplicate1</td>
<td>0.22</td>
<td>0.28</td>
<td>0.26</td>
</tr>
<tr>
<td>5genesreplicate2</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
</tr>
<tr>
<td>5genesreplicate3</td>
<td>0.25</td>
<td>0.26</td>
<td>0.27</td>
</tr>
<tr>
<td>5genesreplicate4</td>
<td>0.23</td>
<td>0.28</td>
<td>0.26</td>
</tr>
<tr>
<td>5genesreplicate5</td>
<td>0.23</td>
<td>0.28</td>
<td>0.25</td>
</tr>
<tr>
<td>5genesreplicate6</td>
<td>0.24</td>
<td>0.27</td>
<td>0.27</td>
</tr>
</tbody>
</table>
Table 3.1 continued

<table>
<thead>
<tr>
<th>5genesreplicate7</th>
<th>0.23</th>
<th>0.29</th>
<th>0.26</th>
<th>0.21</th>
<th>1.48</th>
<th>2.81</th>
<th>1.08</th>
<th>1.08</th>
<th>3.45</th>
<th>1.00</th>
<th>0.71</th>
</tr>
</thead>
<tbody>
<tr>
<td>5genesreplicate8</td>
<td>0.25</td>
<td>0.27</td>
<td>0.27</td>
<td>0.21</td>
<td>1.67</td>
<td>3.09</td>
<td>1.33</td>
<td>1.21</td>
<td>4.50</td>
<td>1.00</td>
<td>0.65</td>
</tr>
<tr>
<td>5genesreplicate9</td>
<td>0.26</td>
<td>0.26</td>
<td>0.26</td>
<td>0.22</td>
<td>1.81</td>
<td>4.10</td>
<td>1.12</td>
<td>1.29</td>
<td>5.83</td>
<td>1.00</td>
<td>0.51</td>
</tr>
<tr>
<td>10genesreplicate0</td>
<td>0.24</td>
<td>0.27</td>
<td>0.28</td>
<td>0.22</td>
<td>1.81</td>
<td>3.31</td>
<td>1.27</td>
<td>1.34</td>
<td>4.77</td>
<td>1.00</td>
<td>0.62</td>
</tr>
<tr>
<td>10genesreplicate1</td>
<td>0.23</td>
<td>0.27</td>
<td>0.27</td>
<td>0.23</td>
<td>1.87</td>
<td>3.32</td>
<td>1.19</td>
<td>1.37</td>
<td>3.89</td>
<td>1.00</td>
<td>0.69</td>
</tr>
<tr>
<td>10genesreplicate2</td>
<td>0.25</td>
<td>0.28</td>
<td>0.26</td>
<td>0.20</td>
<td>1.73</td>
<td>3.60</td>
<td>1.27</td>
<td>1.32</td>
<td>5.15</td>
<td>1.00</td>
<td>0.57</td>
</tr>
<tr>
<td>10genesreplicate3</td>
<td>0.24</td>
<td>0.26</td>
<td>0.26</td>
<td>0.23</td>
<td>1.88</td>
<td>4.02</td>
<td>1.27</td>
<td>1.42</td>
<td>4.70</td>
<td>1.00</td>
<td>0.61</td>
</tr>
<tr>
<td>10genesreplicate4</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
<td>0.23</td>
<td>1.55</td>
<td>3.09</td>
<td>1.13</td>
<td>1.29</td>
<td>3.62</td>
<td>1.00</td>
<td>0.73</td>
</tr>
<tr>
<td>10genesreplicate5</td>
<td>0.24</td>
<td>0.27</td>
<td>0.25</td>
<td>0.24</td>
<td>1.60</td>
<td>3.61</td>
<td>1.19</td>
<td>1.22</td>
<td>4.91</td>
<td>1.00</td>
<td>0.49</td>
</tr>
<tr>
<td>10genesreplicate6</td>
<td>0.25</td>
<td>0.27</td>
<td>0.27</td>
<td>0.22</td>
<td>1.84</td>
<td>3.80</td>
<td>1.23</td>
<td>1.33</td>
<td>4.89</td>
<td>1.00</td>
<td>0.60</td>
</tr>
<tr>
<td>10genesreplicate7</td>
<td>0.23</td>
<td>0.28</td>
<td>0.26</td>
<td>0.24</td>
<td>1.58</td>
<td>3.24</td>
<td>1.23</td>
<td>1.30</td>
<td>3.80</td>
<td>1.00</td>
<td>0.63</td>
</tr>
<tr>
<td>10genesreplicate8</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
<td>0.22</td>
<td>1.66</td>
<td>3.58</td>
<td>1.10</td>
<td>1.31</td>
<td>5.01</td>
<td>1.00</td>
<td>0.54</td>
</tr>
<tr>
<td>10genesreplicate9</td>
<td>0.24</td>
<td>0.28</td>
<td>0.26</td>
<td>0.22</td>
<td>1.64</td>
<td>3.54</td>
<td>1.28</td>
<td>1.33</td>
<td>4.50</td>
<td>1.00</td>
<td>0.62</td>
</tr>
<tr>
<td>15genesreplicate0</td>
<td>0.24</td>
<td>0.28</td>
<td>0.26</td>
<td>0.22</td>
<td>1.66</td>
<td>3.38</td>
<td>1.26</td>
<td>1.34</td>
<td>4.72</td>
<td>1.00</td>
<td>0.56</td>
</tr>
<tr>
<td>15genesreplicate1</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
<td>0.21</td>
<td>1.80</td>
<td>3.67</td>
<td>1.24</td>
<td>1.31</td>
<td>5.20</td>
<td>1.00</td>
<td>0.61</td>
</tr>
<tr>
<td>15genesreplicate2</td>
<td>0.23</td>
<td>0.27</td>
<td>0.26</td>
<td>0.23</td>
<td>1.77</td>
<td>3.90</td>
<td>1.24</td>
<td>1.40</td>
<td>4.71</td>
<td>1.00</td>
<td>0.58</td>
</tr>
<tr>
<td>15genesreplicate3</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
<td>0.22</td>
<td>1.63</td>
<td>3.37</td>
<td>1.17</td>
<td>1.28</td>
<td>4.90</td>
<td>1.00</td>
<td>0.57</td>
</tr>
<tr>
<td>15genesreplicate4</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
<td>0.23</td>
<td>1.86</td>
<td>3.89</td>
<td>1.15</td>
<td>1.34</td>
<td>5.22</td>
<td>1.00</td>
<td>0.57</td>
</tr>
<tr>
<td>15genesreplicate5</td>
<td>0.25</td>
<td>0.26</td>
<td>0.26</td>
<td>0.22</td>
<td>1.91</td>
<td>3.86</td>
<td>1.25</td>
<td>1.41</td>
<td>5.26</td>
<td>1.00</td>
<td>0.61</td>
</tr>
<tr>
<td>15genesreplicate6</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
<td>0.24</td>
<td>1.70</td>
<td>3.35</td>
<td>1.13</td>
<td>1.28</td>
<td>4.24</td>
<td>1.00</td>
<td>0.63</td>
</tr>
<tr>
<td>15genesreplicate7</td>
<td>0.24</td>
<td>0.28</td>
<td>0.26</td>
<td>0.22</td>
<td>1.53</td>
<td>3.21</td>
<td>1.18</td>
<td>1.23</td>
<td>4.66</td>
<td>1.00</td>
<td>0.57</td>
</tr>
<tr>
<td>15genesreplicate8</td>
<td>0.24</td>
<td>0.28</td>
<td>0.26</td>
<td>0.22</td>
<td>1.74</td>
<td>3.41</td>
<td>1.26</td>
<td>1.31</td>
<td>4.28</td>
<td>1.00</td>
<td>0.66</td>
</tr>
<tr>
<td>15genesreplicate9</td>
<td>0.23</td>
<td>0.28</td>
<td>0.26</td>
<td>0.23</td>
<td>1.56</td>
<td>3.31</td>
<td>1.17</td>
<td>1.35</td>
<td>4.32</td>
<td>1.00</td>
<td>0.59</td>
</tr>
<tr>
<td>100genesreplicate0</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
<td>0.23</td>
<td>1.68</td>
<td>3.54</td>
<td>1.24</td>
<td>1.32</td>
<td>4.53</td>
<td>1.00</td>
<td>0.61</td>
</tr>
<tr>
<td>100genesreplicate1</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
<td>0.22</td>
<td>1.74</td>
<td>3.53</td>
<td>1.22</td>
<td>1.32</td>
<td>4.66</td>
<td>1.00</td>
<td>0.58</td>
</tr>
<tr>
<td>100genesreplicate2</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
<td>0.22</td>
<td>1.75</td>
<td>3.64</td>
<td>1.24</td>
<td>1.32</td>
<td>4.78</td>
<td>1.00</td>
<td>0.58</td>
</tr>
<tr>
<td>100genesreplicate3</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
<td>0.22</td>
<td>1.73</td>
<td>3.61</td>
<td>1.22</td>
<td>1.31</td>
<td>4.77</td>
<td>1.00</td>
<td>0.57</td>
</tr>
<tr>
<td>100genesreplicate4</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
<td>0.22</td>
<td>1.79</td>
<td>3.55</td>
<td>1.25</td>
<td>1.36</td>
<td>4.62</td>
<td>1.00</td>
<td>0.63</td>
</tr>
<tr>
<td>100genesreplicate5</td>
<td>0.25</td>
<td>0.27</td>
<td>0.26</td>
<td>0.22</td>
<td>1.70</td>
<td>3.51</td>
<td>1.20</td>
<td>1.31</td>
<td>4.69</td>
<td>1.00</td>
<td>0.59</td>
</tr>
<tr>
<td>100genesreplicate6</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
<td>0.23</td>
<td>1.71</td>
<td>3.58</td>
<td>1.20</td>
<td>1.31</td>
<td>4.53</td>
<td>1.00</td>
<td>0.62</td>
</tr>
<tr>
<td>100genesreplicate7</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
<td>0.22</td>
<td>1.67</td>
<td>3.38</td>
<td>1.20</td>
<td>1.30</td>
<td>4.45</td>
<td>1.00</td>
<td>0.60</td>
</tr>
<tr>
<td>100genesreplicate8</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
<td>0.22</td>
<td>1.73</td>
<td>3.52</td>
<td>1.25</td>
<td>1.35</td>
<td>4.75</td>
<td>1.00</td>
<td>0.60</td>
</tr>
<tr>
<td>100genesreplicate9</td>
<td>0.24</td>
<td>0.27</td>
<td>0.26</td>
<td>0.23</td>
<td>1.74</td>
<td>3.67</td>
<td>1.23</td>
<td>1.34</td>
<td>4.69</td>
<td>1.00</td>
<td>0.59</td>
</tr>
</tbody>
</table>
Table 3.2. The number of superalignment Maximum Likelihood (ML) trees that matched the reference T1 (18S) topology. A total of 60 trees were constructed using up to 15 genes randomly selected from the set of 3,584 *Neurospora crassa* genes that have a homolog in the 48 other filamentous ascomycete genomes. All ML trees have 49 species and were constructed in RAxML with 1000 BS replicates under the General Time Reversible plus gamma (GTR+Γ) model and a random seed of 123.

<table>
<thead>
<tr>
<th>Topologies of Superalignment Trees</th>
<th>Number of Genes Used to Generate Trees</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Reference T1 Topology</td>
<td>7</td>
</tr>
<tr>
<td>Others</td>
<td>3</td>
</tr>
</tbody>
</table>
Table 3.3. Number of trees sharing the same tree topology based on manual evaluations of phylogenetic trees. Single-gene trees were constructed using ClustalX with 1000 bootstrap replicates from multiple sequence alignments generated through MUSCLE followed by manual trimming in Bioedit. Maximum Likelihood (ML) trees were constructed using RAxML under the General Time Reversible plus gamma (GTR+Γ) with 1000 bootstrap replicates and a random seed of 123. Sequence alignments for the ML trees were performed using MAFFT-EINSI and then trimmed with TrimAL using a gap threshold of 0.75. The unique and unresolved category contained tree topologies that either showed a large polytomy or only occurred once out of all genes examined. The remaining topologies that occurred more than once but not as often as topologies 1, 2, and 3 were in the ‘others’ category.

<table>
<thead>
<tr>
<th>Tree Topology</th>
<th>Single-gene NJ Trees</th>
<th>2- to 4-gene Multigene ML Trees</th>
<th>5- to 15-gene Multigene ML Trees</th>
<th>100-Gene Multigene ML Trees</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology 1 (T1;18S)</td>
<td>67</td>
<td>17</td>
<td>30</td>
<td>10</td>
</tr>
<tr>
<td>Topology 2</td>
<td>18</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Topology 3</td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Unique and Unresolved</td>
<td>38</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Others</td>
<td>27</td>
<td>13</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>165</td>
<td>30</td>
<td>30</td>
<td>10</td>
</tr>
</tbody>
</table>
**Initial Random Gene Selection From Pool of Homologous Genes**

1) Genes from the *Neurospora crassa* genome used in Chapter 2 were also used to generate the superalignments.

2) *N. crassa* genes without a homolog scoring higher than 1e^-5 in the 48 other filamentous ascomycete genomes were removed.

3) From pool of 3,584 homologous genes, ten independent sets of 5, 10, and 15 genes were randomly sampled without replacement.

**Follow-Up Random Gene Selection From Pool of Homologous Genes**

1) From pool of 3,584 homologous genes, ten independent sets of 2, 3, and 4 genes were randomly sampled without replacement.

**Random Gene Selection For Large Superalignment**

1) From pool of 3,584 homologous genes, ten independent sets of 100 genes were randomly sampled without replacement.

**Superalignment Generation**

1) Retrieved homologs of selected genes from other genomes.
3) Trimmed ends and gaps from alignments using TrimAL v1.2.
4) Concatenated all genes in each set into superalignments using a script (Appendix 3.1).

**Maximum Likelihood Tree Generation**

1) Converted superalignments to Phylip format using a script (Appendix 3.2).
2) Trees were generated from each superalignment using RAxML v7.2.8.

**Comparisons of Superalignment Trees to the Reference Tree**

The topology and resolution of all superalignment trees were compared to the reference T1 as determined in Chapter 2.

---

Figure 3.1. Flowchart of the steps used in this study to generate and compare superalignment trees made with various numbers of genes.
Chapter 4: Selecting and Testing Potentially Phylogenetically Informative Genes

4.1 Introduction

4.1.1 Background

Completely sequenced genomes cannot be considered truly ‘complete’ because there are gaps often composed of missing repetitive sequences that are difficult to assemble, and they may not represent the full complement of genes and alleles present within a species as comparative genomic studies continue to reveal previously unknown genes (Hsiang and Baillie 2006). However, there are other issues in their use for phylogenetic assessments. Despite the increasing availability of sequence data, it may not be possible to obtain the genome sequence for all living organisms because of biological or technological limitations. For example, some fungal specimens cannot be easily cultured such as obligate biotrophs, or those that require a complex mixture of nutrients and conditions for growth. Additionally, some specimens may only appear in metagenomic studies and nowhere else (Begerow et al. 2010; Martin et al. 2011; Martin and Martin 2009). In some of these cases, minute quantities of DNA can be extracted but they may not be sufficiently pure or of sufficient quality for whole genome sequencing. Therefore, it may be necessary to return to traditional oligonucleotide synthesis and polymerase chain reaction (PCR) to obtain the sequence of genes of interest for phylogenetics.

Carbone & Kohn (1999) described how ribosomal regions may not be sufficient for intraspecific phylogenetic studies and this has led to the search for other potential genes for phylogenetics. Older methods identified potential genes by performing BLAST on various genes to identify regions conserved across different species that could be used as priming regions (Carbone and Kohn 1999). Over time, this gradually led to increasingly sophisticated methods such as one that profiled potential phylogenetic information in genes (Townsend 2007) and
found that DNA sequences are more phylogenetically informative than amino acid sequences (Townsend et al. 2008a). One study identified two potential phylogenetically informative genes from a large pool of purely single-copy protein-coding loci by generating a reference tree from the concatenated gene set and comparing each single-gene tree to the reference (Aguileta et al. 2008). Degenerate primers for the two genes identified by Aguileta et al. (2008) were later developed for the filamentous ascomycetes and demonstrated to be useful in class to genera level phylogenetics (Schmitt et al. 2009). The results of the study conducted by Aguileta et al. (2008) gave rise to an online publicly accessible database of homologous gene families, the Fungal Phylogenomic Database (FunyBASE). The FunyBASE resource allows other researchers to conduct studies using curated alignments of annotated sequence data. One such study utilized FunyBASE to develop a method for identifying phylogenetic markers in genomes that have already been sequenced (Feau et al. 2011).

Results in Chapters 2 and 3 demonstrated that it may be possible to identify potential phylogenetically useful genes through comparative genomics. In addition, combining a minimum of five genes may yield a resolved tree more often than what may be possible from a random single gene. The purpose of this Chapter was to experimentally test those findings. To facilitate obtaining genes from genomes that have not been sequenced, a large-scale computational method was developed to identify the genes with potential priming sites for PCR. Such priming sites may have potential as universal primers or could be useful for large taxonomic groups, but only certain genes may be sufficiently conserved for this task. In addition, a balance must be struck between the level of gene conservation and phylogenetic usefulness at specific taxonomic levels.
4.1.2 Objectives

The objective of this Chapter was to develop and test a method for automated identification of potential conserved priming sites within genes assessed in previous chapters as possibly being able to resolve classes of the filamentous ascomycetes. In Chapters 2, around 40% of the genes in the *Neurospora crassa* genome in alignments with 48 other filamentous ascomycetes were found to yield single-gene trees that were concordant with the currently accepted tree of the filamentous ascomycetes at the class level (T1). By further focusing on a specific range of sequence divergence (45-60% mean identity), 64% of the genes yielded single-gene trees that also matched T1. In this Chapter, genes within different bins of sequence divergence were screened to rapidly identify priming sites across different species of the filamentous ascomycetes. In addition, primers based on the priming regions were synthesized and tested on DNA extracted from tissues of filamentous ascomycetes that have not yet been sequenced. The amplified regions were used in phylogenetic analyses to assess whether they grouped by class and maintained the T1 topology as predicted from related sequence genomes.

4.2 Methods

4.2.1 Gene Screening and Primer Design

In Chapter 2, 165 *Neurospora crassa* genes were systematically selected and aligned with other filamentous ascomycetes to generate single-gene trees. These topologies were used to assess the levels of sequence divergence related to yielding well-resolved trees at the class level. A histogram of the resolution results based on the eight-genome mean identity among the 165 genes (Figure 2.7) was examined to identify the range of mean identity that could yield well-resolved single-gene trees. Based on the histogram (Figure 2.7), the mean identity bins between
50% and 90% were found to have the largest proportion of genes that yielded resolved trees, and thus were selected for screening. Between 50% and 90% mean identity, there were four 10 percentile mean identity bins and each was split in half to yield eight 5% identity bins. From each of the eight bins, five genes were randomly selected as well as their trimmed sequence alignments, which had been used to generate the single-gene trees in Chapter 2.

A Perl script was written to screen all 40 of the sequence alignments to determine if they had potential forward and reverse priming sites. This Perl script uses a rolling window that examines consecutive stretches of 18 base pairs (bp) in a multiple sequence alignment for potential priming sites. It first passed the alignment file through TrimAL v1.2 (Capella-Gutierrez et al. 2009) with the ‘-scc’ option which outputted the amount of similarity at each site as calculated using Euclidean distances. For each site, similarity values ranged between 0 for poorly conserved sites and 1 for perfectly conserved sites. The script calculated the total similarity of the 18 consecutive sites with greater weighting for high similarity in the last 6 bp at the 3’ end. The greater weight was calculated by adding an additional 50% bonus if the similarity level exceeded 50%. The default threshold score was set to a stringent 18, which identified regions that are almost identical or better. For example, a perfect stretch of 18 bp will score 18 + (0.5*6) = 21. It is possible for regions to not be identical and still have potential as a primer as some sites may receive a score of less than one but the total score of the region still passes the threshold. Genes passed this screening process if they had potential forward and reverse priming sites that exceeded the threshold and were separated by at least 200 bp. The script is presented in Appendix 4.1.

Among the genes that passed the screening process, a single gene from each of the eight bins was randomly selected. Based on the recommended priming sites identified by the Perl
script and the multiple sequence alignment, primers were then designed for the eight selected genes and examined with IDT OligoAnalyzer v3.1 (Owczarzy et al. 2008) for potential problems before being synthesized (Table 4.1).

4.2.2 Fungal Cultures and DNA Extraction

One isolate from each of the four filamentous ascomycete classes (Dothideomycetes, Eurotiomycetes, Leotiomycetes, and Sordariomycetes) was obtained from the local fungal stock culture collection in the Hsiang Lab at the University of Guelph. None of the retrieved species were from the 49 sequenced species used for the comparative genomics and phylogenetics analyses in Chapters 2 and 3. Fungal isolates were grown on cellophane placed on 2% potato dextrose agar (PDA, Becton, Dickinson and Company, MD, USA) and incubated at room temperature. Mycelia (100 mg) were removed from the cellophane and stored at -20°C in a 1.5 mL Eppendorf tubes prior to DNA extraction. For genomic DNA extraction, Edwards’s method (Edwards et al. 1991) was used with some modifications (Huang et al. 2001). Mycelia at room temperature were manually disrupted using a mini pestle (Kontes, Vineland, New Jersey, USA) and DNA was subsequently extracted using the Qiagen DNeasy Plant Mini Kit (Qiagen Canada, Mississauga, ON) following the “Mini Protocol” for plant tissue (Qiagen Canada, 2006). Extracted DNA was stored at -20°C.

4.2.3 PCR Amplification and Sequencing

Each of the eight genes selected in the screening above was amplified from the genomic DNA of a single fungal isolate from the four classes examined (Table 4.3) in the study: Eurotiomycetes, Dothideomycetes, Sordariomycetes and Leotiomycetes. The PCRs were initially performed using a final volume of 15 μL to optimize the annealing temperature. Once the
optimal annealing temperature was identified, the reaction volume was doubled for sequencing purposes. Each 15 μL volume used for amplification contained 1x PCR buffer (50mM Tris-HCl, pH 8.5), 2.5 mM MgSO₄, 0.2 mM dNTP, 0.5 μM of each primer, 0.6 U Tsg DNA polymerase (Biobasic, Markham, ON), 1 μL of fungal DNA extracted using the Qiagen DNeasy Plant Mini Kit diluted by a factor of 10, and sterile distilled deionized water to complete the reaction volume. A MyCycler thermal cycler (No. 580BR 10624; BioRad, USA) was used for the amplification with an initial denaturation phase of 94ºC for 2 minutes followed by 35 cycles of 94ºC for 30 s, annealing temperature for 1 min, and elongation at 72ºC for 1 min and a final elongation phase at 72ºC for 10 minutes. Annealing temperatures were optimized based on the primer sets used. An initial annealing temperature of 55ºC was used in all reactions, and adjustments were made accordingly to reduce multiple bands or increase sensitivity (Table 4.1).

The results of the PCR were viewed on 1% ultra pure agarose gels (Invitrogen, Carlsbad, California, USA) made with 0.5X Tris-Borate-EDTA buffer (TBE) (90 mM Tris base, 90 mM boric acid and 2 mM EDTA). DNA samples were loaded with a 6X loading dye at a ratio of 3:1 respectively. The 6X loading dye contained 0.1% (w/v) bromophenol blue, 0.1% (w/v) xylene cyanol, 30% (v/v) glycerol, and 60 mM EDTA. In each gel, one lane was loaded with 5 μL of GeneRuler 100 bp DNA ladder mix (Fermentas, USA) mixed in a 1:6:1 ratio with sterile dH₂O and GeneRuler loading dye (manufacturer’s info) to allow for the determination of band size and concentration. Electrophoresis was performed at 100 V and 0.8 A in a Mupid 21 gel box (Helixx Technologies, Toronto, Ontario, Canada). Following electrophoresis, gels were stained in 2 - 4 μg/mL ethidium bromide for five minutes and then destained in tap water for five to ten minutes. A Syngene UV transilluminator (Synoptics, Cambridge, Cambridgeshire, U.K.) was used to visualize the DNA bands in the gel. Print-outs of all gels were made with a GBC CCTV camera.
(South Hackensack, New Jersey, USA) attached to a video copy processor P67U (Mitsubishi Electric, Cypress, California, USA).

All eight sets of PCR products and their respective primers were sent to BioBasic (Markham, ON, Canada) for DNA sequencing. Chromatograms were viewed in Sequencher 4.1.4 (GeneCodes, Ann Harbor, Michigan, USA) and base calls of sequences were manually trimmed and edited if necessary.

4.2.4 Phylogenetic Tree Construction

For each gene, the sequencing results for the four amplicons were aligned using the EINSI options in MAFFT v6 (Katoh et al. 2009), manually trimmed in BioEdit v7.0.5.3 (Hall 1999), followed by NJ tree construction in ClustalX v2.1 (Larkin et al. 2007). All trees were generated with 1,000 bootstrap iterations.

To determine if the designed primers might be useful for phylogenetics, trees were generated by combining the sequenced amplicons with the original alignment. The four amplicon sequences were appended to the original alignment, which was then trimmed to the length of the amplified region. Two trees were generated from each of the eight alignments. A NJ tree with 1000 bootstrap replicate was generated using ClustalX while an ML tree was generated from the alignment file after being converted into Phylip format with a Perl script (Appendix 3.4). RAxML v7.2.8-alpha was used to produce the ML tree with 1000 bootstrap replicates, and using a random seed of 123, and the GTR-Γ model.

A multigene tree was also generated. The eight alignments, which contained the original 49 aligned sequences and the amplicons, were randomly concatenated into a single superalignment using a Perl script (Appendix 3.4). The superalignment file was converted to Phylip format with a Perl script (Appendix 3.2) to allow for analysis with RAxML. The ML tree
was produced with RAxML (Stamatakis 2006) using a thorough ML search with 1,000 bootstrap replicates under the GTR+Γ model.

4.3 Results

4.3.1 Primer Screening Script

The gene screening Perl script (Appendix 4.1) examines all possible consecutive stretches of 18 bp in a multiple sequence alignment for potential priming sites. It first passes the alignment file through TrimAL v1.2 (Capella-Gutierrez et al. 2009) with the ‘-scc’ option which outputs the amount of similarity at each site as calculated using Euclidean distances. For each site, similarity values ranged between 0 for poorly conserved sites and 1 for perfectly conserved sites. The script calculated the total similarity of the 18 consecutive sites with greater weighting for high similarity in the last 6 bp at the 3’ end. The greater weight is calculated by adding an additional 50% bonus if the similarity level exceeded 50%. The default threshold score was set to a stringent 18, which identified regions that are almost identical or better. For example, a perfect stretch of 18 bp will score $18 + (0.5*6) = 21$. It is possible for regions to not be identical and still have potential as a primer as some sites may receive a score of less than one but the total score of the region still passes the threshold. Genes passed this screening process if they had potential forward and reverse priming sites that exceeded the threshold and are separated by at least 200 bp.

4.3.2 Gene Selection and Screening

Out of the 165 *N. crassa* genes used to generate the single-gene trees in Chapter 2, there were 92 genes with a mean identity between 50% and 90%. The four 10% bins between 50% and 90% were split into eight 5% bins for a more intensive examination. From each of the eight 5%
bins between 50% and 90%, five genes were randomly selected for screening by the script (Appendix 4.1). Out of the 40 genes, 21 passed the screen (Table 4.2). From each 5% mean identity bin, a single gene was then randomly selected from those that passed the primer screen for a total of 8 genes (Table 4.2).

4.3.3 Primer Test and Phylogenetic Analyses

The primers for all eight genes were designed and tested using PCR amplification on nine fungal isolates (Table 4.3), in which each of the four filamentous ascomycete classes examined throughout Chapter 2 and 3 were represented by at least one isolate. Each set of primers successfully amplified at least one representative in all four filamentous ascomycete classes, but only one from each class was submitted for sequencing. Thus, 32 amplicons in total (eight primer sets, four representatives) were sequenced and the bootstrapped trees were constructed (Figure 4.1). Out of the eight amplicon trees built on one sequenced representative from each of the four fungal classes, five were concordant at the fungal class level with the T1 topology (Figure 4.2). Two of the remaining three trees revealed the same topology as T3 and the last one showed the Leotiomycetes taxa grouping with the Dothideomycetes while the Eurotiomycetes grouped with the Sordariomycetes. When compared to the topologies of the single-gene trees generated in Chapter 2, the amplicon trees for six of the eight genes showed the same topology of class level relationships as their respective single-gene trees generated using all 49 sequences (Figure 4.2).

NJ and ML trees were also generated from alignments containing the both the sequenced amplicons and the truncated original alignment. In four of the eight NJ trees, at least one of the sequenced amplicons did not group with its respective class. Moreover, none of the NJ trees topologies matched the respective original single-gene tree from Chapter 2, nor their respective
amplicon trees. Out of the eight ML trees, only one gene yielded the T1 topology. The eight alignments were then concatenated into a single superalignment and used to generate a ML tree using RAxML. The resulting 8-gene multigene tree yielded the T1 topology.

4.4 Discussion

4.4.1 Range of Mean Identity Selected for Priming Region Screening

Based on Figure 2.8 (mean identity bin results for single genes calculated from 8 genomes), the upper limit of the mean identity bins to test was set to 90% because of the scarcity of genes beyond 90%. This high level of conservation may also prevent resolution of taxa as there are fewer phylogenetically informative sites. For the lower identity limit, the choices for the cutoff were 40% identity and 50% identity because the bins within this range had the fewest unresolved single-gene trees while the greatest number of fully resolved single-gene trees was between 45% and 50%. Genes that do not resolve at all may be suffering from other systematic or stochastic problems that cannot be solved by additional taxon sampling and, therefore, the lower limit was set to 50% to avoid the potentially problematic genes between 40% and 50%. By sampling between 50% and 90% instead of 40% and 90%, the genes being screened for primer design may have a greater chance of having potential priming sites, and more likely to yield a resolved tree. The selected range used here was similar to another study that mined for genes using a protein similarity range between 40% and 90% (Townsend et al. 2008b).

4.4.2 Mean Identity and Screening Process

During the screening process, genes with higher mean identity (lower sequence divergence) tended to pass the screen more often than genes with lower mean identity. Intuitively, this makes sense because highly conserved genes are, by chance, more likely to have
consecutive stretches of 18 conserved sites than poorly conserved genes. As seen in Table 4.2, only one gene in each five percent bin between 50% and 65% passed the screen whereas all the genes screened between 80% and 90% passed.

This highlights potential issues in designing ‘universal’ or ‘near-universal’ primers for use in phylogenetics at lower taxonomic levels. When examining phylogenies at high taxonomic levels (class, orders, etc), primers can be designed with relative ease for fairly conserved to highly conserved genes as these may be sufficient for resolving the tree. In contrast, genes useful for lower taxonomic levels may have higher sequence divergence, and thus, designing universal primers for those regions may be more difficult. One solution that has been explored is to design primers in conserved regions that flank the gene of interest. For example, the ITS region is a highly variable region that is frequently used for species level taxonomy and identification in fungi and in order to amplify it, ‘universal’ primers were designed in the more conserved flanking LSU and SSU regions (White T et al. 1990). Although the current study focused on class level relationships of filamentous ascomycetes, this observation shows that conserved gene order may be useful information when designing primers to amplify variable protein-coding genes that may be phylogenetically informative at low taxonomic levels (e.g.: genus and species).

4.4.3 Mean Identity and Ease of Primer Design

PCR amplification of the eight genes with a single representative from each of the four taxonomic classes completed successfully and the amplicons were sequenced. All eight pairs of primers were designed based on the priming sites suggested by the screening script. Thus, the screening process served two purposes: it filtered out the genes that may not be conserved across all these high taxonomic levels and it identified the sites for primer design.
Once the amplicons were sequenced, a NJ tree with 1,000 bootstrap replicates was constructed for each gene based on the amplified regions. The purpose of the PCR test was to determine if the primers can be used to amplify the eight genes in the four filamentous ascomycete classes. Because only one representative was amplified from each taxonomic class, the ability of the primers to yield amplicons that can produce a well-resolved tree was not examined. Instead, the relationships between the four representative taxa inferred by the topologies were examined and compared. More than half of the eight amplicon trees and both superalignment trees grouped in the same way as 18S and T1 (Figure 4.2).

Out of the eight randomly selected genes tested in this Chapter, three of the five trees that passed the screen and yielded T1 were generated from genes with mean identities between 50% and 65%. Because we only tested genes between 50% and 90% identity, these three genes had the lowest mean identity out of the eight. This reinforces the observation that genes with lower sequence divergence may be more phylogenetically informative for resolving classes; however, a lower level of conservation may make it difficult to identify priming sites that are conserved across taxonomic classes.

Based on these results, several potential refinements can be made regarding the amplification of phylogenetically informative genes in the genomes that have not been sequenced. The first is that focus should be placed on the genes with the greatest amount of variation within the range of mean identities without compromising the potential for primer design. In this case, class-level relationships may be properly resolved by the use of genes with mean identities between 50% and 65%. Based on the number of genes that passed the screen at different mean identity ranges in Table 4.2, using a less stringent cutoff in the Perl script may be required to find priming sites in genes with a mean identity lower than 50%. Additionally, when
calculating the mean identities, it may be useful to also retrieve the values for standard deviation. By considering the variance of the identity values, it may be possible to filter out genes that may be less phylogenetically informative.

Alternatively, it may also be possible to work with superalignments from multiple conserved genes. Since it has been demonstrated in Chapter 3 that multigene sets may more consistently yield the same topology, designing primers for several conserved genes and concatenating the sequences from the resulting amplicons into a superalignment may also yield a well-resolved tree. This may be easier than designing primers for less conserved but more phylogenetically informative genes. Moreover, this technique has been employed in recent fungal phylogenetic studies such as the six-gene phylogeny of the Ascomycetes (Schoch et al. 2009b), five-gene phylogeny of the Pezizomycotina (Spatafora et al. 2006), five-gene phylogeny of the Dothideomycetes (Schoch et al. 2009a), and the four-gene phylogeny of the Sordariomycetes (Zhang et al. 2006).

4.4.4 Comparisons of Trees Generated using Amplicons and Whole Genes

The trees generated using only the amplicons were compared to the trees generated using the full gene alignment with all 49 sequences (Chapter 2). Two of the eight genes did not have matching topologies (NC007341 and NC006868 in Figure 4.2) between the amplicon trees generated here and their respective full length gene tree from Chapter 2. The trees of these two genes did not show the T1 topology in Chapter 2 when the full length was used, but showed the T1 topology when only the amplicons were used. Both single-gene trees from Chapter 2 had one class nested within another, but because only one representative was used to test the primers, it was not possible to obtain trees with nested clades, which may explain the different topology.
The topologies of the three amplicon trees that did not match T1 also did not match T1 in Chapter 2 when the full alignment length was used (Figure 4.2). This may suggest that the three genes (NC001938, NC008393, and NC006543) are not phylogenetically useful for the taxa studied, whether full length or primer-limited amplicon length. When the amplicon sequences for all eight genes were added to their respective original 49 sequences, the topologies of the resulting NJ trees were different from both the amplicon trees and the original 49 sequence tree and the species were not resolved into the four classes as in T1. Only one of eight ML trees showed the T1 topology. Interestingly, despite the absence of the T1 topology in the eight NJ trees and seven of the ML trees, the 8-gene ML tree yielded the T1 topology. This reinforces the usage of multigene trees over single-gene trees.

There may be some potential sources of signals and noises that caused the differences between the trees in Chapter 2 and in this Chapter but it may not be possible to separate the effects out without further testing and a larger dataset. One potential source may have been the usage of the amplicon region versus the full-length coding region. Because the amplified region is shorter, there will be a difference in the amount of signal and noise used to generate the tree. Moreover, taxa sampling may also play an issue as the amplicon tree was generated using only four sequences (one from each class) and it has been demonstrated that sparse taxa sampling may lead to low phylogenetic accuracy and high error (Zwickl and Hillis 2002).

Another problem to consider is the extent of phylogenetic signal available in the amplified region. Full length single gene trees may give resolution that partial sequences cannot. A whole gene with over 3,000 bp may be able to yield a well-resolved tree but a short stretch of 750bp may not be as useful. Although a single amplicon region may be insufficient for resolving a tree with strong bootstrap support, combining the short amplified regions from several genes
may be sufficient for generating a topology that may resemble the true tree. This corroborates the multigene methods employed in many fungal systematic studies.

Since only eight genes were examined by primer selection, there may be some potential implications that can be explored in future studies. This test showed that designing primers to amplify a representative region of the gene may not be difficult, but gene selection may strongly affect results as some may be more phylogenetically informative and reflective of the true species tree than others.

### 4.4.5 Comparison of Primer Development Method to Recent Studies

A recent study outlined a method based on relative amounts of transitions (Ti) and tranversions (Tv) to select potential phylogenetically useful genes from the genome data of FunyBASE (Feau et al. 2011). Using a sliding window scripted in Python, mutational hotspots in single-copy protein-coding genes flanked by potential priming regions were identified as phylogenetic markers in a sequence alignment (Feau et al. 2011). This contrasted with the methods used throughout this study as sequence divergence, rather than ratios of Ti and Tv, was used to estimate phylogenetic potential. In addition, we used a sliding window scripted in Perl to screen out genes that might be too difficult to amplify, and to identify the potential priming sites from an alignment containing sequences from 49 genomes. Only one pair of primers was designed for each of the eight genes and all eight sets successfully amplified a single fragment of the expected size in a representative from each of the four taxonomic classes examined. Furthermore, the primers were designed based on highly conserved regions across the 49 sequences in the alignment and amplification was not attempted beyond the four classes.
4.5 Conclusions

The results from Chapter 2 demonstrated that about 40% of the genes in *Neurospora crassa* genome could yield a tree that matched the currently accepted topology of the filamentous ascomycetes at the class level. However, the results were obtained using genome sequence data, which may not be available for all fungal species. When additional taxa sampling is desired from genomes that have not been sequenced, gene data will have to be obtained through targeted amplification using PCR.

In this Chapter, a Perl script was developed to screen multiple sequence alignments to determine if they have conserved priming sites that can be used to amplify from different species. Rather than examining all the genes in the genome, we focused on a subset of the 165 genes used to generate the single-gene trees in Chapter 2. The range of mean identity with the most genes (67%) yielding the T1 topology was determined in Chapter 2 to be between 45-60%. Genes with mean identity values below 50% were not used as many did not resolve at all and instead, we focused on the genes between 50-90%. Genes from this range were screened and eight were selected for primer design based on the recommended priming sites outputted by the scripts. All eight sets of primers successfully amplified in at least one representative from each of the four filamentous ascomycete classes demonstrating the effectiveness of the script in identifying conserved priming regions.

Out of the eight genes, five of the amplicon trees yielded the T1 topology (62%), and three of those five had a mean identity between 50-65%. This corroborated the results in Chapter 2, which showed that genes with mean identities between 45-60% have a high chance of yielding a tree of the filamentous ascomycetes that also matches the currently accepted topology.
However, none of the genes, when using only the amplified regions, led to a strongly supported and well-resolved tree that match the reference T1.

Future adaptations of the Perl script can be made to not rely on an external program to identify potential priming regions as this would increase user-friendliness. In addition, the TrimAl program that was called in the script converts the sequence information into a number to represent the level of similarity of the site across the sequence alignment. Rather than just working with numerical values, an improvement would be to modify the script such that it can handle the nucleotide information to estimate the GC content or give bonuses for priming regions that end with a GC clamp as these may improve the yield of the PCR reaction. Additional methods that can determine how phylogenetically informative and accurate the amplified regions are may also increase the efficiencies of primer development. Furthermore, depending on the taxonomic level being examined and the relationships of the taxa being examined, the stringency of the methods can be altered such that the identified priming sites are more specific or more sensitive.
Table 4.1. Forward and reverse oligonucleotide primers designed for the eight genes using the BuildOligos.pl script. Fragment lengths were calculated relative to the *Neurospora crassa* sequence data. Primers were initially tested with an annealing temperature of 55ºC and based on results of the gel electrophoresis; the annealing temperature was adjusted accordingly. The indicated annealing temperatures are the final optimized annealing temperatures used for the PCR.

<table>
<thead>
<tr>
<th>Gene ID</th>
<th>Primers</th>
<th>Primer Sequence</th>
<th>Fragment Length in <em>N. crassa</em> (bp)</th>
<th>Annealing Temperature (ºC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NC001938</td>
<td>NC001938_156F</td>
<td>5<code> ATCTTCGGCTGGGATATGCA 3</code></td>
<td>867</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>NC001938_1023R</td>
<td>5<code> GTGACTGGAGGCCCAGAAGTT 3</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NC003210</td>
<td>NC003210_260F</td>
<td>5<code> TACTTCAACCACGAGAACAT 3</code></td>
<td>356</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>NC003210_616R</td>
<td>5<code> GCCCTGGTGACTGCTTGAA 3</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NC004139</td>
<td>NC004139_962F</td>
<td>5<code> ATGTGGTACAGTTGGATTGG 3</code></td>
<td>696</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>NC004139_1658R</td>
<td>5<code> CCAGAAGATGACGACATGAAAC 3</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NC004477</td>
<td>NC004477_446F</td>
<td>5<code> CTTGCCGCTATCATGTTGTC 3</code></td>
<td>623</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>NC004477_1069R</td>
<td>5<code> TTAGGTAGAGGGTGGACACC 3</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NC006543</td>
<td>NC006543_368F</td>
<td>5<code> CTCGTTTCCCTCATGATGTT 3</code></td>
<td>626</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>NC006543_994R</td>
<td>5<code> GAGCCAGGCTTCATCTTCTC 3</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NC006868</td>
<td>NC006868_98F</td>
<td>5<code> CGGCAGATGATTGGCGTTTGG 3</code></td>
<td>904</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td>NC006868_1002R</td>
<td>5<code> TGGCCGGCAGCTACATGAA 3</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NC007341</td>
<td>NC007341_176F</td>
<td>5<code> ATGGGCAAGAAACATGTTGTT 3</code></td>
<td>614</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>NC007341_790R</td>
<td>5<code> GGTGGCAAATGCGCTTCTT 3</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NC008393</td>
<td>NC008393_920F</td>
<td>5<code> CAACTTCAATTGTATGTTG 3</code></td>
<td>906</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>NC008393_1826R</td>
<td>5<code> TGTTGGAAGGCATCTTCTC 3</code></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 4.2. Out of the original 165 genes used to determine a measure of phylogenetic potential in Chapter 2, five genes from each of the five percent bins between 50% and 90% were randomly selected to be screened by a primer-development script. Of these 40 genes, 21 genes passed the threshold requirements of the screening script BuildOligos.pl, and a single gene from each bin was randomly selected for the actual primer development.

<table>
<thead>
<tr>
<th>Identity Bins</th>
<th>Genes</th>
<th>Genes Screened</th>
<th>Passed Screen</th>
<th>Tested</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-5</td>
<td>0</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>5-10</td>
<td>0</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>10-15</td>
<td>5</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>15-20</td>
<td>5</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>20-25</td>
<td>7</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>25-30</td>
<td>8</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>30-35</td>
<td>11</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>35-40</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>40-45</td>
<td>12</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>45-50</td>
<td>11</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>50-55</td>
<td>10</td>
<td>5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>55-60</td>
<td>11</td>
<td>5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>60-65</td>
<td>12</td>
<td>5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>65-70</td>
<td>14</td>
<td>5</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>70-75</td>
<td>7</td>
<td>5</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>75-80</td>
<td>7</td>
<td>5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>80-85</td>
<td>23</td>
<td>5</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>85-90</td>
<td>8</td>
<td>5</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>90-95</td>
<td>3</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>95-100</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>165</td>
<td>40</td>
<td>21</td>
<td>8</td>
</tr>
</tbody>
</table>
Table 4.3. Eight sets of oligonucleotides were tested by performing PCR on each of the class representatives in the table. If the gene amplification with the primer pairs was successful, one amplicon from each of the four classes was submitted for sequencing.

<table>
<thead>
<tr>
<th>Species</th>
<th>Isolate</th>
<th>Class</th>
<th>Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diplodia sapinea</td>
<td>SP16a</td>
<td>Dothideomycetes</td>
<td>Botryosphaeriales</td>
</tr>
<tr>
<td>Sydowia polyspora</td>
<td>11145</td>
<td>Dothideomycetes</td>
<td>Dothideales</td>
</tr>
<tr>
<td>Gaumannomyces graminis</td>
<td>GG06</td>
<td>Sordariomycetes</td>
<td>Magnaporthales</td>
</tr>
<tr>
<td>Fusarium foetans</td>
<td>F2</td>
<td>Sordariomycetes</td>
<td>Hypocreales</td>
</tr>
<tr>
<td>Penicillium sp.</td>
<td>00005</td>
<td>Eurotiomycetes</td>
<td>Eurotiales</td>
</tr>
<tr>
<td>Sclerotinia homeocarpa</td>
<td>SH84</td>
<td>Leotiomycetes</td>
<td>Helotiales</td>
</tr>
<tr>
<td>Chalara sp.</td>
<td>07070</td>
<td>Leotiomycetes</td>
<td>Helotiales</td>
</tr>
<tr>
<td>Botrytis pseudocinerea</td>
<td>10091</td>
<td>Leotiomycetes</td>
<td>Helotiales</td>
</tr>
<tr>
<td>Monilinia fructicola</td>
<td>11181</td>
<td>Leotiomycetes</td>
<td>Helotiales</td>
</tr>
</tbody>
</table>
Figure 4.1. Three topologies were identified from the eight Neighbor-Joining trees that were generated from the alignments of the amplicons using ClustalX v2.1 with 1000 bootstrap replicates. For each gene, the four amplicons were aligned using EINSI in MAFFT v6 and manually trimmed in BioEdit 7.0.5.3. A) Five of the eight trees showed the reference T1 topology as determined in Chapter 2. This topology occurred the most frequently out of the single-gene NJ trees (67/165) and was used as the reference topology. B) Two of the eight trees showed the T3 topology. This topology was the third most common topology out of the 165 single-gene NJ trees. It grouped Eurotiomycetes and Leotiomycetes together and Dothideomycetes and Sordariomycetes together. C) One of the eight NJ trees generated showed this topology, which occurred only eight times out of the 165 single-gene NJ trees. In this topology, Leotiomycetes is grouped with Dothideomycetes while Eurotiomycetes and Sordariomycetes are grouped together.
Figure 4.2. The resulting topologies of the eight genes from the *Neurospora crassa* genome that were selected for primer design based on the primer screening process and their single-gene tree topology as determined in Chapter 2. The single-gene trees from Chapter 2 were generated using 49 sequences, while the amplicon trees were generated using only four sequences. The list has been sorted according to ascending mean identity. Out of the eight amplicon trees, five yielded evolutionary relationships that matched T1. The four classes are represented by the first letter of their names (E = Eurotiomycetes; D = Dothideomycetes; S = Sordariomycetes; and L = Leotiomycetes).

<table>
<thead>
<tr>
<th>Gene</th>
<th>Mean Identity</th>
<th>Single-Gene Tree Topology (Chapter 2)</th>
<th>Amplicon Tree Topology</th>
</tr>
</thead>
<tbody>
<tr>
<td>NC004477</td>
<td>52.75</td>
<td>T1</td>
<td>T1</td>
</tr>
<tr>
<td>NC004139</td>
<td>55.06</td>
<td>T1</td>
<td>T1</td>
</tr>
<tr>
<td>NC007341</td>
<td>63.54</td>
<td>T2</td>
<td>T1</td>
</tr>
<tr>
<td>NC001938</td>
<td>66.15</td>
<td>T3</td>
<td>T3</td>
</tr>
<tr>
<td>NC006868</td>
<td>74.65</td>
<td>D1</td>
<td>T1</td>
</tr>
<tr>
<td>NC008393</td>
<td>77.51</td>
<td>D2</td>
<td>T1</td>
</tr>
<tr>
<td>NC003210</td>
<td>81.44</td>
<td>T1</td>
<td>T1</td>
</tr>
<tr>
<td>NC006543</td>
<td>85.40</td>
<td>T3</td>
<td>T3</td>
</tr>
</tbody>
</table>
Chapter 5: General Discussion

Using sequence genome data from eight yeast species, Rokas et al. (2003) recommended a minimum of 20 genes in a multigene dataset to generate a well-resolved tree with high bootstrap support values. However, much criticism has been made of the methodologies used by Rokas et al. (2003). For example, high bootstrap scores did not necessarily mean a tree was correct (Hedtke et al. 2006), long branch attraction was found in the basal nodes of some of their trees (Gatesy et al. 2007; Hedtke et al. 2006), and there may have been some compositional bias in the dataset (Phillips et al. 2004). Similar to the Rokas et al. (2003) study, we wanted to use the power of the increasing amounts of genomic data and address how to use portions of these data for phylogenetic purposes. Some of the issues they encountered might also be avoided by the use of a larger number of genomes, which have been published since their study, and as well, other analytic approaches could be tested.

In Chapter 2, a method was outlined to mine the genomes of sequenced filamentous ascomycetes for phylogenetically informative genes. This was done by first generating trees for a systematic selection of genes and then identifying the concordant topologies shared by groups of single gene trees at the class level. The largest single concordant class-level topology was called T1 and was shared by 67 out of 165 single gene trees. Using T1 as a reference, we assessed the resolution of all remaining single-gene trees. The single-gene tree results showed that a plurality of genes tested (40%) yielded a well-resolved tree, suggesting that it is possible for many single-gene trees to accurately produce well-resolved trees with high bootstrap values. Moreover, there appeared to be a relationship between sequence divergence within alignments and the ability of a gene to yield a well-resolved tree that matches T1. We originally calculated mean identity using a single query genome and used it as a representation of sequence divergence, but some values
did not appear to be correlated with the overall amount of conservation observed in sequence alignments. Consequently, mean identities were recalculated using two representative genomes from each of the four classes for a total of eight genomes. We found that the odds of obtaining a tree with the T1 topology could be improved to 64% (30/47) from the overall 40% by focusing on a narrower eight-genome mean identity range between 45% and 60%.

In Chapter 3, multigene trees involving various numbers of genes were compared against the results of the single-gene trees to assess the point at which a random selection of homologous genes would arrive at a topology concordant with the plurality of single-gene tree topology T1. Maximum likelihood (ML) superalignments trees were generated using sets of genes randomly sampled from a pool of homologous genes. In contrast with single-gene trees, a majority of all superalignments trees (81%) and all of the superalignment trees generated using more than five genes were well-resolved, had high bootstrap support values and matched the reference tree. Thus, we concluded that multigene trees outperform single-gene trees in terms of resolving the relationships of the filamentous ascomycetes at the class level when at least five homologous genes are used.

When working with superalignments and statistical methods of tree inferences, computational constraints must be considered. For example, ten 100-gene superalignment ML trees were generated from a pool of homologous genes to determine if they would yield unusual tree topologies or match with known topologies. The first of the ten trees completed in 44 hours on 20 cores (Intel Xeon L5420 quad-core processors, running at 2.5GHz; all nodes connected using Infiniband) and 15 GB of memory while the last one required 61 hours and 20 GB of memory. If a single core had been used, it would have taken 36 to 50 days for the trees to complete. Given that the ten 100-gene superalignment ML trees all showed the same topology as
T1, like the majority of the superalignment trees generated from 5 to 15 genes, it is probably not necessary to use that many genes for multigene trees.

Although no difference in topology was found between a tree generated with 5, 10, 15, or 100 genes, this may apply only to the particular data used, this set of taxa (filamentous ascomycetes) studied, or the targeted class-level taxonomy. Because the study did not encompass the Basidiomycota and the various classes in that phylum, the recommendation for the number of genes cannot be firmly asserted for those taxa. Perhaps as more Basidiomycete genomes are sequenced, further studies involving multiple taxonomic levels and groups can be considered.

Despite the increasing number of genomes being made available to the public on a regular basis, some phylogenetic studies may require data from taxa that have not been sequenced or are too difficult to sequence. In order to access these sequences, genes of interest will have to be amplified using traditional PCR methods and primer synthesis. However, not all genes are equally suited for this task. As confirmed in Chapter 4, the level of protein sequence divergence can be used to predict if a single gene would be able to yield a well-resolved tree with robust statistical support. In addition, the level of sequence divergence also influences the possibility of designing pairs of primers for amplifying the gene across multiple taxa. As sequence divergence increases, there are fewer stretches of conserved sites in the gene, and this can confound attempts to design primers. To address this issue, a script was used to rapidly score and identify well-conserved regions in gene alignments that can be used for primer design. By filtering out genes with low-scoring regions and which do not have at least 200 bp between potential priming sites, genes that are potentially phylogenetically informative and might be amplifiable using PCR were identified. Out of the 40 genes tested, 21 passed the screen, and primers were designed for eight. After optimizing the annealing temperature of a ‘standard’ PCR
protocol, a single band was successfully amplified in at least one taxon from each of the four filamentous ascomycete classes for all eight primer pairs. The primers generated using this method worked across different classes because they targeted highly conserved regions. This is more economical and efficient as it limits the number of primers that have to be designed in order to study a region of interest.

Even if the amplicon sizes are too small and lack sufficient variation for a well-resolved single gene tree, the results from several genes can be combined to yield a superalignment tree. As demonstrated in Chapter 4, amplicon trees were generated for each of the eight genes, and three did not produce the expected T1. Yet, when the amplicons for all eight genes were combined into a single superalignment, both the resultant NJ and ML trees showed T1 with bootstrap values of at least 90% at all branches. This may indicate that the stochastic noise in the dataset has been drowned out by the phylogenetic signals.

Despite examining only eight genes using this method, the three genes with the lowest mean identity (50-65%) yielded amplicon trees that matched T1. This supports the results from Chapter 2, which showed that genes with mean identity between 45% and 65% have the greatest chance of yielding trees with a T1 topology.

Several genes currently used in fungal phylogenetics were identified in the genome of Neurospora crassa genes to determine their level of sequence divergence. Translation elongation factor one alpha (EF1-α), RNA polymerase II second largest subunit (RPB2), and beta tubulin (β-tub) were found to have mean identities of 83.7, 71.9, and 85.4% respectively. Compared to the rest of the genes in the N. crassa genome, these three genes are among the most highly conserved fungal genes ranking in the top five percentile. Chapter 2 demonstrated that the potential of a gene to yield a tree that is concordant with the accepted tree of the filamentous
ascomycetes at the class level is related to its level of sequence divergence, which supports previously published studies (Graybeal 1994; Rokas et al. 2002). This suggests that these three commonly used genes may be too conserved for single-gene phylogenetic analyses at the genus or species levels. However, Chapter 3 showed that combining genes in sufficient numbers will yield a tree that resembles the currently accepted topology, even if the genes are well conserved. Indeed, many recently published studies do use these conserved genes in multigene phylogenetic analyses at various taxonomic levels including phylum (Schoch et al. 2009b), subphylum (Spatafora et al. 2006), class (Schoch et al. 2009a; Zhang et al. 2006), and species (Couch and Kohn 2002; Zhang et al. 2011).

The direct implication of this study is the confirmation of the currently accepted evolutionary relationships of four classes of the filamentous ascomycetes. Many of the phylogenetic trees generated throughout this study yielded a single topology (T1) that matched a tree generated using 18S and an unrooted version of the currently accepted topology of the filamentous ascomycetes (Schoch et al. 2009b; Spatafora et al. 2006). Finally, when considering a superalignment tree for elucidating the relationships in the Pezizomycotina, a minimum of 5 genes is recommended to generate a well-resolved superalignment tree with highly supported nodes. If sequencing cost, computational power and genome availability are not major issues, then having more genes may improve support values, but concatenating 100 genes may be a waste of computational resources and redundant.

In the future, as molecular techniques improve even more and the price of sequencing and computing falls, complete genome sequences for other fungi will accumulate. With the growing pool of genomes, a more thorough examination of fungal systematics can be made, and discrepancies between conflicting trees can hopefully be eliminated. By including other clades
such as the yeasts, Basidiomycetes, and Zygomycetes, it might be possible to design fungal-
specific primers for applications in medicine and barcoding. Perhaps with enough sequenced
fungal genomes, a way to the fungal tree of life can be determined. Furthermore, complete
genome sequences will most likely accumulate for other organisms such as plants, animals, and
insects. These genomes can be thoroughly combed to identify the groups of genes that can be
targeted to obtain well-resolved trees at different taxonomic levels.
References


115


Stamatakis A, 2008. RAxML manual version 7.0.4. Ecole Polytechnique Federale de Lausanne: School of Computer and Communication Sciences, Laboratory for Computation Biology and Bioinformatics, Distributed by the author.


Takahashi K, Nei M, 2000. Efficiencies of fast algorithms of phylogenetic inference under the criteria of maximum parsimony, minimum evolution, and maximum likelihood when a large number of sequences are used. Molecular Biology and Evolution 17, 1251-1258.


Townsend TM, Alegre RE, Kelley ST, Wiens JJ, Reeder TW, 2008b. Rapid development of multiple nuclear loci for phylogenetic analysis using genomic resources: An example from squamate reptiles. *Molecular Phylogenetics and Evolution* 47, 129-142.


Appendices

Appendix 2.1. This bash script was used on Sharcnet to generate the file used by mpiBLAST to find the location of the database files and also generate the mpiBLAST databases in the folder. mpiBLAST database files were split into 30 pieces and allows BLAST to execute on 30 cores. Fasta files used to generate the databases were stored in /work/$USER/hound/mpiBLAST/database. This script requires an installation of mpiBLAST. ................................................................. 127

Appendix 2.2. This is an example BLAST output from MPIBLAST. ........................................ 128

Appendix 2.3. This Perl script was used to parse text formatted BLAST output files into BLAST statistics of interest in tab-delimited format. ................................................................. 132

Appendix 2.4. This Perl script was used to condense all the parsed BLAST files into a single table to make viewing easier. Downstream calculations such as means and geometric means were performed using the output of this script. ................................................................. 134

Appendix 2.5. This script was used in part to compile the mean identity across several genomes for Neurospora crassa. ................................................................. 136

Appendix 3.1. Top 25 genes from Neurospora crassa (version 4 from Ensemble) with the highest 8-genome mean identity in the filamentous ascomycetes. Mean identities were calculated using 8 genomes as compared to 48 other filamentous ascomycete genomes using tBLASTn. Identity values are relative to the length of the query sequence. ........................................ 137

Appendix 3.2. This Perl script was used to concatenate the alignments of genes into superalignments containing 5, 10, 15, 20, and 25 genes. ................................................................. 138

Appendix 3.3. This Perl script was found online and adapted to convert all Fasta files with a specific extension in the folder into a relaxed Phylip format (no ten character limit for names). ................................................................. 140

Appendix 3.4. This Perl script was used to concatenate the alignments of genes into 100 gene superalignments. ................................................................. 141

Appendix 3.5. This Perl script was used to concatenate all alignments in a directory into a single superalignment. ................................................................. 143

Appendix 3.6. This script was used to randomly select genes for the control superalignment trees. The script reads a list of gene, and randomly selects a specified number of genes without replacement. The number of independent replicates can also be specified. ................................ 145

Appendix 3.7. An example of a multiple sequence alignment (A) before being trimmed by TrimAl v1.2 with a gap threshold of 0.75 and (B) after. ................................................................. 146
Appendix 3.8. This script was used to randomly select a set of genes and outputs the mean of mean identities. The script reads a list of gene, and randomly selects a specified number of genes without replacement. The mean of mean identities of the selected genes is then calculated and outputted. The number of independent replicates can also be specified. ................................. 149

Appendix 3.9. Computational Constraints Encountered in Chapter 3 ........................................... 151

Appendix 4.1. This sample Perl script was used to identify potential priming sites in multiple sequence alignments. This script requires the TrimAL program. ...................................................... 153

Appendix 4.2. Forward and reverse priming regions for the \textit{N. crassa} gene NC001938 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below. ................................................................................................................................. 158

Appendix 4.3. Forward and reverse priming regions for the \textit{N. crassa} gene NC003210 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.................................................................................................................................. 159

Appendix 4.4. Forward and reverse priming regions for the \textit{N. crassa} gene NC004139 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.......................................................................................................................................... 160

Appendix 4.5. Forward and reverse priming regions for the \textit{N. crassa} gene NC004477 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below........................................................................................................................................ 161

Appendix 4.6. Forward and reverse priming regions for the \textit{N. crassa} gene NC006543 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.......................................................................................................................................... 162

Appendix 4.7. Forward and reverse priming regions for the \textit{N. crassa} gene NC006868 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below........................................................................................................................................ 163

Appendix 4.8. Forward and reverse priming regions for the \textit{N. crassa} gene NC007341 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.......................................................................................................................................... 164

Appendix 4.9. Forward and reverse priming regions for the \textit{N. crassa} gene NC008393 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.......................................................................................................................................... 165
Appendix 2.1

This bash script was used on Sharcnet to generate the file used by mpiBLAST to find the location of the database files and also generate the mpiBLAST databases in the folder. mpiBLAST database files were split into 30 pieces and allows BLAST to execute on 30 cores. Fasta files used to generate the databases were stored in /work/$USER/hound/mpiBLAST/database. This script requires an installation of mpiBLAST.

#!/bin/bash
# filename: dbformat
#
# Generates a file with a list of database names from /work/$USER/hound/mpiBLAST/database
# Creates folder in work to hold generated databases
# Generates .ncbirc and points to database folder
# Initiates mpiformatdb.sh and reads each line from the list file into it

# generates list of database names
ls /work/$USER/hound/mpiBLAST/database > /work/$USER/hound/mpiBLAST/dbnames

# checks if folder exists, if not, create a folder in scratch to hold databases
test -d /work/$USER/hound/mpiBLASTDB || mkdir /scratch/$USER/hound/mpiBLASTDB
createncbirc()
{
    # function that creates a hidden file in mpiblast directory to locate compiled dbs in scratch
    echo "[NCBI]" > /work/$USER/hound/mpiBLAST/.ncbirc
    echo "Data=/opt/sharcnet/blast/current/data" >> /work/$USER/hound/mpiBLAST/.ncbirc
    echo "[BLAST]" >> /work/$USER/hound/mpiBLAST/.ncbirc
    echo "BLASTDB=/work/$USER/hound/blastdb" >> /work/$USER/hound/mpiBLAST/.ncbirc
    echo "BLASTMAT=/opt/sharcnet/blast/current/data" >> /work/$USER/hound/mpiBLAST/.ncbirc
    echo "[mpiBLAST]" >> /work/$USER/hound/mpiBLASTDB" >> /work/$USER/hound/mpiBLAST/.ncbirc

    /work/$USER/hound/mpiBLASTDB
    echo "BLASTMAT="/opt/sharcnet/blast/current/data" >>
    /work/$USER/hound/mpiBLASTDB

    /work/$USER/hound/mpiBLASTDB
    echo "[mpiBLAST]" >> /work/$USER/hound/mpiBLASTDB" >> /work/$USER/hound/mpiBLASTDB
    echo "Shared="/work/$USER/mpiBLASTDB" >> /work/$USER/hound/mpiBLASTDB" >> /work/$USER/hound/mpiBLASTDB
    echo "Local="/tmp" >> /work/$USER/hound/mpiBLASTDB
}

# check to see if ".ncbirc" exists, if not, create it
test -e /work/$USER/hound/mpiBLAST/.ncbirc || createncbirc

# format MPI databases
exec</work/$USER/hound/mpiBLAST/dbnames
while read line ; do
    mpiformatdb.sh "-N 30 -i /work/$USER/mpiBLAST/database/$line -o T -p F"
done
### Appendix 2.2

This is an example BLAST output from MPiBLAST.

```
TBLASTN 2.2.15 [Oct-15-2006]

Reference:
Aaron E. Darling, Lucas Carey, and Wu-chun Feng,
"The design, implementation, and evaluation of mpiBLAST".
In Proceedings of 4th International Conference on Linux Clusters: The HPC Revolution 2003,
June 24-26 2003, San Jose, CA

Heshan Lin, Xiaosong Ma, Praveen Chandramohan, Al Geist, and Nagiza Samatova,
"Efficient Data Access for Parallel BLAST".
In Proceedings of 19th International Parallel & Distributed Processing Symposium 2005,
April 3-8 2005, Denver, CO

Query:  gnl|ncrassa|NC000004 pep:known
supercontig:EF1:7.1:970110:972273=-1 gene:EFNCRG00000000004
transcript:EFNCRRT00000000004
(568 letters)
Database: /scratch/huangchv/mpiblastDB/bcinerea.cds
16,448 sequences; 20,217,636 total letters

<table>
<thead>
<tr>
<th>Score</th>
<th>E</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>gnl</td>
<td>bcinerea</td>
<td>bc11620 CTP synthase (2077 nt)</td>
</tr>
<tr>
<td>gnl</td>
<td>bcinerea</td>
<td>bc11238 imidazole glycerol phosphate synthase hisHF...</td>
</tr>
<tr>
<td>gnl</td>
<td>bcinerea</td>
<td>bc12117 choline dehydrogenase (2091 nt)</td>
</tr>
</tbody>
</table>

Query:  >gnl|bcinerea|bc11620 CTP synthase (2077 nt)
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2
```

Query:  IACRCEKPLEQQTINKVASSCQVEVNQLAVROMPTIQVQVXXXXXxxxxxxxxxxTIKLLD
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  IACRC+ PLE+ TI K+A  CQVE  QV+ V ++ + Y VP
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  RCGRKLNLIWVDSEHLEEKERTKEDPTKYHAKWDVCVKGLFGFGFHRGTEGMRARQ
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  RCG+KL LIW++EHLE+ + PT++HKAWH+VC A GILVFSGFG RGTEGMI AA+
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  RCGKKKLINLIVEAEHLEDDAK---PTEFHKAWHEVCTADGILVFSGFGDRGTGEMIAAAK
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  RCRRNKTPFGLGVCGLMGQAVIAARNLCELIKDATSEEDANAEHRVIIFMPEGSKEKLLG
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  WAREQKTPFGLGVCGLMGQAVIAARNLCELIKDATSEEDANAEHRVIIFMPEGSKEKLLG
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  TMLRLTTRSTSFPGGGEFSEKSRLGMYATTEIEERKHPFREHFYEVNFDPYIEKLEQSGLIFIGKDD
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  TMRLRTPTEPGGGSWEKLRLKIGKIDEVHEKVRHKRYENFPEYVDRLOGDGLEFVGKDD
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  SGERMEVVEIKDHPYYVGVQYHPEYTSDRLPSRPFGLGFVAAA1GCLDQITKEILQDAGF
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  SGERME+VEIKDHPY+YVGQ+HPEY SRVL PS+ +LGFVAAA GCL++IT++ L+
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

Query:  ANGSING 565
+ S+NG
Length = 2077
Score = 447 (1149), Expect(3) = 0.0
Identities = 215/367 (58%), Positives = 272/367 (74%), Gaps = 2/367 (0%) Frame = +2

```
128
Sbjct: 2021 LDNSVNG 2041

Score = 240 bits (613), Expect(3) = 0.0
Identities = 114/158 (72%), Positives = 131/158 (82%)
Frame = +3

Query: 44 HGE CFVLH GDGETDL DGLN ERYLVGD LARDNIT TGK VQVI ERK GKVLYLG RT VQVV
Sbjct: 435 HGEVFVL DGG EDV DGLN ERY LNI TQV T VQVV 614

Query: 104 PHVIDAII DTVRSVPF DVGES EPC III E G E SQV TQV THRAGK N
Sbjct: 615 PH LTDA IQW IHR I A P DTV TQ V TQV THRAGK N 794

Query: 164 FINIH VSYP VNV G E QKTPQVHKV S A L I P D L I 201
F+ IHVS VPV+ GEQTKPQ A++ V AGL+PDL+ 795
Sbjct: 795 F LQIHVSLVPV QGEQ KTPQQ A I RDV G A L V PD L V 908

Sbjct: 320 SIKIDPY VMVAGTMNPE 376

Score = 38.1 bits (87), Expect(3) = 0.0
Identities = 16/19 (84%), Positives = 18/19 (94%)
Frame = +2

Query: 25 AIKIDPY I V D TMNPE 43
Sbjct: 320 SIKIDPY VMVAGTMNPE 376

Score = 29.6 bits (65), Expect = 3.2
Identities = 16/46 (34%), Positives = 24/46 (52%), Gaps = 4/46 (8%)
Frame = +2

Query: 363 GGFCHR GTE----GMIR AA QWAREQ K T F P F G V CLGMQVAVIE AA RN 404
G FGH T+ G + A Q + PF+G+C+G+Q + A N 233
Sbjct: 233 GHFOHCM TQFSK AGYLPAIQ KHIDSGKPFMGICVGLQALFEGS AEN 370

Sbjct: 12117 choline dehydrogenase (2091 nt)
Length = 2091

Score = 28.5 bits (62), Expect = 7.0
Identities = 18/50 (36%), Positives = 25/50 (50%)
Frame = +1

Query: 171 YVPVVNGEQKTPQHAKV S A L I P D L I ACRECKPLEQGTINKVASS 220
Y P+ N + PT++ ++ LI AC EKPL +GTI SS 1537
Sbjct: 1537 YFPL*NSKLQLPTRY----IKRH ALI TAPGACAV ERPLSRTG T ITLS PSS 1674

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds
Posted date: Sep 20, 2010 11:41 AM
Number of letters in database: 673,883
Number of sequences in database: 546

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.001
Posted date: Sep 20, 2010 11:41 AM
Number of letters in database: 673,971
Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.002
Posted date: Sep 20, 2010 11:41 AM
Number of letters in database: 673,883
Number of sequences in database: 547

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.003

129
Number of letters in database: 673,883
Number of sequences in database: 547

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.004

Number of letters in database: 673,883
Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.005

Number of letters in database: 673,972
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.006

Number of letters in database: 673,971
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.007

Number of letters in database: 673,971
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.008

Number of letters in database: 673,972
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.009

Number of letters in database: 673,972
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.010

Number of letters in database: 673,972
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.011

Number of letters in database: 673,972
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.012

Number of letters in database: 673,972
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.013

Number of letters in database: 673,971
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.014

Number of letters in database: 673,883
Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.015

Number of letters in database: 673,971
Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.016

Number of letters in database: 673,883
Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.017

Number of letters in database: 673,883
Number of sequences in database: 548
Number of letters in database: 673,883
Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.018
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,882
   Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.019
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,971
   Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.020
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,883
   Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.021
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,883
   Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.022
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,883
   Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.023
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,883
   Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.024
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,883
   Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.025
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,883
   Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.026
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,971
   Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.027
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,883
   Number of sequences in database: 548

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.028
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,970
   Number of sequences in database: 549

Database: /scratch/huangchv/mpiblastDB/bcinerea.cds.029
   Posted date:  Sep 20, 2010 11:41 AM
   Number of letters in database: 673,882
   Number of sequences in database: 548
Appendix 2.3

This Perl script was used to parse text formatted BLAST output files into BLAST statistics of interest in tab-delimited format. When comparing a sequence against a BLAST database, BLAST will return the best matches that received an e-value higher than the default threshold or the user-specified threshold. Sometimes, between a query sequence and its matching subject sequence, BLAST may identify multiple matching regions separated by large gaps and will return them as separate HSPs. For each HSP, BLAST will, by default, return an alignment and the accompanying raw alignment score, bit score, e-value, and identity. In addition, the overall score for the match sequence will be from the best-scoring HSP. An example of a BLAST output is shown in Appendix 2.2.

The identity value outputted by BLAST is relative to the size of the HSP, which may be only over a portion or portions of the query. In addition, there may be multiple HSPs and each will have its own identity score. As a result, not only are the identity scores not comparable when different genomes are queried by the same sequence as the HSP length may not be the identical, but it may also not be reflective of the actual level of identity between the query and subject sequences. Thus, the MapTiling module in BioPerl was used to recalculate the identity value so it can more accurately represent the level of similarity between the query and the subject. This module functions by stitching together the HSPs if multiple HSPs were found, and returns an identity score according to the stitched alignment. Moreover, the module can be set to return an identity score relative to the size of the query, and can also account for overlaps in the HSPs.

Filename: BP_BlastParse.pl
Created on: 2010/08/20
Usage: perl BP_BlastParse.pl blast.file >output.file
# Takes text formatted BLAST files and returns percent ID for top hit, e-values and bit scores for top two hits
# Should work for all flavours of BLAST

use strict;
use warnings;
use Bio::SearchIO;
use Bio::Search::Tiling::MapTiling;

print "Query\tQuery Length\t# of Hits\tSubject01\tSubject Length\t# of HSPS\tBits01\tE-value01\tQuery Coverage\tRecalculated Percent Identity\tSubject02\tBits02\tE-value02\n";
# Open blast file in text format
my $SearchIO = Bio::SearchIO->new(-file=>ARGV[0],-format=>"blast");

# Pull each blast result
while( my $result = $SearchIO->next_result ){  
   # because the top hit is the best hit, the scalar 'i' acts as a counter and allows retrieval of top 2 hits
   my $i=0;
   # print out query name, length and number of hits
   # queries that do not return hits will also be printed
printf("%s\t%d\t%d\t",
    $result->query_name,
    $result->query_length,
    $result->num_hits,
);  
#pull each blast hit
while( my $hit = $result->next_hit ) {  
    if ($i==0) {  
        #top hit
        #Tile the HSPs in the hit
        my $tiling = Bio::Search::Tiling::MapTiling->new($hit);
        #Because this module likes references, they will be converted using printf
        printf("%s\t%d\t%d\t%d\t%3.1f\t%3.1f",
            $hit->name,
            $hit->length,
            $hit->num_hsps,
            $hit->bits,
            $hit->significance,
            $tiling->frac_aligned(-type=> 'query') * 100,
            $tiling->frac_identical(-type=> 'query', -action=> 'max') * 100,
        );
        $i++;
    }  
    elsif ($i==1) {  
        #prints the data for the second hit if it exists
        printf("%s\t%5d\t",
            $hit->name,
            $hit->bits,
            $hit->significance,
        );
        $i++;
    }  
}  
#newline for next blast result
print "\n";
Appendix 2.4

This Perl script was used to condense all the parsed BLAST files into a single table to make viewing easier. Downstream calculations such as means and geometric means were performed using the output of this script.

#Filename: CondenseParsed.pl
#Created on: 2010/09/17
#Usage: perl CondenseParsed.pl
#Takes parsed blast results and condenses them into a single table

use strict;
use warnings;
my $currentline;
my %finaltable;
my $lineforhash;

#Generate an array of all parsed blast files in the BLASTJOBS/ folder
#This can be changed to reflect where your BLAST files are kept and the extension of the parsed file outputs
my @parsedfiles = <BLASTJOBS/*.parsed.xls>;
#The query names will be extracted from the first file and this will be used as the counter
my $i=0;

#Open each blast file in text format
foreach my $files (@parsedfiles) {
    open (THISFILE, $files) || die("Opening file failed");
    while (<THISFILE>) {
        $currentline = $_;
        chomp $currentline;
        #Checks if the word "query" is at the beginning of the line, if so, this line contains the headers
        if (($i==0) && ($currentline =~ m/^Query/)) {
            next;
        }
        #Extracts the info and places into memory
        #Query name is used as the hash key to contain the information
        elsif ($i==0){
            $lineforhash = "$2$3$4$5$6";
            $finaltable{$1} = $lineforhash;
        } #if no hit is returned, query name is still grabbed for the hash key
        elsif ($currentline =~ /(\S+\t)(\S+\t)(\S+\t)(\S+\t)/) {
            $lineforhash = "$2$3$4$5$6";
            $finaltable{$1} = $lineforhash;
        }
    }
}

134
$lineforhash = "$2\t\t\t";
$finaltable{$1}= $lineforhash;
}
# Subsequent blast jobs do not need to regenerate hash keys
# Data from subject is extracted and appended to hash
else {
    if ($currentline =~ /(^S+t)(S+t)+S+t(S+t)+/ { 
        $lineforhash = "$3$4$5$6";
        $finaltable{$1}= $finaltable{$1}.$lineforhash;
    }
    # If no hits, then tabs are appended to hash as placeholder
    elsif ($currentline =~ /(^S+t)(S+t)+/ ) { 
        $lineforhash = "$\t\t\t";
        $finaltable{$1}= $finaltable{$1}.$lineforhash;
    } }
    close THISFILE;
    $i++;
}

# Print results to file
open (OUTPUT, "parsedresults.xls") || die("Could not output\nCheck if file already exists\n");
my @test = keys(%finaltable);
foreach (@test) { 
    print OUTPUT "$_$finaltable{$_}\n";
}
close OUTPUT;
Appendix 2.5
This script was used in part to compile the mean identity across several genomes for *Neurospora crassa*.

#Filename: AssignMeanID
#Created on: 2011/03/15
#Usage: perl AssignMeanID.pl <list of genes> <mean ID file>
#This script was used in part to compile the mean identity across several genomes for ncrassa
#It takes the ncrassa and the target subject genome hits, and matches mean identity calculated for
#the gene in the subject genome to the ncrassa gene id

use strict;
use warnings;

my @source;
my %meanid;
my $line;
my @keyid;

#open list of n.crassa gene ids and its corresponding subject gene ids, read to memory
open LIST, $ARGV[0];
@source = <LIST>;
close LIST;
chomp @source;

#open list of subject gene id with their mean identity values, read to memory
open IDENTITY, $ARGV[1];
while (<IDENTITY>) {
    $line = ";
    chomp $line;
    if ($line =~ /^((S+) t(S+))/) { $meanid{ $1 } = "$2"; }
}
close IDENTITY;

@keyid = keys(%meanid);

foreach my $currentline (@source) {
    if ($currentline =~ /(S+) t(S+)/) {
        print "$1$t$2$t$meanid{ $2 }\n";
    } else {
        print "$currentline\n";
    }
}
Appendix 3.1
Top 25 genes from *Neurospora crassa* (version 4 from Ensemble) with the highest 8-genome mean identity in the filamentous ascomycetes. Mean identities were calculated using 8 genomes as compared to 48 other filamentous ascomycete genomes using tBLASTn. Identity values are relative to the length of the query sequence.

<table>
<thead>
<tr>
<th>Gene ID</th>
<th>Sequence Description</th>
<th>8-Genome Mean Identity</th>
</tr>
</thead>
<tbody>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC009282</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC004982</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC006444</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC000306</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC003616</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC006543</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC002582</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC001100</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC008941</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC000314</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC005022</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC000581</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC008456</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC003139</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC004337</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC003261</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC006216</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC002502</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC003921</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC005670</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC001503</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC003898</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC007365</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC001649</td>
</tr>
<tr>
<td>gln</td>
<td>ncrassa</td>
<td>NC001633</td>
</tr>
</tbody>
</table>
Appendix 3.2

This Perl script was used to concatenate the alignments of genes into superalignments containing 5, 10, 15, 20, and 25 genes. The user supplies up to 25 alignments in a directory, but if fewer than 25 alignments are available, the script will generate superalignments up to the largest superalignment possible. For example, if only 19 alignment files are in the directory, the script will only generate the 5-gene, 10-gene, and 15-gene superalignments.

#!/bin/perl
#
#Usage: perl BP_Cataligns.pl
#Place into folder containing all the alignments with the extension "seq" and run. All alignments are assumed to have the same number of taxa and names.
#The order of the sequences does not matter. Script uses bioperl to concatenate the sequences into a superalignment.
#Will output a superalignment containing 5, 10, 15, 20 and 25 alignments.

use strict;
use warnings;
use Bio::SeqIO;

my %catalign;
my $key;
my $counter = 0;
my $previous;

#Generate an array containing all filenames in folder containing 'seq'
my @alignedfiles = <*.seq*>;

#For each of the files, open them using SeqIO and read
foreach my $files (@alignedfiles) {
    $counter ++;
    my $source = Bio::SeqIO->new(-file => $files,-format => "Fasta");

    #If this is the first alignment file, it will generate an entry in the hash
    if ($counter == 1) {
        while (my $currentob = $source->next_seq()) {
            $key = $currentob->id();
            $catalign{$key} = $currentob->seq();
        }
    }
    #All subsequent alignment files will be added to the hash
    else {
        while (my $currentob = $source->next_seq()) {
            $key = $currentob->id();
            $previous = $catalign{$key};
            $catalign{$key} = $previous.$currentob->seq();
        }
    }
}
#Outputs superalignment file containing 5, 10, 15, 20, 25 alignments
if ($counter == 5) {
    my $filename = "5.seq";
    open OUT, ">$filename";
    foreach my $key (keys(%catalign)) {
        print OUT ">$key
$catalign{$key}
";
    }
}
elsif ($counter == 10) {
    my $filename = "10.seq";
    open OUT, ">$filename";
    foreach my $key (keys(%catalign)) {
        print OUT ">$key
$catalign{$key}
";
    }
}
elsif ($counter == 15) {
    my $filename = "15.seq";
    open OUT, ">$filename";
    foreach my $key (keys(%catalign)) {
        print OUT ">$key
$catalign{$key}
";
    }
}
elsif ($counter == 20) {
    my $filename = "20.seq";
    open OUT, ">$filename";
    foreach my $key (keys(%catalign)) {
        print OUT ">$key
$catalign{$key}
";
    }
}
elsif ($counter == 25) {
    my $filename = "25.seq";
    open OUT, ">$filename";
    foreach my $key (keys(%catalign)) {
        print OUT ">$key
$catalign{$key}
";
    }
}
Appendix 3.3

This Perl script was found online and adapted to convert all Fasta files with a specific extension in the folder into a relaxed Phylip format (no ten character limit for names).

#!/usr/local/bin/perl
#From: http://protist.wikispaces.com/RAxML+Perl+Scripts
my @files = <*.seq>;
foreach $t1 (@files) {
    my @sequences = ();
    $outf = " > ".$t1.".phylip";
    open (F, " < ".$t1);
    $taxa = -1;

    while($line = <F>) {
        if($line =~ />/) {
            $taxa++;
            $name = $line;
            $name =~ s/\[\s+|\(|\)|\],|;\]/_/_g;
            $name =~ s/,//g;
            $name =~ s/>//g;
            $taxonNames[$taxa] = $name;
        } else {
            $seq = $line;
            $seq =~ s/\s+//g;
            $sequences[$taxa] = $sequences[$taxa].$seq;
        }
    }

    close(F);

    for($i = 0; $i <= $taxa; $i++) {
        print $taxonNames[$i]." ".(length($sequences[$i]))."\n";
    }

    $s = $taxa + 1;
    $bp = length($sequences[0]);

    open (F, $outf);

    print F $s." ".$bp."\n";

    for($i = 0; $i <= $taxa; $i++) {
        print F $taxonNames[$i]." ".$sequences[$i]."\n";
    }
}
Appendix 3.4

This Perl script was used to concatenate the alignments of genes into 100 gene superalignments. The user indicates the number of superalignments desired and the script will try to generate as many superalignments as requested until all the genes have been used up, or the indicated number of superalignments has been reached. Because the script samples without replacement, if 950 alignments were supplied but 10 100-gene superalignments were requested, only nine 100-gene superalignments will be generated.

#!/usr/bin/perl
#use lib "/opt/sharcnet/biop/Perl/current/lib/perl5";
#use Bio::Perl;
#
#Usage: perl bp_st-cataligns-xtrees.pl num.of100genetrees
#Goes through list of *.fasta alignment files and concatenates into the specified number of 100
gene superalignments
#Script exits when all the genes have been used up
#Genes are randomized through the Fisher-Yates shuffle algorithm

use strict;
use warnings;
use Bio::SeqIO;

my %catalign;
my $key;
my $counter = 0;
my $previous;
my $jcounter = 1;
my $genecounter = 100;
my $limit = $ARGV[0] + 1;

#Retrieve a list of the fasta files
my @alignedfiles = <*.fasta>;

#Calls Fisher-Yates shuffle subroutine
fy_shuffle(@alignedfiles);

foreach my $files (@alignedfiles) {
    $counter ++;
    #Open file
    my $source = Bio::SeqIO->new(file => $files, format => "Fasta");
    #If it is the first sequence, generate the hash
    if ($counter == 1) {
        while (my $currentob = $source->next_seq()) {
            $key = $currentob->id();
            $catalign{$key} = $currentob->seq();
        }
    }
}
# Otherwise, append to the hash
else {
    while (my $currentob = $source->next_seq()) {
        $key = $currentob->id();
        $previous = $catalign{$key};
        $catalign{$key} = $previous.$currentob->seq();
    }
}

# When the number of genes hit 100, output to file, reset counter and hash
if ($counter == $genecounter) {
    my $filename = "100-$jcounter.seq";
    open OUT, ">$filename";
    foreach my $key (keys(%catalign)) {
        print OUT ">$key
$catalign{$key}\n";
    }
    foreach my $key (keys(%catalign)) {
        $catalign{$key} = "";
    }
    $genecounter = $genecounter + 100;
    $jcounter++;
}
    # Exit if there are no more genes
    if ($jcounter == $limit) {exit;
}

# Subroutine for Fisher-Yates Shuffle algorithm
sub fy_shuffle {  
    my $array = shift;
    my $i = @$array;
    while ( --$i ) {
      my $j = int rand( $i+1 );
      @$array[$i,$j] = @$array[$j,$i];
    }
}
This Perl script was used to concatenate all alignments in a directory into a single superalignment.

```perl
#!/usr/bin/perl
#use lib "/opt/sharcnet/bioperl/current/lib/perl5";
#use Bio::Perl;
#Usage: perl bp_st-cataligns-super.pl
#Generates superalignment

use strict;
use warnings;
use Bio::SeqIO;
my %catalign;
my $key;
my $counter = 0;
my $previous;

my @alignedfiles = <*.fasta>;

#Calls Fisher-Yates shuffle subroutine
fy_shuffle(@alignedfiles);

foreach my $file (@alignedfiles) {
    $counter ++;
    my $source = Bio::SeqIO->new(file => $files, format => "Fasta");
    if ($counter == 1) {
        while (my $currentob = $source->next_seq()) {
            $key = $currentob->id();
            $catalign{$key} = $currentob->seq();
        }
    } else {
        while (my $currentob = $source->next_seq()) {
            $key = $currentob->id();
            $previous = $catalign{$key};
            $catalign{$key} = $previous.$currentob->seq();
        }
    }
}

my $filename = "$counter.seq";
open OUT, ">$filename";
foreach my $key (keys(%catalign)) {print OUT ">$key
$catalign{$key}
";}

#Subroutine for Fisher-Yates Shuffle algorithm
sub fy_shuffle {
    my $array = shift;
    my $i = @$array;
    my %catalign;
    my $key;
    my $counter = 0;
    my $previous;

    my @alignedfiles = <*.fasta>;

    #Calls Fisher-Yates shuffle subroutine
    fy_shuffle(@alignedfiles);

    foreach my $file (@alignedfiles) {
        $counter ++;
        my $source = Bio::SeqIO->new(file => $files, format => "Fasta");
        if ($counter == 1) {
            while (my $currentob = $source->next_seq()) {
                $key = $currentob->id();
                $catalign{$key} = $currentob->seq();
            }
        } else {
            while (my $currentob = $source->next_seq()) {
                $key = $currentob->id();
                $previous = $catalign{$key};
                $catalign{$key} = $previous.$currentob->seq();
            }
        }
    }

    my $filename = "$counter.seq";
    open OUT, ">$filename";
    foreach my $key (keys(%catalign)) {print OUT ">$key
$catalign{$key}
";}

    #Subroutine for Fisher-Yates Shuffle algorithm
    sub fy_shuffle {
        my $array = shift;
        my $i = @$array;
```

Appendix 3.5

This Perl script was used to concatenate all alignments in a directory into a single superalignment.

```perl
#!/usr/bin/perl
#use lib "/opt/sharcnet/bioperl/current/lib/perl5";
#use Bio::Perl;
#Usage: perl bp_st-cataligns-super.pl
#Generates superalignment

use strict;
use warnings;
use Bio::SeqIO;
my %catalign;
my $key;
my $counter = 0;
my $previous;

my @alignedfiles = <*.fasta>;

#Calls Fisher-Yates shuffle subroutine
fy_shuffle(@alignedfiles);

foreach my $file (@alignedfiles) {
    $counter ++;
    my $source = Bio::SeqIO->new(file => $files, format => "Fasta");
    if ($counter == 1) {
        while (my $currentob = $source->next_seq()) {
            $key = $currentob->id();
            $catalign{$key} = $currentob->seq();
        }
    } else {
        while (my $currentob = $source->next_seq()) {
            $key = $currentob->id();
            $previous = $catalign{$key};
            $catalign{$key} = $previous.$currentob->seq();
        }
    }
}

my $filename = "$counter.seq";
open OUT, ">$filename";
foreach my $key (keys(%catalign)) {print OUT ">$key
$catalign{$key}
";}

#Subroutine for Fisher-Yates Shuffle algorithm
sub fy_shuffle {
    my $array = shift;
    my $i = @$array;
```
while ( --$i ) {
    my $j = int rand( $i+1 );
    @$array[$i,$j] = @$array[$j,$i];
}
}
Appendix 3.6

This script was used to randomly select genes for the control superalignment trees. The script reads a list of gene, and randomly selects a specified number of genes without replacement. The number of independent replicates can also be specified.

#!/bin/perl
#Created on: 2011/02/03
#Filename: SelectRandom.pl
#Usage: perl SelectRandom.pl genelist.file num.inset num.reps
#Shuffles list of genes and randomly selects and prints 'y' sets of 'x' number of genes from list
#
use strict;
use warnings;

my $i=0;
my $k=0;
my $number = $ARGV[1];
my $reps = $ARGV[2];

#Opens list of gene and reads into array
open LIST, $ARGV[0];
my @array = <LIST>;
close LIST;
chomp @array;

for ($k=0; $k<$reps; $k++) {
    print "$number genes, replicate: $k
    #Calls Fisher-Yates shuffle subroutine
    fy_shuffle(@array);
    $i=0;
    #Prints out specified number of genes
    for ($i=0; $i<$number; $i++) {
        print "$array[$i]
    }}
#Subroutine for Fisher-Yates Shuffle algorithm
sub fy_shuffle {
    my $array = shift;
    my $i = @$array;
    while ( --$i ) {
        my $j = int rand( $i+1 );
        @$array[$i,$j] = @$array[$j,$i];
    }
}
Appendix 3.7. An example of a multiple sequence alignment (A) before being trimmed by TrimAl v1.2 with a gap threshold of 0.75 and (B) after.
Appendix 3.8

This script was used to randomly select a set of genes and outputs the mean of mean identities. The script reads a list of gene, and randomly selects a specified number of genes without replacement. The mean of mean identities of the selected genes is then calculated and outputted. The number of independent replicates can also be specified.

#!/bin/perl
#Created on: 2011/02/03
#Vincent Huang
#Filename: SelectRandom-avgmeanid.pl
#Usage: perl SelectRandom-avgmeanid.pl genelist.file num.inset num.reps
#Shuffles list of genes, randomly selects genes, and retrieves its mean identity
#Calculates the mean of the mean identities of the sampled genes
#Used to determine if random selection will yield a mean close to the mean of the distribution from Chapter 2

use strict;
use warnings;

my $i=0;
my $k=0;
my $number = $ARGV[1];
my $reps = $ARGV[2];
my %genelist;
my @genearray;
my @totalscore;
my @tempscore;
my $avgscore;

#Opens list of gene and mean id, reads into hash, then move genes into array
open LIST, $ARGV[0];
while (my $currentline = <LIST>) {
  chomp $currentline;
  $currentline =~ /\S+/\t\S+/;
  $genelist{$1}=$2;
}
my @array = keys %genelist;
close LIST;
for ($i=0;$i<10;$i++) {
  $totalscore[$i]=0;
}
for ($k=1; $k<=$reps; $k++) {
  #print "Replicate number $k\n";
  Calls Fisher-Yates shuffle subroutine
  fy_shuffle(@array);
  #Prints out specified number of genes
@genearray=();
for ($i=0; $i<$number; $i++) {
    push(@genearray, $array[$i]);
}

#Pass gene array into subroutine to get average mean identity of selected genes in bins
calc_avg(@genearray);
print "Average mean identity of $number genes. Replicate $k of $reps:	$avgscore"
}

#Subroutine for Fisher-Yates Shuffle algorithm
sub fy_shuffle {
    my $array = shift;
    my $i = @$array;
    while ( --$i ) {
        my $j = int rand( $i+1 );
        @$array[$i,$j] = @$array[$j,$i];
    }
}

#Subroutine to calculate the average score of the genes in the array
sub calc_avg {
    my $array = shift;
    my $i=0;
    my $score = 0;
    $avgscore = 0;
    foreach my $currentline (@$array) {
        $score += $genelist{$currentline};
    }
    $avgscore = $score / $number;
}
Appendix 3.9 – Computational Constraints Encountered in Chapter 3

Maximum likelihood tree generation for large datasets requires a substantial amount of computing power and time. One of the potential problems was the maximum runtime allowed on both Sharcnet and Westgrid clusters. None of them permit jobs to run for more than a week without a special project request.

For the ten 100-gene trees bootstrapped with 1,000 replicates, the first tree to complete took about 30 hours on 20 cores (Intel Xeon L5420 quad-core processors, running at 2.5GHz; all nodes were connected using Infiniband) and 15 GB of memory while the last one to complete required 80 hours and almost 20 GB of memory. On a single core and without considering the amount of memory required, the fastest 100-gene tree would have required 600 hours or 25 days to complete, and 67 days for the slowest tree. In addition, time and computing requirements do not scale linearly with the number of genes in the superalignment. Thus, generating a large superalignment tree using over 100 genes with 1,000 bootstrap replicates may have required weeks of computing time. Based on the multigene tree results however, using more than 15 genes may not have been necessary since all ten 100-gene superalignment trees returned the same tree shape as the 5-, 10, and 15-gene trees.

Due to computational constraints, a model test was performed on three superalignments rather than all 70 superalignments. All three returned GTR+Γ+I as the optimal model to use. This led to another problem since the addition of the ‘I’ parameter to the evolutionary model ramps up the run time of RAxML such that the larger superalignments will no longer complete within the maximum allowable run time and memory on the Checkers computer cluster. The ‘Γ’ parameter is used in evolutionary models to estimate the amount of substitution rate heterogeneity among sites, which is more realistic than assuming an equal distribution of rates. The ‘I’ parameter is the proportion of invariant sites. When the ‘I’ parameter is included, the program has to
simultaneously optimize the values of both ‘I’ and ‘Г’. It has been argued that since ‘I’ is merely a subset of ‘Г’, it is often not necessary to include ‘I’ as a parameter when generating an ML tree (Stamatakis 2008). Since model tests were also attempted for the 100-gene superalignments but each attempt failed to complete due to memory issues, GTR+Г was used for all superalignments.

For large real-world biological datasets, model tests will frequently identify GTR+Г+I or GTR+Г as one of the most optimal if not the most optimal model to use because the less complex models lack the parameters to account for real-world biological complexity (Stamatakis 2008). In addition, it has been shown that trees constructed using alternative models with likelihood scores close to the optimal model are not statistically different from the optimal model though the differences that do appear between trees as a result of model choice occur at poorly supported nodes (Ripplinger and Sullivan 2008). In these cases, the problem may lay within the data or data choice rather than the model choice.
Appendix 4.1

This sample Perl script was used to identify potential priming sites in multiple sequence alignments. This script requires the TrimAL program.

#Filename: BuildOligos.pl
#Usage: perl BuildOligos.pl alignment.file
#Prints out potentially conserved regions that can be used to build primers for PCR
#based on similarity scores of the sites in the alignment
#Trimal is needed because it is used to calculate the similarity scores
#

use strict;
use warnings;
use autodie;
use List::Util qw(max);

#This is the alignment file, if it cannot open, die and return error
my $alnfile = $ARGV[0] || die "Invalid file name";
#Grabs the main part of the filename for outputting purposes
my ($filename) = $alnfile =~ /(\w+).*/;

my %scores;
my $i;
my @window;
my $windowsum;
my $max;
my $start;
my $stop;
my $threshold = 18;
my @forwardprimers;
my %allfprimers;
my @reverseprimers;
my %allrprimers;
my $currentline;

#Calls the trimal program and generates the score file, if it cannot generate, it dies and returns an error message
system ("g:\lab\vincent\TRIMAL\trimAL\bin\trimal.exe -in $alnfile -scc >$filename.scores") == 0 or die "Could not generate scores. Check alignment file.";

#reads the score file that was just generated into memory
open SOMEFILE, "$filename.scores";
while (<SOMEFILE>) {
    my $temp = $_;
    chomp $temp;
    if ($temp =~ /\s+(\d+)\s+(S+)/) {
        my $score = \1;
        my $position = \2;
        $scores{$position} = $score;
    }
}

my %scores;
my $i;
my @window;
my $windowsum;
my $max;
my $start;
my $stop;
my $threshold = 18;
my @forwardprimers;
my %allfprimers;
my @reverseprimers;
my %allrprimers;
my $currentline;
$scores{$1} = $2;
}
} close SOMEFILE;

#Sort and grabs the number of entries in the score list. This is the maximum length of the alignment
my @hashkeys = sort{$a <=> $b} keys(%scores);
$max = scalar @hashkeys;

#Builds the initial window of 18 residues (sites 0-17)
i=0;
until ($i==18) {
    #print "$i\t$scores{$i}\n";
    push(@window, $scores{$i});
    $i++;
}

#Sends array to subroutine and calculates the score
$windowsum = &calcforward( @window );
$start = $i-17;
$stop = $i;

#Place into hash that will contain all the potential sites
$allfprimers{"$start-$stop"} = "$windowsum";

#Checks if the score is greater than the threshold. If so, add to list of primers.
if ($windowsum >= $threshold) {
    $currentline = "$start-$stop\t$windowsum\n";
    push(@forwardprimers, $currentline);
}

$windowsum = &calcbackward( @window );
$start = $i;
$stop = $i-17;
$allrprimers{"$start-$stop"} = "$windowsum";
if ($windowsum >= $threshold) {
    $currentline = "$start-$stop\t$windowsum\n";
    push(@reverseprimers, $currentline);
}

#From here on in, it will add one base to the end and remove a base from the beginning; calculate and checks if window score has surpassed the threshold for potential primer
#If score passes threshold, output range of bases and the score for it
until ($i==$max) {
    shift(@window);
push(@window, $scores{$i});

#This is for the forward primer
$windowsum = &calcforward( @window );
$start = $i-16;
$stop = $i+1;
$allfprimers{ "$start-$stop" } = "$windowsum";
if ($windowsum >= $threshold) {
    $currentline = "$start-$stop\t$windowsum
    push(@forwardprimers, $currentline);
}

#This is for the backward primer
$windowsum = &calcbackward( @window );
$start = $i+1;
$stop = $i-16;
$allrprimers{ "$start-$stop" } = "$windowsum";
if ($windowsum >= $threshold) {
    $currentline = "$start-$stop\t$windowsum
    push(@reverseprimers, $currentline);
}
$i++;

open SOMEFILE, ">$filename.primers";

#Best priming sites
print SOMEFILE "Primers generated from alignment file: \n $filename\n
print SOMEFILE "Highest Scoring Primers:\n
my @forwardkeys = sort { $allfprimers{ $b } <=> $allfprimers{ $a } } keys %allfprimers;
my @reversekeys = sort { $allrprimers{ $b } <=> $allrprimers{ $a } } keys %allrprimers;
print SOMEFILE "Forward\n
print SOMEFILE "$forwardkeys[0]\t\tallfprimers{ $forwardkeys[0] }\n
print SOMEFILE "Reverse\n
print SOMEFILE "$reversekeys[0]\t\tallrprimers{ $reversekeys[0] }\n
#Primers exceeding threshold
print SOMEFILE "Primers exceeding optimal threshold:\n
print SOMEFILE "Forward\n
foreach (@forwardprimers) {
    print SOMEFILE "$_\n
print SOMEFILE "\n
print SOMEFILE "Reverse\n
foreach (@reverseprimers) {
    print SOMEFILE "$_\n
print SOMEFILE "\n
#Best primers in general, even if threshold not met
print SOMEFILE "nBest primers possible from alignment file:n
n"
print SOMEFILE "Forward\n"
my $j=0;
until ($j>=10) {
    if (exists $forwardkeys[$j]) {
        print SOMEFILE "$forwardkeys[$j]\t$allfprimers{\$forwardkeys[$j]}\n";
        $j++;
    } else { $j=10; }
}
print SOMEFILE "nReverse\n"
$j=0;
until ($j==10) {
    if (exists $reversekeys[$j]) {
        print SOMEFILE "$reversekeys[$j]\t$allrprimers{\$reversekeys[$j]}\n";
        $j++;
    } else { $j=10; }
}

#Subroutine to calculate the total of the array
#Calculates the score but gives greater weight to conservation at the 3` end
sub calcforward {
    my $current = 0;
    my @internal = @_; my $j=0;
    my $jmax = scalar(@internal);

    #Calculates the total initial 13 bases
    until ($j==13) {
        $current = $current + $internal[$j];
        #print "$j\t$current\n";
        $j++;
    }

    #Calculates the rest of the bases; if the score of the column is greater than 0.5, apply a bonus of 50%
    #Otherwise, just add it on
    until ($j==$jmax) {
        if ($internal[$j] > 0.5) {
            $current = $current + $internal[$j]*1.5;
        } else {
            $current = $current + $internal[$j];
        }
    }
# Calculates the score but gives greater weight to conservation at the 5` end
sub calcbackward {
    my $current = 0;
    my @internal = @_; # Calculates the total initial 5 bases; if the score of the column is greater than 0.5, apply a bonus of 50%
    my $j=0;
    my $jmax = scalar(@internal);

    # Otherwise, just add it on
    until ($j==5) {
        if ($internal[$j] > 0.5) {
            $current = $current + $internal[$j]*1.5;
        } else {
            $current = $current + $internal[$j];
        }
        $j++;
    }
    # Calculates the rest of the bases
    until ($j==$jmax) {
        $current = $current + $internal[$j];
        $j++;
    }
    return $current;
}

# Disclaimer
print SOMEFILE "

**** NOTE ****
This script will only generate primers based on levels of similarity in the alignment given.

If your alignment has been trimmed, or edited, some primers suggested may not work.

Remember, garbage in, garbage out.
";
Appendix 4.2. Forward and reverse priming regions for the *N. crassa* gene NC001938 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.

```
<table>
<thead>
<tr>
<th>10</th>
<th>367</th>
<th>877</th>
<th>887</th>
<th>897</th>
</tr>
</thead>
</table>
B.cinerea | ACCTTCTGGGGATAGCACAACCTCTGGGGTCACGTAAC |
S.sclerotiorum | ATTTTCTGGGGATAGCACAACCTCTGGGGGACCGTAAC |
M.fijiensis | ATCTTCTGGGGGATAGCACAACCTCTGGGGACCGTAAC |
M.graminicola | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
P.marneffei | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
T.stipitatus | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
P.chrysogenenum | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
A.flavus | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
A.oryzae | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
A.nidulans | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
A.niger | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
A.clavatus | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
A.fischeri | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
A.fumigatus | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
M.canis | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
M.gypseum | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
T.rubrum | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
T.equimun | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
P.tritici-repens | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
C.heterostrophus | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
A.brassicicola | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
G.clavigera | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
C.parasitica | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
P.anserina | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
M.oryzae | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
N.discreta | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
N.crassa | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
N.tetrasperma | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
N.haematococca | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
F.graminearum | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
P.oxysporum | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
C.globosum | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
S.thermophile | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
A.capsulatus | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
U.reesel | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
C.immitis | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
C.posadasii | ATCTTCTGGGGGATAGCACAACCTCTGGGGGACCGTAAC |
```

Consensus

```
ATCTTTGGCTGGGACATGCAACCCGTCAC
```
Appendix 4.3. Forward and reverse priming regions for the *N. crassa* gene NC003210 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.
Appendix 4.4. Forward and reverse priming regions for the *N. crassa* gene NC004139 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.

<table>
<thead>
<tr>
<th>Species</th>
<th>Forward Region</th>
<th>Reverse Region</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>S. nodorum</em></td>
<td>ATGACTACAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. brassicicola</em></td>
<td>ATGACTACAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>P. tritici-repentis</em></td>
<td>ATGACTACAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>C. heterostrophus</em></td>
<td>ATGACTACAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>T. equinum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>T. tonsurans</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>S. sclerotiorum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>B. cinerea</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. capsulatus</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>M. canis</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>M. gypseum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>T. rubrum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. benhamiae</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>T. verrucosum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>U. reesei</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>C. immitis</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>C. posadasii</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>P. marneffei</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>T. stipitatus</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>P. chrysogenum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. clavatus</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. fischeri</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. fumigatus</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. terreus</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. niger</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. nidulans</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. flavus</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>A. oryzae</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>M. fijiensis</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>M. gaminicola</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>M. oryzae</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>G. clavigera</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>V. albo-atrum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>V. dahliae</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>T. reesei</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>T. atroviride</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>T. virens</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>N. haematococca</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>F. graminearum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>F. oxysporum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>F. verticilloides</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>N. discreta</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>N. crassa</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>N. tetrasperma</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>C. globosum</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>C. parasitica</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>P. anserina</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
<tr>
<td><em>S. thermophile</em></td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
<td>ATGATCAAGGGACGCTGTCCTCACTGATCGTCCTGG</td>
</tr>
</tbody>
</table>

Consensus: ATGTAGTACATGTGGATTGGGTTTCATGCTCATCTTCTGG
Appendix 4.5. Forward and reverse priming regions for the *N. crassa* gene NC004477 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.
Appendix 4.6. Forward and reverse priming regions for the *N. crassa* gene NC006543 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.
Appendix 4.7. Forward and reverse priming regions for the *N. crassa* gene NC006868 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.

![Consensus](image-url)
Appendix 4.8. Forward and reverse priming regions for the *N. crassa* gene NC007341 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.

Consensus

ATGGGCAAGAACACCATGGTAAGGACCGCATTGCCAACCC
Appendix 4.9. Forward and reverse priming regions for the *N. crassa* gene NC008393 and its homologs from the other filamentous ascomycete genomes. The level of similarity is displayed below.