A Thesis Submitted for the Degree of Doctor of Philosophy

Computational Discovery and Analysis of rDNA Sequence Heterogeneity in Yeast

Claire Louise West

September 2013

University of East Anglia
Institute of Food Research

Abstract

Ribosomal RNA genes, known as ribosomal DNA or rDNA, are commonly found in tandem arrays of hundreds of repeating units. The sequences of each unit in an array were thought to be near-identical but it is now known that frequent mutations may occur, causing heterogeneity amongst units. Opposing these divergent mutational processes, unit sequences are homogenised through concerted evolutionary processes such as unequal sister chromatid exchange (USCE) and gene conversion (GC).

In this study Perl software has been used to uncover rDNA sequence variation in the yeast *Saccharomyces paradoxus*, using data derived from the Saccharomyces Genome Resequencing Project. This analysis, in conjunction with a reanalysis of the *Saccharomyces cerevisiae* data from the same project, has provided detailed information regarding rDNA sequence heterogeneity in two contrasting, yet closely-related yeast species. Additionally, the rDNA flanking sequences of four yeast strains have been characterised via an analysis of new next generation sequencing reads, adding to our knowledge of concerted evolutionary processes in these genomic regions.

Partial Single Nucleotide Polymorphisms (pSNPs) within these datasets are shown to reflect genome mosaicism within a population, and to identify strains with signs of genome hybridisation undetectable by other means. This information provides further insights into the dynamics of the rDNA region in the two yeast species. In particular, examination of the percentage occupancies of pSNPs reveals U-shaped distributions which differ between the two species.

Further investigations of rDNA evolutionary dynamics through the development of two Java simulation tools (SIMPLEX and CONCERTINA), which model USCE and GC events, follow the fate of both single and multiple pSNPs in one or more rDNA arrays. Initial simulations show the distribution of pSNPs varies depending upon the balance between mutations and concerted evolutionary events, and provide a framework to investigate the mechanisms involved in altered rDNA dynamics in various cellular processes.

Contents

List of Figures

List of Tables

Acknowledgements

I'd first like to acknowledge direct contributions. Dr Steve James, for identifying and explaining complex mutations, Dr Jo Dicks for the statistical analyses in section 3.3.2, and performing some of the BLAST alignments in Chapter 7. Finally, Dr Jo Dicks, Dr Ian Roberts, and Dr Steve James for extensive proof reading and editing, especially Chapter 4.

More generally, I would like to thank the BBSRC for funding this project. My supervisor Dr Ian Roberts for his support throughout this project, including some difficult transition periods, a disheartening case of writer's block, and helping to give me the confidence to talk about my work in front of more than 100 people! Also to my co-supervisor, Dr Jo Dicks, to whom I can attribute my desire to do a PhD, for editing and extensive proof reading (and patience when my ability to write seemed to have escaped me!), for explaining logically all plans for the project, for her support, and for being an inspiring female scientist. Also, the rest of my supervisory team for their time throughout this project. I'd further like to thank Dr Rob Davey for his help with his program, although it was impossible to beat his acronyms! I'd also like to thank the NCYC for being such a welcoming group, and patiently explaining any yeast related information.

As part of the wider experience, I'd like to thank the CSB department for providing a home for most of my PhD. The bioinformaticians for the nice coffee breaks, and making me feel like part of a team, and the systems biologists for stimulating and occasionally bewildering lunch time conversations, and for making me realise you are big softies underneath it all. I'd like to acknowledge my lovely friends Marcus, Jen and Claire for being there for me and *many* cups of tea. I'd like to especially thank Antony, for a ton of support, proof-reading, cheering me up, stopping me worrying (as much as possible!), listening to me talk about this *a lot*, and generally being the great person that he is.

Finally I'd like to thank my family, and in particular my parents, for the love, support and encouragement that they've always given. They've supported me through having M.E, anxiety, and now a science PhD! Thanks for always being there, and none of this would have happened without you. x

1. Introduction

1.1. Ribosomes and Ribosomal RNA

The central dogma of molecular biology is that DNA is transcribed into mRNA, which is then translated into protein. Translation of the mRNA into protein is undertaken by the ribosome, which is therefore an essential part of the cell's molecular machinery. A review of the process of translation relating to the ribosome's structure is presented in (Ramakrishnan, 2002), and a simplistic view of the process is shown in figure 1.1. Due to the ribosome's function it is essential for the survival of a cell, making it a target of antibiotics to inhibit ribosomal function in certain pathogens (Yonath, 2005). The importance of a detailed understanding of the ribosome has recently been recognised by the awarding of the Nobel Prize in Chemistry 2009 to Venkatraman Ramakrishnan, Thomas A. Steitz and Ada E. Yonath for their work in studying ribosomal structure and function by x-ray crystallography (Ramakrishnan, 1986; Ban et al., 2000; Harms et al., 2001). The background of their work on the ribosome, its importance, and information on the Nobel prize work can be found in a press release accompanying the prize (Ehrenberg, 2009).

Due to the size and complexity of the ribosome, a high-resolution structure has only recently been determined for the eukaryotic 80S yeast ribosome (Ben-Shem et al., 2011). The sub structure of the eukaryotic ribosome consists of RNA strands in a small subunit (18S RNA with 33 proteins) and a large subunit (5.8S, 5S and 26S RNA with 46 associated proteins), as illustrated in figure 1.2. The majority of the structure, about 60%, consists of RNA, and given it's role in protein production the structure is much larger than that of most proteins.

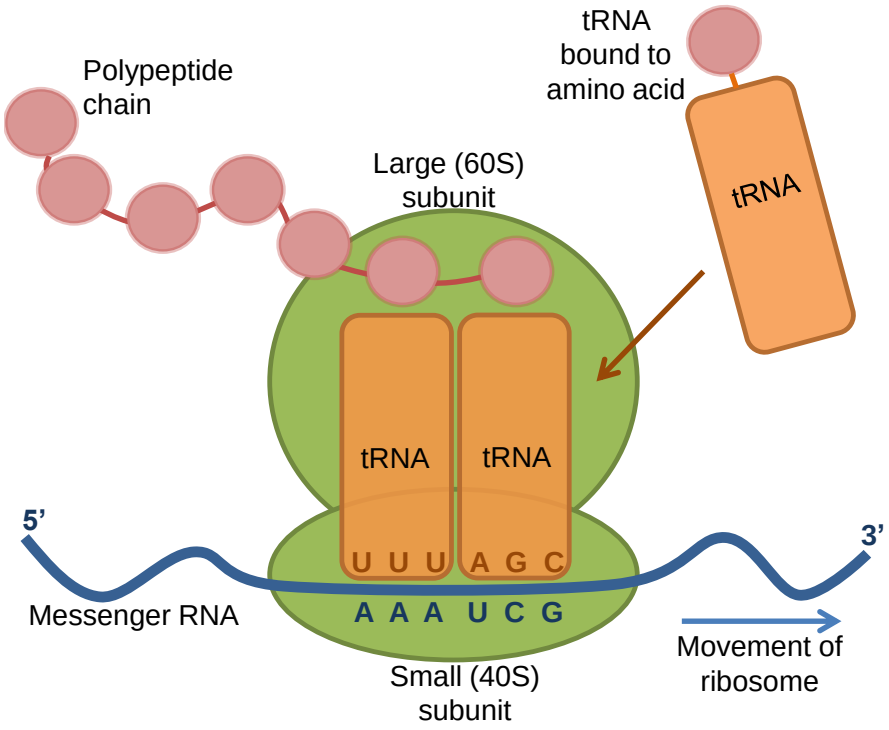

Figure 1.1.: Illustration of function of the ribosome (shown in green) in polypeptide synthesis.

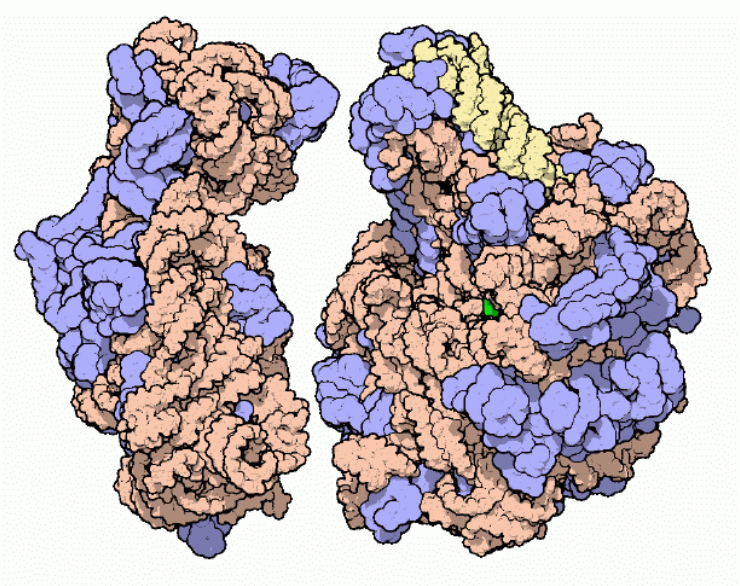

Figure 1.2.: Illustration of small subunit (left) and large subunit (right) of the ribosome, from `http://www.rcsb.org/pdb/education_discussion/molecule_of_the_month`. Orange and yellow chains are RNA strands, blue are proteins. Image from the RCSB PDB September 2008 Molecule of the Month feature by David Goodsell (`doi:10.2210/rcsb_pdb/mom_2008_9`)

1.1.1. Ribosomal DNA Structure

Due to the essential nature of ribosomes, their sequence and structures are highly conserved within, and across, species. The genes which encode ribosomal RNA are referred to as rDNA (ribosomal DNA).

Ribosomal RNA genes are encoded in a tandem array of repeating units, illustrated in figure 1.3 in yeast. Each unit is separated by non-transcribed regions of DNA, also called intergenic spacer (IGS) regions (shown in grey in figure 1.3), which contain sites for a replication fork barrier (RFB, in IGS1) and an autonomous replicating sequence (ARS, in IGS2). Within the unit itself, there are regions encoding the RNA components of the small and large subunits of the ribosome, an external transcribed spacer region (ETS region) at the end of the unit, and internal transcribed spacers (ITS) between the RNA encoding genes. In the yeast *S. cerevisiae*, the rDNA is present as a single array of tandemly repeated units (approximately 150 copies), accounting for 60% of Chromosome XII. In eukaryotes the rDNA forms the nucleolus organizer region (NOR), around which the nucleolus is made, as shown in figure 1.4. If the rDNA is present at more than one locus, each one comes together within the nucleolus, the site at which rDNA is transcribed. A three-dimensional model of the yeast genome was created in 2010, by using experimental techniques including chromosome conformation capture-on-chip and high-throughput parallel sequencing to detect interactions within and between chromosomes, which were then used to create a map (Duan et al., 2010). The authors findings implicated the nucleolus and rDNA in preventing interactions between the ends of Chromosome XII by forming a barrier between the ends of the chromosome.

RNA has long been associated with *S. cerevisiae*, with RNA originally being named “yeast nucleic acid”. There is far more RNA than DNA in yeast cells, at a ratio of 50:1, and with most of that RNA (80%) comprising of rRNA, rDNA has been a natural target for studies in yeast.

Large numbers of ribosomes are needed during phases of rapid growth. In *S. cerevisiae* 200,000 ribosomes are present in each cell (Warner, 1999), and so many copies of the rDNA unit are necessary. Studies have shown that one copy of the rDNA unit would not be enough to satisfy the amount of ribosomes needed during high cell growth periods in *E. coli* (Bremer, 1975). Furthermore, each stage of transcription of DNA to RNA, then translation of mRNA to protein results in

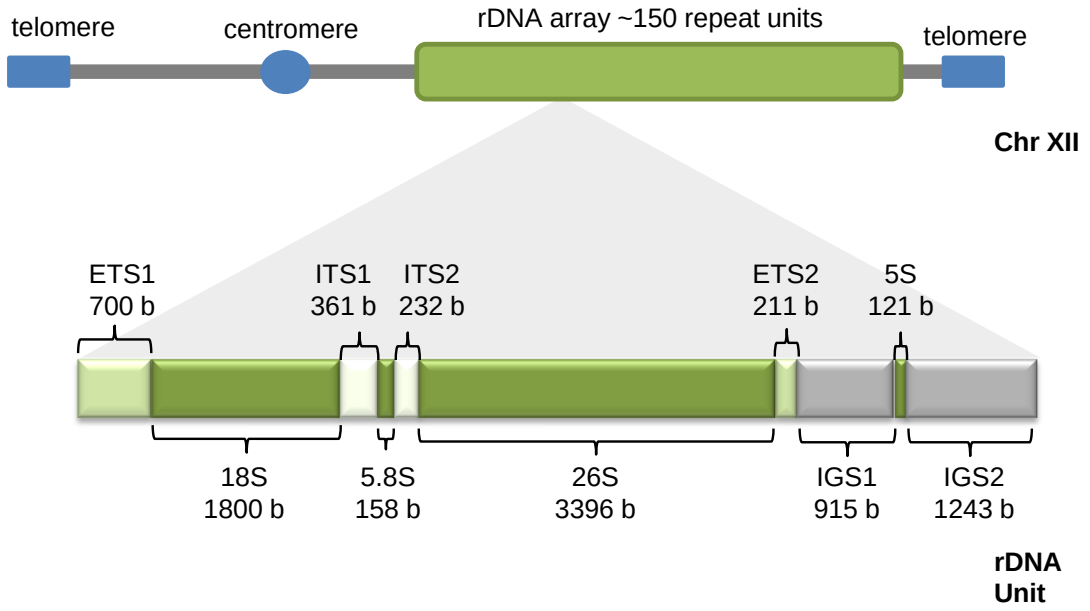

Figure 1.3.: rDNA repeat structure. The rDNA locus in Chromosome XII in the yeast *Saccharomyces cerevisiae*, and an example of an rDNA unit, including the length, in bases, of each region.

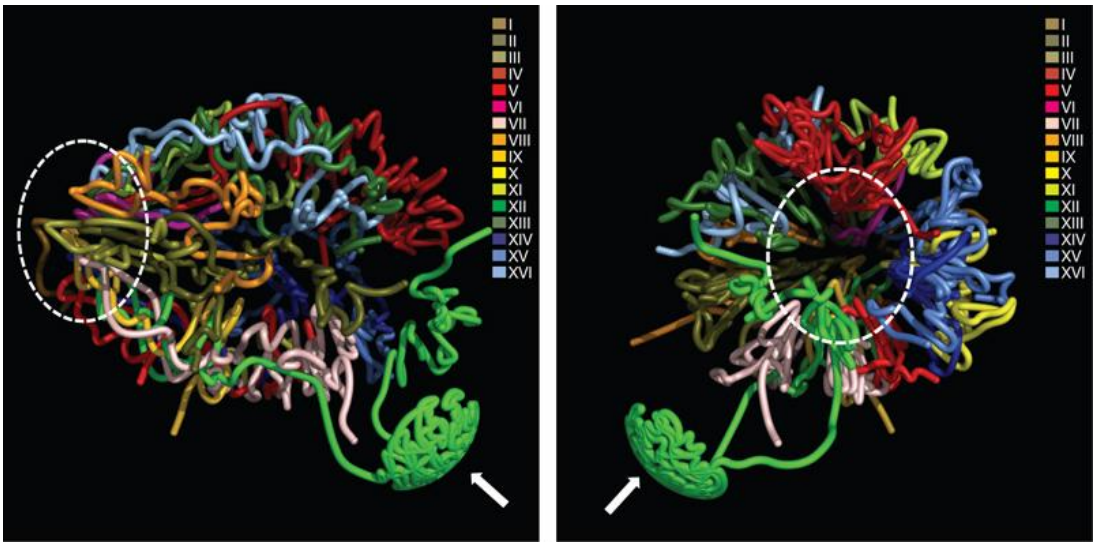

Figure 1.4.: Three-dimensional model of the yeast genome from (Duan et al., 2010). Chromosomes cluster in the nucleus at one pole, shown as the dotted oval. The rDNA repeats on Chromosome XII (shown in green), separate and form the nucleolus, identified by the white arrow. Reprinted by permission from Macmillan Publishers Ltd: Nature (Duan et al., 2010), copyright 2010

amplification in a product of a gene (where one gene is transcribed into many RNA molecules, each of which is translated many times into many copies of the protein). Because rRNA is not translated into protein, it does not benefit from the amplification that would usually occur at this stage. However, the number of repeats of the rDNA genes varies greatly between organisms. This variation will be reviewed in the next section.

1.2. rDNA Variation

1.2.1. Genomic Organisation of rDNA Loci

In eukaryotes the number of rDNA repeats varies greatly, as does the number of loci, for example; humans possess approximately 400 copies of the rDNA unit across 5 different chromosomes(chromosomes 13, 14, 15, 21 and 22; Henderson et al., 1972), the model plant *Arabidopsis thaliana* contains ∼570 copies across 2 chromosomes (chromosome 2 and 4; Weiss and Maluszynska, 2000), and the model organism *Drosophila melanogaster* has a few hundred rDNA units on the X and Y chromosomes (Stage and Eickbush, 2007). As previously mentioned, in the yeast *Saccharomyces cerevisiae* the rDNA is present as a single tandem array, which also contains the 5S locus as part of the array (Hillier et al., 1997). In most hemiascomycetes the 5S gene is within this array, however in the majority of other eukaryotes the 5S is dispersed throughout the genome, either as part of another array (as in *Drosophila*), or as discrete units (Richard et al., 2008).

Also of interest, although the location of rDNA within a chromosome has been shown to be important for its function, a recent study using synteny to investigate the evolution of the location of rDNA in 17 yeast species of Saccharomycetaceae found that the complete rDNA array has moved around the genome on a number of occasions (Proux-Wéra et al., 2013).

1.2.2. rDNA Copy Number Variation

The number of repeated elements in the rDNA array varies between species. There is much interest in copy number variation of rDNA and its implications,

for example how this might affect the use of rDNA in the study of microbial diversity. Due to this interest and the sizeable amount of variation between species a publicly available database which collates data on rRNA operon copy numbers in Bacteria and Archaea, called rrnDB (`http://rrndb.mmg.msu.edu`; Klappenbach et al., 2001; Lee et al., 2009), has been developed. Several general rRNA sequence databases are also available, including the SILVA database; (`http://www.arb-silva.de/`, Pruesse et al., 2007), the Ribosomal Database Project II (`http://rdp.cme.msu.edu/`; Cole et al., 2009), and the greengenes project (`http://greengenes.lbl.gov/`; DeSantis et al., 2006).

A study by Prokopowich (Prokopowich et al., 2003) to investigate correlations between genome size and rDNA copy number in eukaryotes looked at 162 species of plants and animals. They found that copy number varied between 39 and 19,300 in animals, and between 150 and 26,048 repeats in plants. This study found a positive correlation between genome size and rDNA copy number using a Pearson product-moment correlation. This result does not appear to hold true for prokaryotes, as a paper by Fogel *et al.* in 1999 found that there appeared to be no correlation between rDNA copy number and genome size in this taxonomic group (Fogel et al., 1999).

Prokaryotic organisms have much lower copy numbers of ribosomal DNA units than eukaryotes, many with only one unit. The review by Fogel *et al.* looked at 101 different prokaryotic taxa, and found the mean copy number to be 3.8 (Fogel et al., 1999). They found that more than 10% of the prokaryotes investigated had 1 rDNA unit, with the highest copy number (12 units) being found in a small number of *Bacillus cereus* strains (ATCC 10987; Johansen et al., 1996). This review also noted that some *Azomonas* and *Bacillus* species possessed strain-specific copy numbers. A study by Liao also looked at rDNA in bacteria and archea, although focusing on the mechanism of evolution and whether it may differ from eukaryotes (Liao, 2000). As part of this study Liao noted that although multiple copy genes are unusual in prokaryotes, the presence of multiple units of ribosomal RNA genes is nevertheless necessary due to a high demand for fast protein synthesis in growing cells. It is also of note that, unlike eukaryotic rDNA, multiple copies of the rRNA genes are not tandemly arrayed, but are spread throughout the genome. The size of the repeat unit itself is smaller in prokaryotes, but Liao comments that this smaller repeat number and gene unit size may mean a more comprehensive analysis of the repeating gene and its mechanism of evolution is easier in prokaryotes (Liao, 2000). A more recent review by Tourova considered the effect that multiple copy number of rDNA in prokaryotes may have on phylogenetic analyses for use

in identification of prokaryotes in environmental samples (Turova, 2003). This study noted that genomic separation of the multiple copies meant that the rDNA copies are transcribed separately, and that approximately half of all prokaryotes in the rrnDB database had only one or two copies of the operon. Whether there is an ecological advantage to having a higher rDNA copy number is not well defined. Klappenbach *et al.* (Klappenbach et al., 2000) investigated if rDNA copy number in a community of diverse bacteria correlated to the rate of reaction to the availability of resources. They found that on average those bacteria with a higher number of rDNA copies had a faster response time, and copy number correlated with the rate that soil bacteria formed colonies in response to resources.

It is of note that some researchers believe that the variation in rDNA copy number between species can introduce bias into estimations of species abundance. Crosby *et al.* compared four different rRNA genetic techniques which are regularly used to assess microbial community diversity. The study found that error was introduced due to the variation in copy number between species, with a bias towards those organisms containing a higher copy number (Crosby and Criddle, 2003). Therefore this bias should be kept in mind in any estimations using these techniques.

Importance and Maintenance of rDNA Copy Number

It is interesting to note that whilst a high rDNA copy number is maintained within many species, there is an in-built redundancy such that not all units are actively transcribed. A 1993 study on rDNA chromatin structure within the rDNA (Dammann et al., 1993) showed that only a proportion of the rDNA regions were actively transcribed, and furthermore that this proportion could be changed in response to different growth conditions. However, although only approximately half of the units are transcribed in the case when there are around 150 copies, studies have shown yeast strains with lower copy number still produce the same overall amount of rRNA (French et al., 2003).

A number of studies have been carried out into how the rDNA repeat number is stabilized and the importance of maintaining it, particularly by the Kobayashi group, including a recent review (Kobayashi, 2011). The expansion and contraction of the number of repeats in the rDNA array in the yeast *S. cerevisiae* has been investigated (Kobayashi et al., 1998). This study identified that when a subunit of RNA polymerase I (Pol I) was absent , there was a gradual decrease in the number of repeats in the rDNA array, dropping to approximately half the original number.

When Pol I was reintroduced the number of repeats slowly increased back to the initial number, illustrating how copy number is maintained. This study also implicated a DNA replication fork blocking protein, Fob1, in maintaining repeat number, demonstrating that replication fork blocking stimulates recombination by encouraging Double-Strand Breaks (DSBs). Fob1 is therefore involved in changing rDNA copy number, and so sequence homogeneity (Kobayashi et al., 1998). The Fob1 protein has also been suggested as a possible mechanism for movement of the rDNA to different chromosomes in the evolution of yeast (Proux-Wéra et al., 2013).

rDNA copy number has also been linked to genome integrity, with large numbers of rDNA repeats in yeast being shown to protect against DNA damage by mutagens (Ide et al., 2010). Non-transcriptionally active rDNA units were shown to facilitate recombinational repair mechanisms by aiding cohesion between sister chromatids, and so aid efficient repair to damaged DNA. Genome wide effects of rDNA were also highlighted in a study in *Drosophila*, which linked changes in copy number of the rDNA on the *Drosophila* Y chromosome to changes in gene expression elsewhere in the genome (Paredes et al., 2011).

The location of the rDNA locus has also been linked to copy number, in *S. cerevisiae* (Kim et al., 2006). As described in an earlier section, in *S. cerevisiae* rDNA is present as a single locus on Chromosome XII. In this study, a series of truncated variants of Chromosome XII were created, splitting the chromosome either side of the rDNA locus, with only one variant containing the rDNA. The authors observed that those variants containing the left side of the chromosome, or the left side plus the rDNA, had shorter lifespans and accumulated extrachromosomal rDNA circles (ERCs). This study indicates that the placement of rDNA within chromosome XII is pertinent to maintaining copy number and also to the function of rDNA.

1.2.3. rDNA Sequence Variation

Several studies investigating the level of variability of rDNA sequences within and between species have been carried out. Ben Ali *et al.* (1999) constructed a variability map for one area of the rDNA sequence, that encapsulated the large RNA subunit. The authors used a Substitution Rate Calibration method to examine the evolutionary rate of a particular nucleotide in comparison to the

average evolutionary rate (Ben Ali et al., 1999). They identified conserved and variable sites within this coding region of rDNA across 77 eukaryotic species. Their findings indicated that the less variable regions often encode functionally important structures. However, Lachance *et al.* found that the large rDNA subunit in the yeast *Clavispora lusitaniae* contained polymorphisms, despite its functional importance, questioning the use of this area in indicating species boundaries (Lachance et al., 2003).

rDNA variation can be affected by different factors. Stage and Eickbush studied sequence variation within the rDNA of 12 different Drosophila species using Whole Genome Shotgun Sequencing (WGSS), and found results consistent with concerted evolution (discussed in the following section) (Stage and Eickbush, 2007). However, they found fewer polymorphisms than may have been expected in the 28S gene, which they hypothesize is due to localized gene conversion or DNA repair within retrotransposable elements specific to that subunit. A study by Ganley and Kobayashi (Ganley and Kobayashi, 2007) found little sequence variation within the rDNA arrays of 5 fungal species, and inferred from this that there must be a mechanism of rapid homogenization. In contrast to this a study by James *et al.* found that rDNA was highly variable in 34 different strains of *S. cerevisiae*, especially in the IGS regions (James et al., 2009). The authors also note that many of the polymorphisms were not fully resolved (i.e. that sequence variation between units within an array exists). A more recent study of the ITS region of rDNA in *Arabidopsis thaliana* also found variation within individual genomes (Simon et al., 2012). An older study of rDNA variation within plant populations and individual plants by Schaal and Learn found the IGS region to be variable within both populations and individuals, and of potential use in the study of microevolution (Schaal and Learn, 1988).

1.3. Uses and Consequences of rDNA Sequences

1.3.1. rDNA Sequences in Phylogenetics

rDNA sequences have been used in molecular phylogenetics for a number of years. Several reviews on this area have been developed; a comprehensive review by Hillis and Dixon in 1991 summarized how rDNA has been used to infer phylogenetic relationships (Hillis and Dixon, 1991), with other reviews by Olsen and Woese in

1993 (Olsen and Woese, 1993), Woese in 2000 (Woese, 2000), and Turova in 2003 (Turova, 2003), amongst others.

There are many reasons why rDNA sequences have been popular in their use in phylogenetics. Woese explained that rDNA sequences are resistant to horizontal gene transfer events, which can complicate phylogenetic inferences (Woese, 2000). Another review also explained that rDNA sequences vary in size, and rate of substitution varies across the rDNA unit, which enables rDNA to be used to infer both distant and fine scale phylogenetic relationships (Olsen and Woese, 1993). Not only are rDNA sequences present in all species, but they have the same function in each, and are experimentally easy to work with (Woese, 2000). rDNA has also been studied for a relatively long time as its sequence could be characterized before DNA cloning and sequencing methods were available (Eickbush and Eickbush, 2007), meaning methods of analysis are well developed.

There has been criticism on the use of rDNA sequences in phylogenetic studies however. Many state that trees developed from rDNA information are not truly organismal as they only represent a small part of a genome. However it is noted by Woese that there is no consensus as to what would be a more appropriate alternative (Woese, 2000).

Commercial applications of phylogenies obtained from rDNA variation have also been made. A phylogenetic analysis of *S. cerevisiae* by Montrocher *et al.* (Montrocher et al., 1998) used polymorphisms within the rDNA spacer regions to construct phylogenetic relationships in wine yeasts, and suggested that this method could be used to rapidly characterize yeast strains for the food industry.

1.3.2. rDNA, Disease and Ageing

As well as rDNA being important for phylogenetic studies, there are a number of other scientific areas which involve the analysis of rDNA sequences. For example, in humans, overexpression of rDNA has been observed in prostate cancer (Uemura et al., 2012). rDNA also shows promise as a predictor of disease progression (Stults et al., 2009), due to it possessing recombinational hotspots and therefore being a common site of chromosomal aberrations in tumours.

rDNA has also been identified as playing a role in cellular ageing and senescence.

Extrachromosomal rDNA circles (ERCs) have been shown to accumulate in old cells and can cause replicative ageing, and are the reason for the nucleolar enlargement seen in older cells (Sinclair and Guarente, 1997). One study suggested ERCs reduce the replicative lifespan by inducing instability within the rDNA, which causes senescence (Ganley et al., 2009). There have been a number of reviews on ageing in yeast, which discuss other potential mechanisms for ERCs causing a decrease in replicative lifespan. These include ERCs resulting in more rRNA within the cell, which could impair ribosome production and function, or transcription factors which are associated with rDNA interact with ERCs instead, limiting their normal role with the rDNA (Steinkraus et al., 2008; Kaeberlein, 2010). ERCs are formed by intra-chromatid recombination, when a double-strand break is repaired by homologous recombination. If the broken end pairs with a unit on the same chromatid, a circle of one or more rDNA units is formed. A recent study identified a major quantitative trait locus (QTL) in yeast, linking an increase of 41% in replicative lifespan to the rDNA region (Kwan et al., 2013). This QTL was identified as a polymorphism in the origin of replication in rDNA, which reduced replication starting from within the rDNA, but increased replication throughout the rest of the genome. Approximately a third of the origins of replication within yeast are found within the rDNA, so having fewer and weaker origins in the rDNA allows other, weaker genomic origins of replication to compete. This offers an alternative explanation to another study which found that origin activity reduced the number of ERCs, which paradoxically decreased replicative lifespan, which the authors attributed to increased rDNA instability (Ganley et al., 2009).

1.3.3. Problems with rDNA Arrays in Assembly of Genomes

The repeating units of the rDNA in *Saccharomyces cerevisiae* were found to be on Chromosome XII by Petes in 1979 (Petes, 1979). However when the yeast whole genome sequence was published in 1996 only the units at the extreme ends of the rDNA repeat were published (Goffeau et al., 1996). This reduction was in part due to the high degree of similarity between repeats leading to difficulties in distinguishing between them, and thus not allowing full assembly. Furthermore it may be difficult to distinguish between genuine variation and sequencing errors, especially in highly similar tandem arrays. Consequently all repeats throughout the array were assumed to possess identical sequences.

A study by Stults *et al.* in 2008 investigated ribosomal RNA gene clusters, discussing the difficulties in assembling highly repetitive regions of the genome in the Human Genome Project, and how the human rDNA region was also left unassembled (Stults et al., 2008). A modest number of programs attempting to assemble these types of highly repetitive sequence have now been developed. In 2003 Tammi *et al.* introduced a program to assemble shotgun sequencing data, including highly similar repetitive data, called the Tandem Repeat Assembly Program or TRAP (Tammi et al., 2003). The TRAP method involved five steps:

1. **Preparation** - sequences scanned against database to remove contamination. The remaining vector sequences and any poor quality 5' or 3' reads are marked for later removal.
2. **Computation of overlaps**.
3. **Analysis of overlaps from repeated regions** - overlaps scored using error frequencies and false overlaps removed using multiple alignments.
4. **Generate fragment layout** - using heuristic algorithm.
5. **Generate consensus sequence**.

Another tool developed more recently by the same research team is called DNP Trapper. This software is a shotgun sequencing finishing tool which allows manual estimations and visualizations as well as automatically assigning placements (Arner et al., 2006). Their paper also notes that the more recent Whole Genome Shotgun Sequencing (WGSS) technologies can introduce more problems with assembling repeated regions. This is due to previous sequencing techniques allowing handling of repeat regions locally, whereas WGSS requires all repeat regions to be handled at the same time, even if they are spread throughout the genome (Arner et al., 2006). Although their method aids assembly of repetitive regions, the test data they used from *Trypanosoma cruzi* only had a maximum of 8 repeat units, with most averaging one or two copy number repeats. However, the authors stated that the software could be used to visualise mammalian size genomes.

1.4. Mechanisms of rDNA Variation

To understand and predict evolutionary relationships between species or strains using rDNA variation, the mechanisms underlying it need to be understood. The current consensus of opinion is that concerted evolution is the process by which variation (usually point mutation) introduced into a single rDNA unit is

homogenised across the array, either leading to fixation or loss. The concerted evolutionary processes are explained in more detail in the following sections.

1.4.1. Concerted Evolution - Unequal Crossover and Gene Conversion

Concerted evolution is a term used to describe the observation that repeated genes evolve together, "in concert" with each other, with the phrase being in use since 1980 (Zimmer et al., 1980). Prior to that date, the same process was known under many names, including "Horizontal Evolution" (Brown et al., 1972) and "Coincidental Evolution" (Hood et al., 1975).

Although it has been believed for some time that homogenisation of the rDNA array is due to concerted evolution, the exact contribution of the possible mechanisms involved has not been resolved. A review by Eickbush and Eickbush summarizes concerted evolution in relation to rDNA (Eickbush and Eickbush, 2007), with the two major mechanisms described below.

Unequal Sister Chromatid Exchange

This mechanism, also known as unequal recombination, could explain the ability of rDNA repeats to evolve in tandem together. Selection pressures alone cannot explain the uniformity of rDNA sequences within a species (Eickbush and Eickbush, 2007). Instead, a correction mechanism would be needed to spread any mutations throughout the array to maintain conformity. Furthermore, as we saw in section 1.2.2, the number of repeats in the rDNA array can vary between individuals, a phenomenon potentially explained by homologous recombination between different repeats in the array (Eickbush and Eickbush, 2007). Random unequal crossover could account for chromosomes with differing numbers of repeats harbouring a polymorphism. Chromosomes with mutations in the transcriptional units would be selected against generally, whereas those in the noncoding spacer regions would be under no deleterious selective pressure, and therefore the number of repeats with mutations in these regions would increase and decrease in a stochastic manner. This would mean that over time a substitution would become present or absent from all of the repeated rDNA units, a duration known as the fixation time.

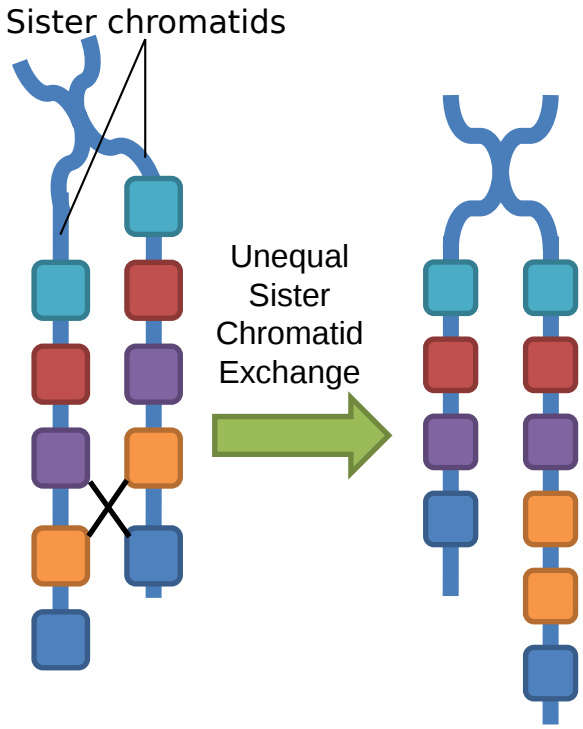

Figure 1.5.: Illustration of Unequal Sister Chromatid Exchange (USCE), with different coloured blocks representing different units within an rDNA array. Sister chromatids misalign, and can crossover during the DNA repair process (crossover indicated by the crossed lines), resulting in sister chromatids being unequal in size. In this way sequences can proliferate throughout a region, or become lost.

In 1980 Petes inserted the yeast *LEU2* gene into the rDNA array of *S. cerevisiae* and followed the outcome of meiotic events on the presence of this gene using tetrad analysis (Petes, 1980). The study found that the presence of the *LEU2* marker became unstable during meiosis, and that marker loss in one array was coupled with duplication in another. Furthermore, this was shown to only occur as a result of exchange between sister chromatids, not homologous chromosomes. This provided evidence for unequal recombination between rDNA sequences on sister chromatids as a major mechanism for homogenisation, as illustrated in figure 1.5. However, despite the ability of unequal recombination to explain much of the variation observed within the rDNA array, it became uncertain whether it could account for all of it.

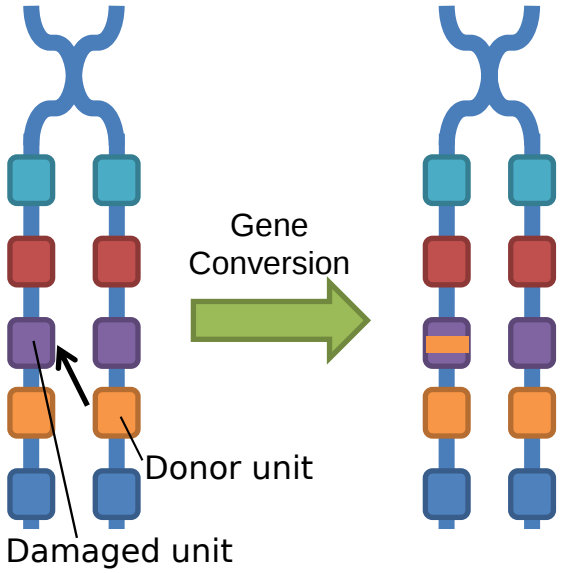

Figure 1.6.: Illustration of gene conversion, the different units within an rDNA array are represented as different coloured blocks. A double-stranded break in a sequence can be repaired using a template from a homologous region, resulting in a section of DNA being copied from one area to another. In the example above, the orange unit is used as a template for repair, and so its sequence is spread.

Gene Conversion

The mechanism of gene conversion was demonstrated as a possible contributor to rDNA variation and homogenization in yeast via mathematical models, most prominently the model of Nagylaki and Petes in 1982 (Nagylaki and Petes, 1982). Gene conversion is the mechanism in which recombination occurs between different DNA helices in a non-reciprocal manner, in a "copy-paste type" event, as demonstrated in figure 1.6. On introducing their program for estimating gene conversion rates from SNP data (Yin et al., 2009), Yin *et al.* describe a gene conversion in a descendant sequence as the result of copying a small segment or 'conversion tract' from a particular location in one parent sequence, to the same position in the other parent sequence. In contrast, Yin *et al.* describe a crossover descendant as containing a prefix of one parent with the suffix of the other.

Gene conversion can explain features of concerted evolution over and above those explained by unequal crossover (Eickbush and Eickbush, 2007):

- how sequence homogeneity of rDNA units on both homologous and non-homologous chromosomes could occur.
- how the sequence uniformity at the terminal repeats of the rDNA could be

accounted for.

- how, more generally than the rDNA case, sequence homogeneity of multigene families dispersed through a genome could be achieved.

However, it has been relatively difficult to gain experimental evidence of the involvement of gene conversion in concerted evolution of the rDNA sequence. A study by Hillis *et al.* (1991) implicated biased gene conversion as the main driver of concerted evolution in rDNA in asexual parthenogenic lizards (Hillis et al., 1991). This study also demonstrated that rDNA concerted evolution can be driven by biased, directional processes as well as stochastic ones. The triploid genome of the asexual parthenogen is created from two distinct parental haploids, resulting in three Nucleolar Organising Regions (rDNA) in the resulting triploid, two from one parent and one from another. When 109 parthenogenic individuals from the *Heteronotia* species were investigated, one parental genotype was favoured and had either fixed or was in a greater proportion than the other genotype, which indicated biased gene conversion rather than USCE (which would have resulted in some individuals possessing a fixed rDNA genotype from the other parent). However, this observation could be limited to the rather specialized case of parthenogenic species.

Another study implicating gene conversion with rDNA expansion and contraction was by Gangloff *et al.* (Gangloff et al., 1996). The authors found that less than 30% of their results in maintaining sequence homogeneity in rDNA within yeast could be explained by unequal sister chromatid exchange. In prokaryotes, Liao found 'striking' patterns of concerted evolution, and found gene conversion played a major role in sequence homogenization (Liao, 2000).

One biological model used to explain the process of gene conversion is the double-stranded break repair (DSBR) model, roughly illustrated in figure 1.7 (Szostak et al., 1983). In the DSBR model, a double-stranded break is repaired using a homologous sequence as a template. Depending on how the resulting double Holliday junction is resolved either a crossover or non-crossover product is produced, with the non-crossover product being a gene conversion, copy-paste type event as shown in figure 1.6.

Another model very similar to DSBR is the Synthesis-Dependent Strand Annealing (SDSA) model, which is identical to the that shown in figure 1.7 until the "New DNA synthesis" section. In SDSA, only one D-loop is formed, and no Holliday structures are seen. A thorough description of this model, and others involved in

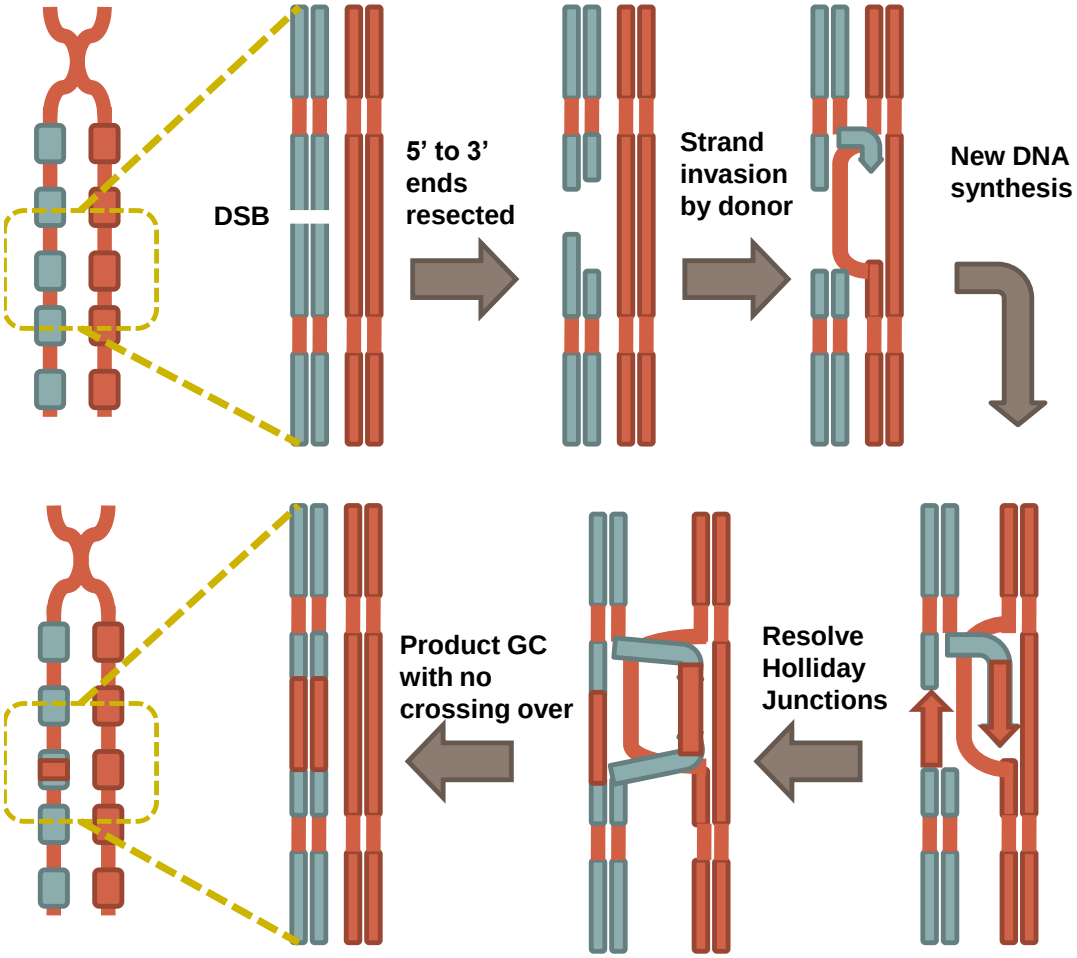

Figure 1.7.: Simplified model of Gene Conversion (GC) via Double-Stranded Break Repair (Szostak et al., 1983, adapted from Pâques and Haber, 1999). A double stranded break (DSB) is introduced. The ends are resected in the 5' to 3' direction, and then invade the homologous donor which acts as a template for repair. Two Holliday junctions are formed, and depending on how they are resolved, either a crossover or noncrossover product is obtained, in this case a noncrossover gene conversion product.

recombination, including those potentially implicated in concerted evolution, is given in Pâques and Haber (1999).

Author	Recombination events per Generation
(Szostak and Wu, 1980)	1 x 10^{-2}
(Merker and Klein, 2002)	1.3 x 10^{-3}
(Kobayashi et al., 2004)	7.4-7.5 x 10^{-4}

Table 1.1.: Table of different experimentally estimated rDNA recombination events per generation in *Saccharomyces cerevisiae*.

Other Potential Mechanisms of Concerted Evolution

A number of alternative mechanisms are potentially implicated in homogenizing the rDNA array, and in maintaining its copy number. As well as USCE and gene conversion, which both have compelling arguments and evidence for their involvement in concerted evolution, experimental evidence supporting the involvement of additional processes also exists. These mechanisms include:

- **Intrachromatid Recombination** - mentioned in an earlier section on maintaining copy number and ageing, this process produces extrachromosomal rDNA circles (ERCs) when a chromatid repairs a double-stranded break by looping over, pairing with itself, and crossing over (Sinclair and Guarente, 1997).
- **Single Strand Annealing (SSA)** - after a double-strand break is made between repeats, the broken ends can resect and anneal to another repeat further along the array, resulting in deletion of one or more units (Ozenberger and Roeder, 1991; Pâques and Haber, 1999).

Both of these mechanisms are related to recombination. A number of studies have been made which estimate the number of recombination events per generation in rDNA, however no consensus estimate has been determined at present. Examples of the different rates estimated from these studies are shown in table 1.1, obtained by investigating the rate of marker loss in the rDNA of *Saccharomyces cerevisiae*.

The balance between the various mechanisms of concerted evolution in achieving sequence homogeneity in real datasets is currently unknown. However, a number of mathematical models have attempted to describe the action of concerted evolution, which might present a way of estimating such a balance.

1.4.2. Mathematical and Computational Models

To understand the origins and evolution of rDNA variation in more detail, several mathematical and computational models have been developed over the past few decades.

In the 1970s three key papers introduced models of unequal crossover, showing how this mechanism could homogenise or create tandem arrays of genes. The first of these papers suggested that tandem repeats are the natural state of DNA not maintained by selection (Smith, 1976). In this study, a single DNA lineage undergoing unequal crossover between sister chromatids was computationally simulated. Simulations began with a 500 base pair sequence, which was subsequently modified by a random base pair substitution, followed by a series of attempted crossovers in each evolutionary cycle. Constraints for crossover product size, and sequence similarity near the crossover point were implemented. Results from these simulations illustrated that periodic tandem repeats formed from a starting sequence which contained no repeats. Smith (1976) went on to suggest that a long repeated sequence is formed from an expansion of small repetitive sequence arrays. However, the pattern of repeats, or the probability of a particular repetitive sequence being achieved, remained unknown as it would be dependent upon the mechanism used for crossover.

A further study that year built upon previous models of "coincidental" evolution by intrachromosomal unequal recombination, using principles from population genetics (Ohta, 1976). This model followed the evolution of repetitive units in a multigene family, allowing crossovers to shift the array by one unit, with alternating events leading to duplication and deletion. This study concluded that estimating the frequency of gene lineages becoming fixed in a multigene family is in principle the same as analysing the fixation of mutant alleles within a population, a standard population genetics model. Therefore the diffusion model of Kimura could be applied to this problem. In this study it was estimated that 20000, 4000, 2000, and 800 crossovers were needed for fixation in the case where the mean number of units duplicated or deleted in a single crossover event was 1, 5, 10 and 25 respectively.

The following year an additional model investigating unequal crossover in multigene families was developed. As with the previous model, misalignments of one repeat unit were permitted, but the results were estimated for a greater number of

units (Perelson and Bell, 1977). The authors constructed four different models of unequal crossing over between sister chromatids. In those models in which there was an equal probability of an expanded or contracted chromosome being kept in future generations, they inferred that if the mutation generation time was long compared to the gene fixation time a homogenised multigene family would result, exhibiting coincidental evolution. Another of their models incorporated the diffusion model, as in Ohta's model (1976), but again assumed that crossovers were balanced between duplications and deletions of equal length. This study also discussed a number of mathematical difficulties in expanding or solving some of the problems in their models, including placement of repeats across a chromosome, and in expanding their model to account for crossover between chromosomes in diploids.

An example of a model devised specifically for rDNA variation was that of Nagylaki and Petes (Nagylaki and Petes, 1982). This model proposed intrachromosomal gene conversion as the main mechanism to maintain sequence homogeneity within repeated genes. The model derives from fixation probabilities, examining the mean time it takes for a variant to become fixed or lost within a population. The model made the following assumptions:

- Heteroduplexes can form between a pair of repeated genes either symmetrically or asymmetrically.
- Interactions occur within an array in a chromatid, between repeats on sister strands, but not between chromosomes.
- All repeats have an equal probability of interacting.
- All mismatches are corrected.
- There is no reciprocal recombination.
- Sister strand interactions occur once per cell generation. Interactions within an array can occur multiple times per cell generation, but it is assumed the interactions do not overlap in time.

Their results imply that gene conversion does act in sufficient time to contribute to maintaining sequence homogeneity. The authors note that although unequal recombination is also shown to be involved in concerted evolution and possesses the necessary attributes to promote sequence homogeneity in repeated genes, gene conversion has several advantages as a correction mechanism. Firstly, it can be directional (with certain sequences being more likely to act as a donor than others). Secondly, it has the possibility of correcting errors without making gene dosage changes, and thirdly it can act on dispersed repeats as well as those in a

tandem array (Nagylaki and Petes, 1982). The latter point may be of particular interest in prokaryotes where repeats are often dispersed throughout the genome and are not tandemly arrayed.

Other models developed in the 1980s include a further model by Ohta (1985), building on the previous model (Ohta, 1976). This model assumed that repeats in a gene family were dispersed throughout a chromosome. It allowed duplicative transpositions (duplicating a single repeat and then moving it to another location, an event at that time believed to be important in the concerted evolution of transposon families) and gene conversions to occur between genomes within a diploid organism. The model enabled calculation of allelic identity coefficients, the probabilities that alleles chosen at random from a population are identical. The model did not incorporate unequal recombination, or examine at genetic correlations between chromosomal distance or bias in gene conversion or transposition.

Another study investigated the mechanism of homogenisation of the sub repeats found in the IGS (here referred to as the Non-Transcribed Spacer or NTS) region of rDNA units (Dvorák et al., 1987). The resulting model was applied to experimental data of the pattern of mutations in the NTS region across 7 clones. The results agreed that the pattern of mutation observed was consistent with a mechanism where the further away repeats are, the less likely they are to form a heteroduplex, and the less likely gene conversion is to occur between them.

A generic model for tandemly repeated genes, not specifically rDNA, was proposed by Elemento *et al.* (Elemento et al., 2002). Their model assumed unequal recombination as the main mechanism for variation and change, but also that no gene conversion events or deletions occurred. After creating a model for unequal recombination, the authors constructed an algorithm to assess the likelihood of rooted phylogenetic trees containing a duplication event, and then used these likelihoods to find the optimal duplication event tree using maximum parsimony methods. They tested their model using data from human immunoglobulin and T-cell receptors.

A more recent model was constructed during an investigation of silencing and recombination in yeast rDNA (O'Kelly, 2008). O'Kelly based his model on the idea of Unit Recombination Events, or UREs, where one repeat in the rDNA can randomly overwrite another, and such that polymorphisms can become incorporated or lost from the whole array over time. O'Kelly noted that although

unit amplification and deletion events occur *in vivo*, as the number of repeats remains the same on average these can be ignored. Another term included in this model is the 'complete fixation time', which is the amount of time taken for all repeats in a current generation to have arisen from a single repeat of an ancestor. O'Kelly looked at occupancy ratios of a mutation throughout the repeats of an array. The occupancy number follows a random walk, with absorbing barriers at 0 (where the repeat is lost), up to n (the array size) where it is fixed across the entire array. O'Kelly then simulated this model to determine occupancy ratios, and tested different models of recombination. A Bayesian analysis framework was used to assess which model was the most appropriate for available yeast datasets. After performing these analyses, O'Kelly determined that a non-uniform recombination model best explained the observed occupancy ratio distribution. In this model (unlike that of Nagylaki and Petes) some repeats were more likely to undergo recombination, with the probability of a repeat being implicated in a URE increasing linearly with distance from the edge of the array. Subsequent experimental results confirmed this URE model as the best model of those suggested (O'Kelly, 2008).

A recent attempt to model rDNA variation was also alluded to in a study which established an experimental evolution approach to studying the rate and dynamics of concerted evolution (Ganley and Kobayashi, 2011). However, to date no work has yet been published on the computational model.

A way to experimentally follow the results of concerted evolution would be beneficial, not just examining copy number but also looking at repeat sequence variation. In the next section a new type of variation is discussed that may enable a greater understanding of concerted evolutionary processes to be made.

1.5. Discovery of pSNPs

James *et al.* (James et al., 2009) analyzed over 35Mbp of rDNA sequences obtained from a Whole-Genome Shotgun Sequencing (WGSS) project involving 34 different strains of *Saccharomyces cerevisiae*. The authors looked for variation within the rDNA arrays and found that, contrary to previous findings (Ganley and Kobayashi, 2007), significant variation existed within the rDNA arrays of individual genomes. Furthermore, they found that not all repeats in a genome had fully resolved SNPs, so that only a subset of the units in the rDNA array contained SNPs at a

particular position. James *et al.* termed this new type of variation partial single nucleotide repeats or pSNPs, further suggesting them as a measure of genome stability and divergence. James *et al.* went about classifying variation within the rDNA arrays using the method outlined in figure 1.8. This method followed three stages to process the raw sequence reads.

1. Using the rDNA consensus sequence from the *S. cerevisiae* reference strain S288c (Goffeau et al., 1996), a number of 100 bp rDNA subsequences were selected in a sliding window approach at 20bp intervals, and used for gapped BLAST (Altschul et al., 1997) queries against the *S. cerevisiae* WGSS database to identify all rDNA reads. Those sequences which aligned to at least 70bp of a 100bp query sequence, and with no more than 30 mismatches, were proposed as rDNA-specific sequences. These selected reads formed a new database.
2. Less stringent BLAST searches were performed on the new database to find rDNA reads that may be divergent from the consensus S288c sequence. False positives from sequencing errors were accepted at this stage to allow full sampling of rDNA sequence variability. Minimal penalty values for mismatches and gaps in BLAST scoring were therefore used in alignments. This time the middle 20bp of the 100 bp rDNA subsequences were used to look for polymorphisms across the rDNA reads, the flanking 40 bases each side used to ensure specificity in searches. At this stage reads were accepted for further analysis if they matched to 62 or more bp of the 100bp window. These reads were then collected for multiple alignments.
3. Multiple alignments of rDNA reads were performed using MUSCLE (Edgar, 2004) with default parameters, with all redundant reads from the previous steps excluded. To distinguish polymorphisms arising from sequencing errors, Phred quality scores that were published with the SGRP WGSS database were extracted and analyzed, and stringent quality score filtering applied to the results from the previous step. Base substitutions were only accepted as true polymorphisms if they had a quality score of 40 or more. Also, polymorphisms from single reads were only accepted if they were found on two or more strains. rDNA polymorphisms were then identified and mapped for each strain, and the frequency of each polymorphism for a given strain calculated.

When the WGSS data were analyzed using this method, it was found that variation between strains differed greatly, ranging from 10 to 76 identified polymorphisms per strain. Polymorphisms identified and analysed were single-base type substitutions

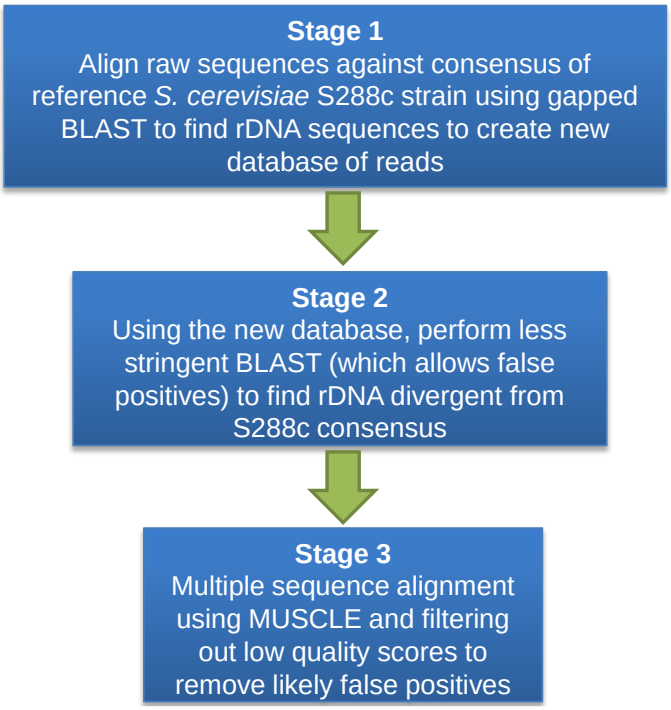

Figure 1.8.: Outline of method for discovering pSNPs (James et al., 2009).

(transitions and transversions), comprising of SNPs and pSNPs. Polymorphisms were not found to be evenly distributed across the rDNA repeat, most being found in the IGS regions. The study found that approximately 70% of the polymorphisms identified were pSNPs, some of which were only found in particular strains, others which had different frequencies according to the strain. The majority of pSNPs were found at low frequency, so when present were only found in less than 10% of the repeats in a particular array. Of particular interest, it was discovered that pSNP number correlated with mosaicism of a genome, with mosaic genomes (those with hybrid origins) having more pSNPs on average than structured genomes. This discovery led to the possibility of pSNPs being an indicator of genomic origin, with this idea being used recently in a population genomics study of wild and domestic yeast (Liti et al., 2009).

Since the James *et al.* (2009) study, other groups have identified pSNPs within the rDNA arrays of various organisms. In *Arabidopsis thaliana* pSNP variation was found in the ITS region (Simon et al., 2012). ITS intragenomic variation which affected a resulting phylogenetic analysis was discovered in species of *Laetiporus* (Lindner and Banik, 2011). Partially resolved variation within the IGS1 region was also identified in *Rhodocollybia laulaha* (Hawaiian mushroom) (Keirle et al., 2011). pSNPs are in essence a snapshot of concerted evolution in action, and have considerable potential to enable analysis of concerted evolutionary processes.

Consequently, efficient automated ways of investigating pSNP diversity would be beneficial.

1.6. SNP Calling Algorithms and their Limitations for Identifying pSNPs

Considerable effort has been put into developing computational tools to identify SNPs from DNA sequences, generally known as 'SNP calling algorithms'. SNPs can be used as genotypic markers, for example to identify common polymorphisms which could be associated, via genome wide analyses, with disease. Selected SNPs can then be used in microarray screening programs, for example on relatives of individuals with a particular disease. Two SNP calling algorithms reviewed in (Hua et al., 2007) and (Liu et al., 2003) cite disease markers as reasons to develop these tools.

But what are the difficulties in detecting SNPs? Currently no sequencing methods are error free, and with the depth of sequence coverage also varying between methods, it can be difficult to discern between a genuine SNP and the result of a sequencing error. Therefore all SNP calling methods need a way to categorize which nucleotide polymorphisms are genuine and which are false positives.

An approach by Brockman *et al.* (Brockman et al., 2008) is aimed at improving SNP detection in a particular sequencing technique, Sequencing-By-Synthesis (SBS). The SBS technique results in over and undercalls of indels, rather than the miscalls prevalent in techniques such as Sanger or Illumina sequencing. Brockman *et al.* note that it is still important to be able to compare quality scores between different sequencing techniques, even though the SBS bases may be lower quality. The algorithm itself uses Neighbourhood Quality Standard (NQS) windows to select SNPs after the sequence has been aligned to a reference, by selecting unambiguous reads to score the SNPs, where unambiguous reads have over 80% identity to the reference.

Another SNP detecting algorithm which uses NQS in a final step is the ssahaSNP program developed by Ning *et al.* (Ning et al., 2001) as part of the SSAHA database searching algorithm. SSAHA performs searches on large genomes quickly by building a data structure (hash table) containing 14-base sections of the

reference genome. For SNP detection in random human genomic reads, it uses the high quality region of a read as a query sequence, and aligns it to the reference genome. If it matches to less than 10 locations on the genome, a base by base alignment is made, and high quality base discrepancies using the Neighbourhood Quality Score (NQS) are reported as SNPs. This method was used to detect over one million SNPs that are registered within the dbSNP database.

In all SNP detection methods, allelic variation needs to be distinguished from sequencing error. Therefore a threshold of quality needs to be used to determine which of these scenarios is more likely, with low quality scores being more likely in low sequence coverage datasets. Quinlan *et al.* developed a Bayesian approach using 'Data Likelihoods', which allowed SNP calling even in the presence of low sequence coverage (Quinlan et al., 2008). They applied their application, called Pyrobayes, to datasets in which the base quality score would normally be too low to detect SNPs for 75% of the data.

Hua *et al.* use a classification based method for SNP detection, called SNiPer-HD. The method uses an expectation-maximization algorithm with parameters based on a sample training set (Hua et al., 2007) to accurately identify genotypes from thousands of SNPs, which includes steps to assign qualities or confidence in SNPs and removing those which fall below a threshold. They solve a major problem for SNP calling, the existence of a low minor allele frequency which could be ignored. However, training based algorithms require datasets with enough sample points to establish accurate parameter estimation, so efficient sampling is unlikely for thousands of SNPs with low frequency.

Liu *et al.* (2003) developed another SNP calling method, based upon the PAM classification and dissimilarity matrix (MPAM). Neural network based solutions of the SNP calling problem have also been devised. For example, Forage, developed by Unneberg *et al.* (Unneberg et al., 2005). This method uses neural networks and Bayesian approaches for SNP discovery, and uniquely uses a dynamic threshold to distinguish SNPs from sequencing errors by utilizing the non-linear classification abilities of neural networks. Furthermore, the method uses a dual network approach which only scores a SNP if both networks classify it as such. In a comparison of the Forage algorithm with the NQS based and the Pyrobayes approaches outlined above, Forage found slightly fewer false positives and negatives than PyroBayes, and considerably fewer than the NQS approach.

1.6.1. Reference-free SNP Calling

More recently, a number of reference-free SNP calling methods have been developed, which are important for identifying SNPs in more complex genomes which do not yet have an assembled genome to use as a reference. With Next-Generation Sequencing resulting in more genomes being sequenced quickly and cheaply, the need for reference-free SNP calling has increased.

A pipeline for identifying SNPs (and small indels) between closely related genomes, called DIAL (De novo Identification of Alleles), was published in 2010 (Ratan et al., 2010). The main aim of this study was to investigate genetic diversity of endangered species (the pipeline was tested upon Orangutan sequence data), which do not yet necessarily have a reference genome. As part of this pipeline, reads are gathered into "clusters" of similar sequence, and those that are likely to come from repeat regions, or from duplicate reads due to PCR errors or sequencing artifacts (such as poly-A reads), are removed. Then micro-assemblies of these clusters are used to compare between reads and call SNPs, including quality constraints such as variation being present in more than one read, and the putative SNP having at least 40-50 bp flanking it either side.

Another approach utilises coloured de Bruijn graphs for *de novo* assembly and subsequent SNP calling (Iqbal et al., 2012). The authors implement this in their software, called CORTEX, and demonstrate its effectiveness in four different experiments, including calling variants within 10 chimpanzees for which there is no reference sequence, and estimating genotypes for the highly variable human leukocyte antigen gene. The different colours within the graph represent different genomes, allowing multiple genomes to be analysed together in a single graph, resulting in detecting variants without the need for a reference. De Bruijn graphs illustrate different lengths of sequence (k-mers), as nodes, with the edges between the nodes representing k-mers which overlap, and are seen next to each other in the input sequence. Within the graph, variants are visible as bubbles (or more complex structures) within the path of the graph.

Another recent implementation of SNP calling in genomes without a reference used an improved Maximum-Likelihood algorithm (Dou et al., 2012). The authors describe the improved accuracy of their method, which is implemented by eliminating false positives that arise from repetitive regions. To identify SNPs *de novo*, reads are assembled together in read clusters (as in other reference-free

methods such as that in Ratan *et al.* earlier), from which SNPs can be called. However, repetitive regions form "composite clusters", where reads from more than one location are clustered together. The author's method utilises a mixed Poisson model to identify these composite clusters, and removes the repetitive regions from any SNP calling, preventing them from producing "false SNPs".

Traditional SNP calling, and more recent reference-free SNP detection methods, are not designed to look for intragenomic variation between multi-copy genes. In fact, most methods are designed to exclude it. Repetitive sequence is a confounding factor in SNP calling, and variation within the rDNA is ignored or removed. Therefore these programs cannot be used to identify pSNP type variation within repetitive regions, and so although SNP detection in the rDNA region is possible using some of the aforementioned methods, pSNPs would require a different method to be identified.

Instead, a method similar to that demonstrated in James *et al.* must be used. The Python scripts used by James *et al.* (2009) were never released, but the approach they used formed the basis of a pSNP discovery tool called TURNIP, which is introduced in Chapter 2.

1.7. Chapter Summary

This chapter has described the structure of rDNA, how the copy number of the rDNA is dynamic, and current understanding of the mechanisms involved in homogenising the rDNA repeat unit sequence through the process of concerted evolution. rDNA has been widely used in phylogenetics. The recent identification of pSNPs has given us the unique opportunity to utilise these polymorphisms to study rDNA variation, and to begin incorporating them into a model of concerted evolution. The rest of this thesis will discuss work using the TURNIP software to identify rDNA sequence variation within two contrasting yeast species. The thesis goes on to discuss the uncovered variation, and inferences that can be made from it. Initial work on computational simulation of concerted evolution is described, incorporating knowledge gained from the earlier identification of variation. Finally there is a discussion upon the implications of this work, and future directions that could be taken, building upon the findings presented here.

2. Identification of rDNA Variation

Chapter Abstract

This chapter discusses a study to identify variation within the Saccharomyces Genome Resequencing Project dataset using the TURNIP software suite, focussing on the discovery of partial SNP (pSNP) and other polymorphism types in the rDNA genomic region. It describes bespoke scripts written either to identify a broader range of variation within the rDNA than included in previous studies, or to analyse the data in new ways. To ensure accurate identification of variation with TURNIP a number of bugs were removed in TURNIP resulting in a new version being released. Further simulated datasets were generated containing known variation in order to test the TURNIP suite, and the default parameters were updated in accordance with these results. Lastly the SGRP dataset was filtered to remove both sequence contamination and falsely identified polymorphisms. All remaining variation was then assumed to be genuine, and could be analysed further. The results include the first detailed analysis of rDNA variation in *S. paradoxus*, the nearest wild relative of *S. cerevisiae*.

2.1. Background

2.1.1. The Saccharomyces Genome Resequencing Project

The Saccharomyces Genome Resequencing Project (SGRP) is a collaborative project to sequence the genomes of multiple strains of two closely related yeast species: *Saccharomyces cerevisiae* and its wild relative *Saccharomyces paradoxus*. As part of the SGRP project 37 *S. cerevisiae* and 27 *S. paradoxus* strains were sequenced using Sanger sequencing on ABI 3730 DNA sequencers (Liti et al., 2009), to a depth of between 0.42x and 3.92x, resulting in 1.42 million sequence reads. In addition, four *S. cerevisiae* strains and 10 *S. paradoxus* strains were sequenced using Illumina Solexa technology, although the resulting data was not used here.

The SGRP web server, containing downloadable data, documentation on the project, BLAST servers for the two species and a genome browser for each species can be found here (www.sanger.ac.uk/research/projects/genomeinformatics/sgrp.html). A key aim of the SGRP project was to advance understanding of genomic diversity, variation and evolution within these two species. A population genomics paper using data from the project was published in 2009 (Liti et al., 2009). The genome-wide polymorphism data derived from the sequence reads allowed the two populations to be characterised, showing *S. paradoxus* to be split into distinct, well separated geographical populations, as opposed to *S. cerevisiae* which possessed a more closely related, mosaic structure, influenced by human intervention. The SGRP dataset has since been well studied, cited in more than 270 publications (273 as of 2nd May 2013 according to www.scopus.com) with sequencing data from the two closely related species providing insights into many evolutionary and genomics studies.

The different geographical locations, sources and, in the case of *S. cerevisiae*, uses of each strain analysed further here (34 *S. cerevisiae* and 26 *S. paradoxus*) are shown in tables 2.1 and 2.2. Four strains (three *S. cerevisiae* and one *S. paradoxus*) were excluded from this analysis, with details of these strains provided in section 2.4. The majority of the *S. paradoxus* strains originated from exudate from oak trees (genus *Quercus*), whereas *S. cerevisiae* strains covered a more diverse range of habitats, including clinical samples, soil isolates, brewer's and baker's strains.

The rDNA genomic region of the SGRP *S. cerevisiae* strains has been studied previously within the group (James et al., 2009), as discussed in the Chapter 1. The aim here was to analyse the *S. paradoxus* dataset to allow a comparison of polymorphisms between the two species, and also to re-analyse the *S. cerevisiae* data using a new methodology, including use of a suite of programs to more stringently identify sequence variation. Furthermore, this analysis attempted to gain additional insight into possible evolutionary processes involved in the origins of this variation, and to discover whether there are differences between the intra-species variation observed within the wild and domestic strains. Finally, the variation uncovered allows fine-scale phylogenetic inferences to be made between the closely related strains.

The SGRP dataset was ideal for the proposed study. Firstly it contains a large number of closely related strains, enabling fine-scale phylogenetic inferences using only the rDNA region to be demonstrated. Secondly the results can also be compared to the whole-genome results from (Liti et al., 2009). Thirdly, as both

yeast species contain a single rDNA locus, key evolutionary features of the rDNA tandem array could be compared between strains, and findings applied to their population history. This also reduced any confounding factors of homogenisation or recombination between multiple rDNA loci. Furthermore the majority of the data is Sanger sequence data, and so has the longer read-lengths necessary for input to the TURNIP computer program, which was used to identify variation, as discussed in the following section.

Strain*	**Source**	**Geographic location**	**Genome type**	**Lineage[C]**
27361N	Clinical isolate (fecal)	Royal Victoria Infirmary, Newcastle, UK	Mosaic	NA
322134S	Clinical isolate (Throat-sputum)	Royal Victoria Infirmary, Newcastle, UK	Mosaic	NA
378604X	Clinical isolate (Sputum)	Royal Victoria Infirmary, Newcastle, UK	Mosaic	NA
BC187	Barrel fermentation	Napa Valley, USA	Structured	Wine/ European
DBVPG 1106	Grapes	Australia	Structured	Wine/ European
DBVPG 1373	Soil	Netherlands	Structured	Wine/ European
DBVPG 1788	Soil	Turku, Finland	Structured	Wine/ European
DBVPG 1853	White Teff	Ethiopia	Mosaic	NA
DBVPG 6040	Fermenting fruit juice	Netherlands	Mosaic	NA
DBVPG 6044	Bili wine, from *Osbeckia grandiflora*	West Africa	Structured	West African
DBVPG 6765	Unknown	Unknown	Structured	Wine/ European
K11	Shochu sake strain	Japan	Structured	Sake
L_1374	Fermentation from must Pais	Cauquenes, Chile	Structured	Wine/ European
NCYC 110	Ginger beer from *Z. officinale*	West Africa	Structured	West African
NCYC 361	Beer spoilage strain from wort	Ireland	Mosaic	NA

Strain	Source	Geographic location	Genome type	Lineage[C]
S288c[A,B]	Rotting fig	Merced, California, USA	Mosaic	NA
SK1[B]	Soil	USA	Mosaic	NA
UWOPS03-461-4	Nectar, Bertram palm	Telok Senangin, Malaysia	Structured	Malaysian
UWOPS05-217-3	Nectar, Bertram palm	Telok Senangin, Malaysia	Structured	Malaysian
UWOPS05-227-2	Stingless bee (*Trigona* sp.)	Telok Senangin, Malaysia	Structured	Malaysian
UWOPS83-787-3	Fruit, *Opuntia stricta*	Great Inagua Island, Bahamas	Mosaic	NA
UWOPS87-2421	Cladode, *Opuntia megacantha*	Puhelu Road, Maui, Hawaii	Mosaic	NA
W303[B]	Laboratory generated	NA	Mosaic	NA
Y12	Palm wine strain	Ivory Coast	Structured	Sake
Y55[B]	Grape	France	Mosaic	NA
Y9	Ragi (similar to sake wine)	Indonesia	Structured	Sake
YIIc17_E5	Wine	Sauternes, France	Mosaic	NA
YJM975	Vaginal isolate from patient with vaginitis	Ospedali Riuniti di Bergamo, Italy	Structured	Wine/ European
YJM978	Vaginal isolate from patient with vaginitis	Ospedali Riuniti di Bergamo, Italy	Structured	Wine/ European
YJM981	Vaginal isolate from patient with vaginitis	Ospedali Riuniti di Bergamo, Italy	Structured	Wine/ European
YPS128	Soil beneath *Quercus alba*	Pennsylvania, USA	Structured	North American
YPS606	Bark of *Q. rubra*	Pennsylvania, USA	Structured	North American
YS4	Baker's strain	Netherlands	Mosaic	NA
YS9	Baker's strain	Singapore	Mosaic	NA

Table 2.1.: *S. cerevisiae* strain information, including source, geographical location, genome type and lineage, compiled by Dr Steve James. *S. cerevisiae* [A]Reference strain; [B]Laboratory strain; [C]Classification according to (Liti et al., 2009).

Strain	Source	Geographic location	Population
A4	Bark of *Quercus rubra*	Mont St-Hilaire, Quebec, Canada	American
A12	Soil beneath *Q. rubra*	Mont St-Hilaire, Quebec, Canada	American
CBS 432[A,NT]	Bark of *Quercus* sp.	Moscow area, Russia	European
CBS 5829	Mor soil (pH 3.6)	Denmark	European
DBVPG 4650	Fossilized guano in a cavern	Marche, Italy	European
DBVPG 6304	*Drosophila pseudoobscura*	Yosemite, California, USA	American
IFO 1804	Bark of *Quercus* sp.	Japan	Far Eastern
KPN 3828	Bark of *Q. rubra*	Novosibirsk, Siberia, Russia	European
KPN 3829	Bark of *Q. rubra*	Novosibirsk, Siberia, Russia	European
N-17	Exudate of *Q. robur*	Tatarstan, Russia	European
N-43	Exudate of *Q. mongolica*	Vladivostok, Russia	Far Eastern
N-44	Exudate of *Q. mongolica*	Terney, Russia	Far Eastern
N-45	Exudate of *Q. mongolica*	Terney, Russia	Far Eastern
Q32.3	Bark of *Quercus* sp.	Windsor Great Park, UK	European
Q59.1	Bark of *Quercus* sp.	Windsor Great Park, UK	European
Q62.5	Bark of *Quercus* sp.	Windsor Great Park, UK	European
Q89.8	Bark of *Quercus* sp.	Windsor Great Park, UK	European
Q95.3	Bark of *Quercus* sp.	Windsor Great Park, UK	European
S36.7	Bark of *Quercus* sp.	Silwood Park, UK	European
T21.4	Bark of *Quercus* sp.	Silwood Park, UK	European
UFRJ 50791	*Drosophila* sp.	Catalao Point, Rio de Janeiro, Brazil	American
UFRJ 50816	*Drosophila* sp.	Tijuca Forest, Rio de Janeiro, Brazil	American
Y6.5	Bark of *Quercus* sp.	Silwood Park, UK	European
Y7.2	Bark of *Quercus* sp.	Silwood Park, UK	European
YPS138	Soil beneath *Q. velutina*	Pennsylvania, USA	American
Z1.1	Bark of *Quercus* sp.	Silwood Park, UK	European

Table 2.2.: *S. paradoxus* strain information, including the source and geographical location of each strain, compiled by Dr Steve James. *S. paradoxus* [A]Reference strain; [NT]Neotype strain.

2.1.2. The TURNIP Software Suite

Due to their recent discovery, there is only one piece of software currently publicly available for the discovery of partial Single Nucleotide Polymorphisms (pSNPs): TURNIP (or **T**racking **U**n**R**esolved **N**ucleot**I**de **P**olymorphisms). This suite of software, written in Perl, identifies micro-variation in hard-to-assemble repetitive DNA sequences such as rDNA. It is available online from the NCYC website (`www.ncyc.co.uk`), and is explained further in a 2010 publication (Davey et al., 2010).

TURNIP carries out the steps necessary for pSNP discovery (for an overview see figure 2.1) using similar principals to the method discussed in the James *et al.* paper (James et al., 2009), and described briefly here in Chapter 1. This previous method analysed DNA sequence data obtained from Sanger sequencing (which results in longer reads than the current next-generation sequencing platforms)using multiple alignment methods to align sequence reads to a consensus rDNA unit, and scoring pSNPs and SNPs from the resulting alignments. However, TURNIP makes several improvements over this earlier method, including the ability to resolve features such as indels of varying lengths and poly-A tracts, in addition to identification of the desired SNPs and pSNPs.

Another program which examines heterogeneity within repetitive regions is DNP Trapper (Arner et al., 2006), based on the TRAP algorithm (Tammi et al., 2003). Although like TURNIP it uses multiple sequence alignments in the assembly method, the main aim of this program is assembly rather than variation discovery. However, DNPTrapper would be unlikely to be suitable for the analysis of highly repetitive rDNA regions with high copy number (it has only been used for relatively short repetitive regions so far), and for discovery of pSNPs. This is because DNP Trapper requires assembly of the entire genomic region undergoing analysis which is currently infeasible for the highly repetitive rDNA. As TURNIP does not require sequence assembly prior to analysis it is the only currently available program to identify pSNPs in rDNA.

A summary of the steps involved in identifying variation using TURNIP are shown in figure 2.1, but a more detailed description is given below:

1. **Input** - FASTQ files obtained from whole or partial genome sequencing projects that have been split into FASTA and quality score files (e.g. Phred

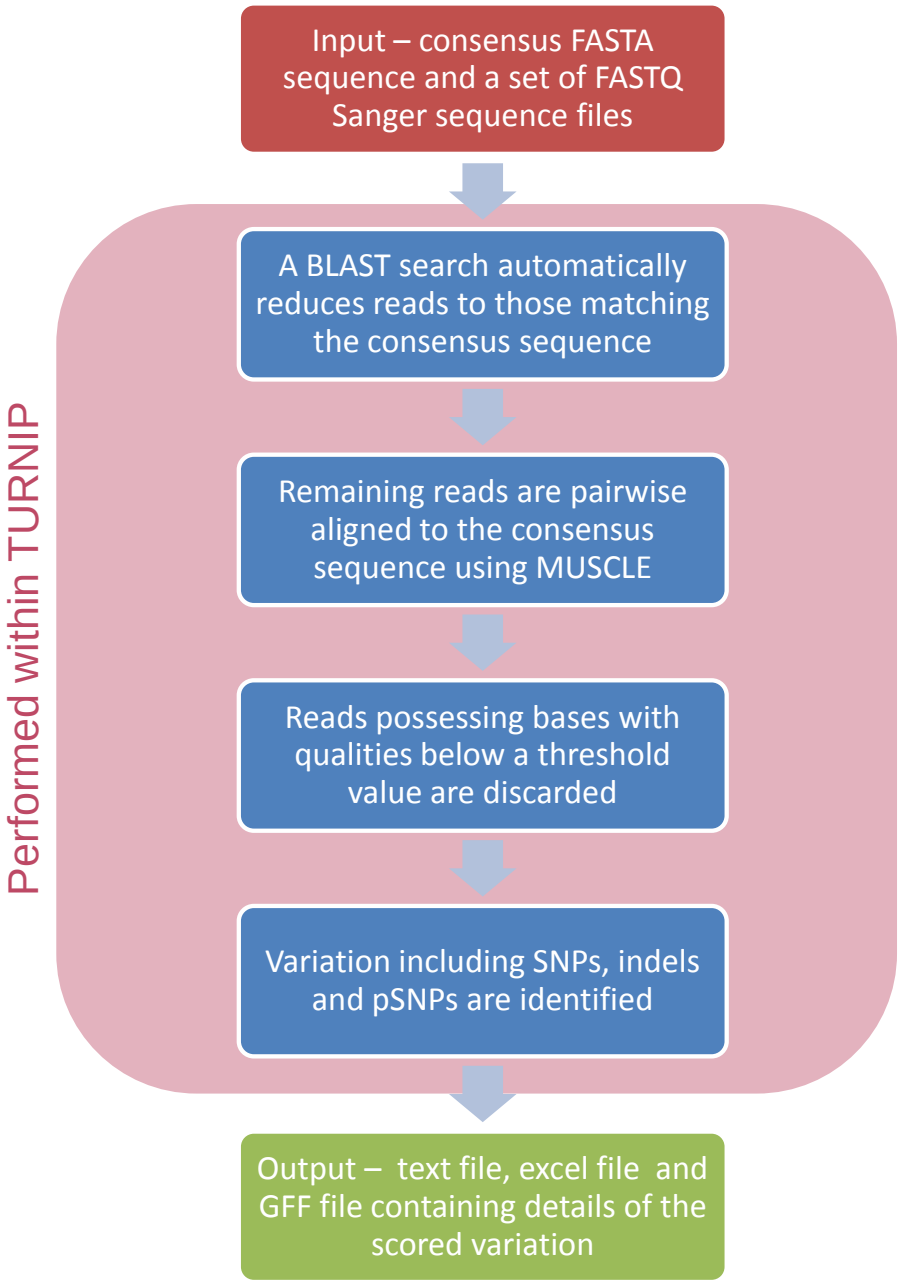

Figure 2.1.: Overview of the flow of data through the TURNIP suite

or Solexa depending on sequencing method used) for each genome of interest (in our case yeast strains), plus a consensus sequence with which to compare the genomic sequence reads (in our case a reference rDNA sequence). The names of the files containing these input data are specified in a configuration (or conf) file.

2. **Sequence reads matching the consensus sequence are identified using BLAST** - rDNA specific sequences, a subset of the genomic sequence reads, are found by a BLAST sequence similarity search (Altschul et al., 1997) of the input FASTA files against the supplied rDNA consensus sequence.
3. **High-scoring reads split into 100-mers and reblasted** - All sequence reads that align to the consensus sequence are temporarily stored. The high scoring reads are split into 100-mers (maximum length, with a 20bp sliding window flanked on either side with a region between 10bp and 40bp i.e. the first read will be 20 + 40 flanking on one side equalling a 60-mer). A representation of this process is shown in figure 2.2a, where blue boxes represent the central 20 bases, which is the region of interest. A less stringent BLAST is performed.
4. **Reads with 100% identity to consensus are discarded** - In this optimization step, sequence reads with 100% identity to the consensus sequence are discarded as the program is only looking for variation within the repetitive region, not for the full assembly. Only distinct high-scoring pairs are needed. However, their presence within the dataset is recorded as this information may be needed at a later point in the analysis when estimating polymorphism frequencies across the read set.
5. **Gapped multiple alignment** - a gapped multiple alignment using MUSCLE (Edgar, 2004) aligns all reads in each 20bp window to the consensus sequence. This identifies insertions and deletions (indels).
6. **20-mers with low quality bases are discarded** - if any 20-mers contain one or more bases associated with a quality score below a given threshold (for gaps this is the average score surrounding the gap), they are removed from the analysis. This process is shown in figure 2.2b.
7. **Variation is called** - The remaining 20-mers are stacked in a multiple sequence alignment, compared to the consensus and called for variation i.e. indels, SNPs or pSNPs. An example of each type of variation in a multiple sequence alignment is shown in figure 2.2c.
8. **Output** - The output is stored in txt, Microsoft Excel, SQL and GFF files. The location, type and frequency of variation is recorded.

Output files are written at each stage to enable simple and efficient repetition of

the process if only parameters (rather than data) are changed, or if the process is interrupted mid-way. Parts of the program can be run concurrently on a multi-core or cluster environment.

TURNIP is the first program to deal effectively with partially resolved SNPs, and to overcome problems inherent to highly repetitive regions such as the rDNA. Therefore it is the logical choice of software for our study.

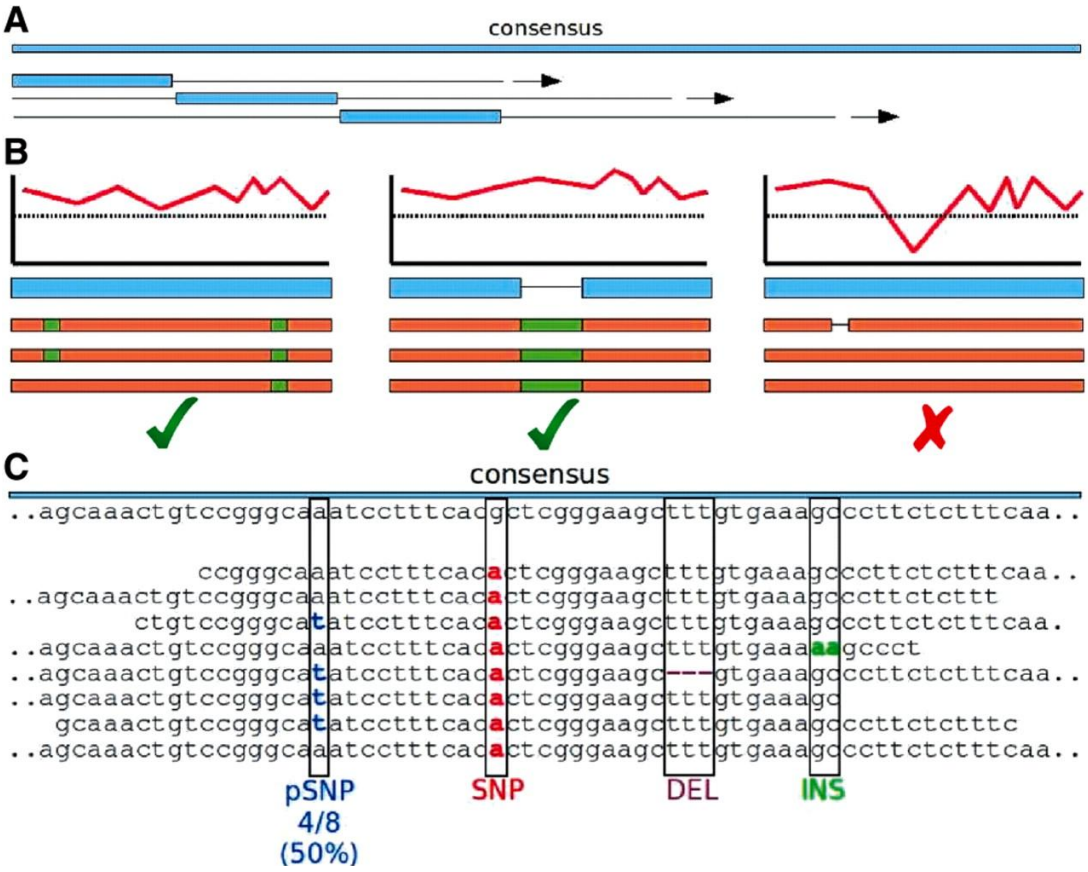

Figure 2.2.: Overview of the workings of the TURNIP suite (A) Sliding window approach, depicting the central 20mer region anchored by longer flanking regions. (B) Seed read filtering procedures employed whereby quality scores are checked across each 20mer and rejected if any drop below a given threshold. (C) Stacking of reads that align to a single copy consensus to ascertain SNP, indel and partial SNP (pSNP) variation. Variation is discarded if it is only resolved in a single read per 20mer window, e.g. the insertion and deletion would both be discarded here. Reproduced with permission from the lead author, (Davey et al., 2010) and by permission of Oxford University Press.

2.2. Preliminary Analysis using TURNIP: Bug Fixing

Identification of polymorphisms (comprising SNPs, pSNPs, insertions and deletions) within the rDNA sequence of the SGRP strains was made using the TURNIP suite of software (Davey et al., 2010), described in the previous section (The TURNIP Software Suite). Installation was performed according to instructions on the website (www.ncyc.co.uk/turnip/turnip-howto.html), and the following setup was used:

- TURNIP version 1.2_20100818
- BioPerl version 1.6.1
- Perl version 5.10.1 Modules:
  - Benchmark
  - Data::Dumper
  - List::Util
  - Parallel::ForkManager
  - Set::Scalar
  - Spreadsheet::WriteExcel
- BLAST version 2.2.24 (note this is a legacy version, not BLAST+)
- ImageMagick 6.5.1

All of the analyses and programs were run on a desktop PC with an Intel Core 2 Duo 3.16 GHz processor and 4Gb RAM, running the Linux Fedora 11 operating system.

The raw sequencing reads for *S. cerevisiae* and *S. paradoxus* were downloaded from the SGRP site and formatted using the **process_fastq.pl** script in TURNIP. This script splits the FASTQ files (containing the combined sequence and quality scores for each nucleotide along the read) for each strain into .fasta files (sequence only) and .qual quality files (PHRED scores only) for subsequent use in TURNIP with the BioPerl SeqIO module.

In order to run the resulting .fasta sequence files through BLAST (and through TURNIP which calls BLAST), the fasta files were first formatted using **formatdb**, with the command `# formatdb -i <filename> -o T -p F`, where the -o flag relates to parsing the sequence id, and -p relates to whether or not the sequence is nucleotide or protein.

A preliminary TURNIP run was carried out upon the *S. cerevisiae* data analysed in a previous study (James et al., 2009). Prior to analysis, the dataset had been clipped to form an rDNA-only database (i.e. all non-rDNA sequence reads had been removed) using filters described in the methods section of that article. Analysis of this dataset enabled a comparison to be made between the variation identified by TURNIP and the published variation identified by a collaborator's software for the same dataset. The run both identified various bugs in the TURNIP software and discrepancies with the results of the previous study that were investigated further.

2.2.1. TURNIP Bug 1:Parsing file names

Files with underscores in their names could not be parsed in TURNIP and their subsequent analysis failed. The regular expression in **Hitseries.pm** was changed to the following `[$aname =~ m/(\d+)\_ascriptions\.[n]?dat/]` to correctly parse the file name.

2.2.2. TURNIP Bug 2: Memory Leak

A memory leak in TURNIP was identified. For initial runs of TURNIP the *S. cerevisiae* dataset, which comprised 34 strains, was used. Here the first 3-4 strains were processed relatively quickly (a few minutes per strain). However, despite both CPUs working at 100% capacity, the software was still running 16 hours later. To address this problem, bash scripts were written to call TURNIP multiple times with different conf files (i.e. configuration files which specify which strains to run through TURNIP, and the parameters to be used), with each conf file specifying a small number of strains (1, 3 or 11 strains specified per conf file). The results of the runtimes for these runs are shown in figure 2.3, as the blue series. A run was cancelled for the case of one conf file containing all 34 strains as they had not completed after running over a weekend, with the CPUs working at full capacity. This suggested a memory leak in the TURNIP software. The output files from TURNIP from each stage of the program were compared, for the same strain run in different positions in the conf files, using the **diff** command in Linux. This showed the HitSeries.out files were different for the same strain when it was run at varying positions in the strain list, suggesting the part of the program producing this file (or earlier) contained an error. After looking through

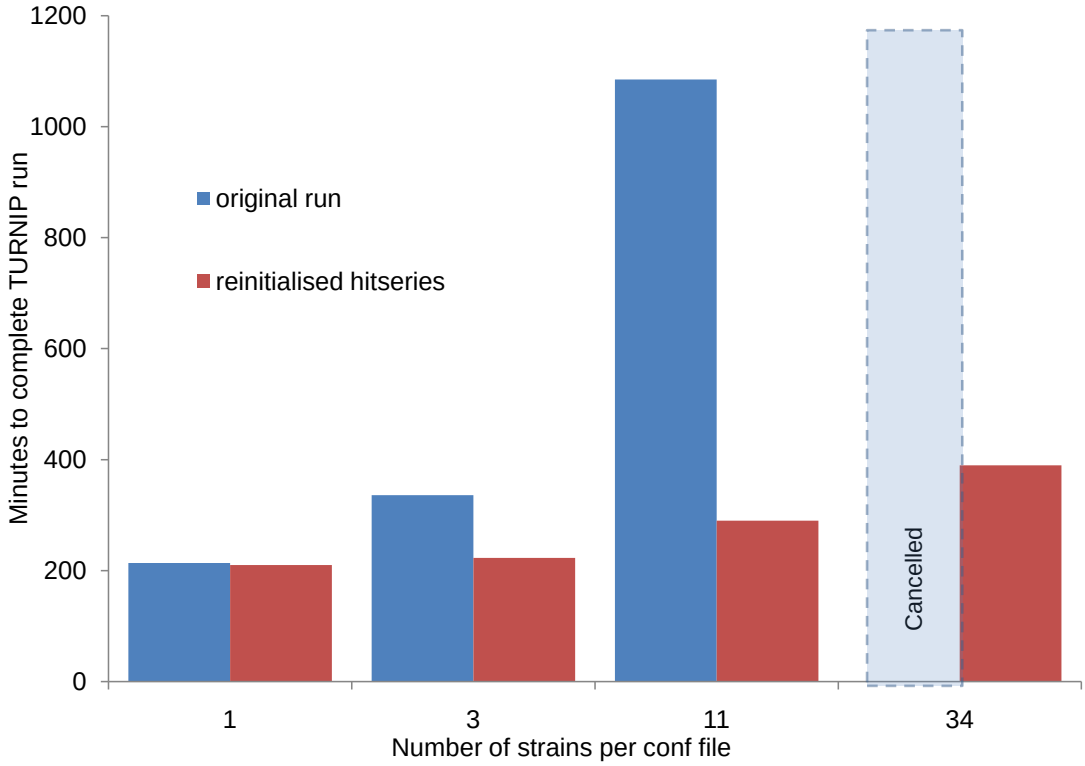

Figure 2.3.: Comparison of TURNIP run times with different numbers of strains in each conf file, before and after flushing the hit_series array. The original run with 34 strains in a conf file was cancelled after running for 2 days, with the run not yet completed.

the code, it appeared that the array hit_series was not being reinitialised (flushed) between analysis of different strains in a single TURNIP run. A line of code was added to reinitialise this array each time a new strain was analysed, and the runs were repeated (with 1, 11 and 34 strains specified per conf file), shown as the red series in figure 2.3. Following the memory leak correction there was a large reduction in the time to run 34 and 11 strains in TURNIP.

2.2.3. Inconsistency in Identifying Variation In Strains

Upon examination of the results of TURNIP runs with differing numbers of strains specified per conf file, as noted in section 2.2.2, some differences were noted between identified polymorphisms in each strain, varying with the order in which the strain was run. This indicated some variation was being inaccurately recorded. An example of such differences is shown in table 2.3. After the fix of reinitialising the hit_series array mentioned in section 2.2.2, the number of polymorphisms identified in each strain were then consistent between different TURNIP runs

(i.e. whether it was the first strain in the list to be run through TURNIP, or the 3rd or the 11th). Other arrays and hashes within the TURNIP code were also reinitialised but no further differences to the runtimes or results were observed. Therefore reinitialising the hit_series array was shown to fix a major part of the memory leak, and to eradicate problems of polymorphism identification due to strain order.

	1 strain	**3 strains**	**11 strains**	**x strains (after fix)**
pSNP	30	27	33	30
SNP	39	37	42	39

Table 2.3.: Number of pSNPs and SNPs identified by TURNIP 1.2 in *Saccharomyces cerevisiae* strain YS4. Strain YS4 was the last strain to be run in each file. Values differ between analysis order of strains through TURNIP, but are consistent after TURNIP fix (for 1, 11 or 34 strains per conf file)

2.3. Validating TURNIP Output

2.3.1. The Problem

As has just been seen, the SGRP *S.cerevisiae* dataset was analysed with TURNIP in order to validate its installation and to act as a preliminary test of the program on a carefully chosen dataset. During this analysis, discrepancies between results of runs with different numbers of strains were noted. The underlying software bugs, identified here, were subsequently fixed in a later version of the program (version 1.3_20110323). However, this process highlighted the need for a systematic and easy way to validate the results obtained from TURNIP (Davey et al., 2010), and to provide a quality check for any future versions of the program.

A plan was made to develop a computer script to generate simulated input datasets for TURNIP, with a known number and position of SNPs and pSNPs (of a known occupancy). Such a script could provide the means by which the results from a TURNIP run could be subsequently validated.

2.3.2. Overview of the Script to Simulate TURNIP Datasets

A decision was made to develop the script in the Perl programming language, which has a number of advantages for use in this setting. Firstly, generating variation in sequence reads is a form of text string manipulation, something that is done simply and efficiently in Perl. Secondly, TURNIP is written in Perl, so the validation script could easily become part of a future TURNIP version.

The Perl script needed to generate a specified number of reads, with a known number and position of SNPs and pSNPs from a supplied consensus sequence, to then be run through TURNIP. The TURNIP results could then be compared to the known values and the accuracy of the TURNIP output assessed.

The script required certain pre-defined parameters that could be changed between different experimental runs. These include:

- A consensus sequence
- The desired read length
- The coverage
- Number of repeats
- Number of pSNPs to generate
- Number of SNPs to generate

The number of reads that need to be generated to simulate this coverage can be easily calculated using the coverage calculation of Lander and Waterman (Lander and Waterman, 1988). The calculation is shown below, where N is the number of reads to generate, C is the coverage, r is the rDNA repeat number, G is the consensus length and L is the read length.

$$N = \frac{CGr}{L} \tag{2.1}$$

The script is laid out in a number of subroutines to aid ease of reading for certain tasks. The subroutine name and a brief overview of the code is summarised in table 2.4.

Running the script, called generate_data_v10.pl results in all of the input files required for a TURNIP run (a fasta file of reads, a corresponding quality file, and

a BLAST formatted database of the reads. It also generates a summary text file containing the known positions, bases involved and reads changed for each SNP and pSNP generated. This summary gives the expected results that would be seen if TURNIP is wholly accurate in its analysis.

Subroutine	Overview
read_in_consensus	Reads in the consensus sequence from a named file into a string, removing any whitespace and adding a tail of N's (tail = half a read length of N's) to each flank.
calculate_reads	Calculates the number of reads to generate given the input parameters using equation 2.1. If a number of reads is specified, do not calculate.
generate_reads	Generates the reads. Each read is represented as an array, containing an integer referring to its start position within the consensus sequence, and the sequence of the read (generated by a substring method from the start position using the consensus sequence). Each read is stored in an array, creating an array of arrays (reads).
run_polymorphisms	Generates an array of random positions in the consensus sequence, one for each SNP requested, and another array with positions for pSNPs. Another array is generated for pSNP position occupancies. This then calls the generate_SNP and generate_pSNP subroutines, once for each SNP and pSNP.
generate_SNP	Compares the SNP position to the starting position of each read generated. If the starting position results in a read which will contain the SNP, the relevant base within the read string is substituted for the SNP, and the read details added to a SNP summary array.
generate_pSNP	Compares the pSNP position to the starting position of each read generated. If the starting position results in a read which will contain the pSNP, the read is copied to a possible read array. Depending on the occupancy, a subset of reads in this array are chosen to have their relevant base within the read string substituted for the pSNP. The occupancy is then readjusted to be the observed occupancy, and the read details added to a pSNP summary array.
adjust_positions	After the SNPs and pSNPs are generated, this method adjusts the positions to be those of the original consensus, not the position of the pSNPs and SNPs in the consensus plus the tails. Needed for correct calling of SNP and pSNP positions in the output.
write_summary	Writes all of the summary array information to file. This includes the SNP and pSNP positions and occupancies, plus the identities of the reads involved. Also includes a header of the parameters used in generating the file.
write_reads	Writes all of the generated reads into a fasta formatted file, and a corresponding quality file with a generated quality score, for each position in each read.
format_fasta	Runs **formatdb** on the generated fasta file to get all of the files needed for input into TURNIP

Table 2.4.: Summary of the subroutines within generate_data.pl

2.3.3. Overview of Script to Compare Generated Data to TURNIP output

After running the generate_data_v10.pl script to generate reads, and running the reads through TURNIP, it quickly became clear that a simple and systematic method to compare the output of TURNIP to the summary file from the generated data would be advantageous. This again was a Perl script, which takes the summary text file from generate_data_v10.pl and the pSNP table text file from TURNIP as input. It then compares the position of each SNP and pSNP, and the occupancy of the pSNPs, and outputs an .xls file highlighting those instances where the two results are in disagreement, or if the percentage difference between the pSNP occupancies is above a certain threshold.

The script is called compare_files_v8.pl, as it compares the two sets of output files. It uses the Spreadsheet::WriteExcel Perl module, similarly to the pSNP_table_to_excel.pl script in TURNIP. The threshold at which the difference in pSNP occupancy becomes highlighted can be changed, to aid detection of more divergent results. Part of a small screenshot of a typical .xls output is shown in figure 2.4.

	A	B	C	D	E	F	G
1	Comparing SNPs and pSNPs in TURNIP summary to generated summary: Thu Jun 9 10:37:42 2011						
2							
3	Input files: SpDBVPG4-pSNP_table_summary.txt and SpDBVPG4 summary.txt.						
4							
5	Expected pSNPs:	5					
6	TURNIP pSNPs:	5					
7	Expected SNPs:	2					
8	TURNIP SNPs:	2					
9							
10	SNP/pSNP not scored in both	0					
11	Occupancies > 1% different	1					
12							
13	Position	Consensus	Target	% Occupancy (TURNIP)	% Occupancy (Generated)	Occupancy difference	Type
14	695	c	a	64.9484536082	63.7	1.2484536082	pSNP
15	2410	t	a	100	100	0	SNP
16	2718	t	a	84.8591549296	84.9	-0.0408450704	pSNP
17	2965	g	a	100	100	0	SNP
18	4711	t	c	54.3554006969	53.7	0.6554006969	pSNP
19	5014	g	c	10.9540636042	10.9	0.0540636042	pSNP
20	8482	g	a	18.315018315	18.9	-0.584981685	pSNP

Figure 2.4.: A screenshot from an example output .xls format file from the compare_files_v8.pl script, comparing the generated data summary to the results from the TURNIP run on this data. A pSNP at position 695 shows a greater than 1% difference in occupancy from the expected.

2.3.4. Validating TURNIP

To test the generate_data_v10.pl and compare_files_v8.pl Perl scripts, and to validate the chosen version of TURNIP, a series of experiments were undertaken. A number of datasets were generated from the *S. cerevisiae* and *S. paradoxus* consensus sequences, with the number of SNPs and pSNPs chosen as being similar to those from known strains. Strains from *S. cerevisiae* and *S. paradoxus* with low, average and high numbers of pSNPs and SNPs were chosen as examples to model realistic numbers for generated strains. A further two generated strains had an average number of pSNPs and SNPs, but had a smaller and larger read length respectively, similar to the outer limits of read lengths expected for Sanger reads. The details of these experiments are shown in table 2.5.

Name	**Strain based upon**	**No. pSNPs**	**No. SNPs**	**Read length-/bp**	**Consensus used**
ScUW83	UWOP83_787_3	37	9	800	*S.cerevisiae*
ScSpDB	SpDB44	10	14	800	*S.cerevisiae*
ScYJM975	YJM975	4	6	800	*S.cerevisiae*
SpUWOP	UWOPS91_917_9	345	57	800	*S. paradoxus*
SpKPN3829	KPN3829	16	6	800	*S. paradoxus*
SpDBVPG	DBVPG4650	5	2	800	*S. paradoxus*
ScRead400	n/a	12	9	400	*S.cerevisiae*
ScRead1000	n/a	12	9	1000	*S.cerevisiae*

Table 2.5.: Summary of the parameters used to generate data for the experimental runs. Name is the filename of the files with the stated parameters generated to run through TURNIP. The parameters used to generate the data for each file are shown in subsequent columns. The number of pSNPs and SNPs specified are the same as the strain the run is based on, except ScRead400 and ScRead1000, where the pSNP and SNP numbers are based upon the average number of each polymorphism within all of the SGRP *S. cerevisiae* strains.

Each of the generated data "strains" was created using the generate_data_v10.pl script, and run through TURNIP using default parameters. The "pSNP_table_summary.txt" TURNIP output files from each strain were compared to the summary files from the generated data using the compare_files_v8.pl script producing an excel file for each dataset. No differences between generated and estimated numbers of SNPs were identified in any of the eight datasets analysed. However, some differences were observed for pSNPs, notably in occupancy values.

For many positions at which there was a greater than 1% difference in pSNP occupancy, inspection of the TURNIP output files showed that they were covered by the maximum permitted (according to default TURNIP settings) number of sequence reads (250). A tandem array with 140 repeats, sequenced to a depth of 2x, might be expected to produce 280 (2 x 140) reads, and therefore an alignment of depth 280 for variation discovery. By limiting alignment depth to 250, pSNP occupancy frequencies may have been distorted. To quantify the strength of this distortion, if indeed one exists, a second TURNIP run was performed on each generated strain using non-default BLAST parameters. To do this, a line was added to the BlastFactory.pm module of TURNIP, which set the parameters **b** (the number of database sequences with HSPs to the query) and **v** (the number of one line descriptions of database sequences) to be 800. This run is referred to as TURNIP 2, whereas the default parameter run is TURNIP 1.

2.3.5. Results and Discussion

The Microsoft Excel output files for each generated strain, for each run, were gathered into a single directory. The occupancies for each run from the generated summary file, together with TURNIP 1 and TURNIP 2 results, were accumulated into one spreadsheet, so that the differences between the runs could be amalgamated into one dataset. The results of these runs are shown in figure 2.5.

The bar chart in figure 2.5 is similar to graphs produced in a previous analysis of the SGRP *S. cerevisiae* data (James et al., 2009), where the pSNP occupancies for each strain were placed into frequency bins. Comparing the generated data to the two different result sets, shown in blue, red and yellow in the figure, there are differences in the number of polymorphisms per bin between at least two of the categories in all but the 100% bin, which represents the SNPs, and the 20 - 29.9% bin. In general, the TURNIP 2 run, which does not use the default parameters for the maximum number of reads aligned to the consensus sequence, is more similar to the generated data, with regard to the estimated number and occupancy of pSNPs, than the default TURNIP 1 run, with all of the bins being within 3 pSNPs of the expected number. As well as the two bins where both runs identified all pSNPs, there are 5 cases in which the TURNIP 2 parameters were only 1 value different to the expected, as opposed to only one case where TURNIP 1 was. The default TURNIP run estimates are within a few pSNPs

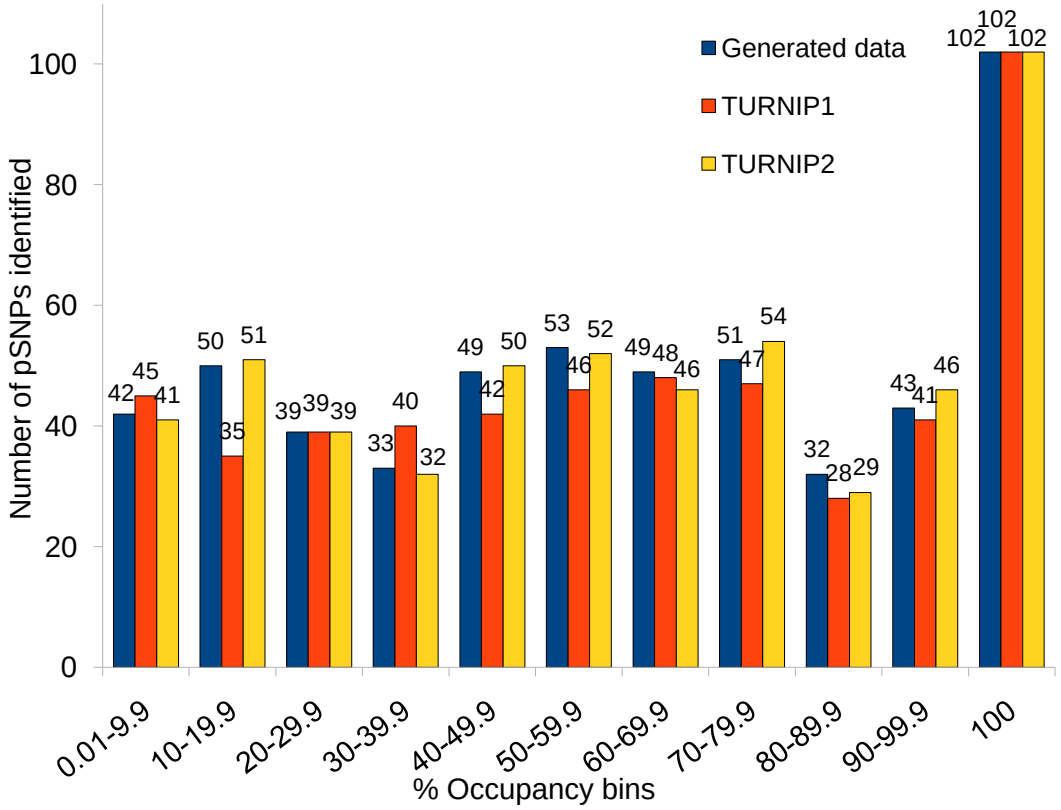

Figure 2.5.: Overall number of pSNPs and SNPs (the latter with 100% occupancy) in different percentage occupancy bins for the generated data (blue), default TURNIP output (TURNIP 1 - red) and TURNIP with different BLAST values (TURNIP 2 - yellow).

identified to the expected values, but a few bins show a larger difference. For example the 10-19.9% bin has a large difference of 15 pSNPs from the expected (35 estimated when 50 expected), but there are also three more cases where there is a 7 pSNP difference (for bins 30-39, 40-49 and 50-59). The large difference between TURNIP 1 and TURNIP 2 in identifying pSNPs in the 10-19.9% bin, could be explained by lower occupancy pSNPs being present in fewer reads, and therefore they may not be represented in the reads when the alignment depth is limited in the default BLAST parameters.

In theory, the TURNIP 2 run should give identical results to the expected values, as all of the read alignments are used for variation discovery. To investigate the observed differences between real and estimated pSNP occupancies more carefully, the TURNIP 1 and TURNIP 2 runs were compared to the generated data on an individual polymorphism basis. The differences in their occupancies to the expected values are summarised in figure 2.6. This figure illustrates more clearly the effect the differences between the default BLAST parameters has on the

estimated occupancies. In the TURNIP 1 results, although a large proportion (approximately 40%) of the calls are within 1% of the expected occupancy, a large number that show greater differences remain, with 40 calls differing by 10% or greater. In comparison, although TURNIP 2 still has differences to the expected occupancies from the generated data, 528 calls (approximately 97%) are within 1% of the expected values, with the remaining 13 being within 2%.

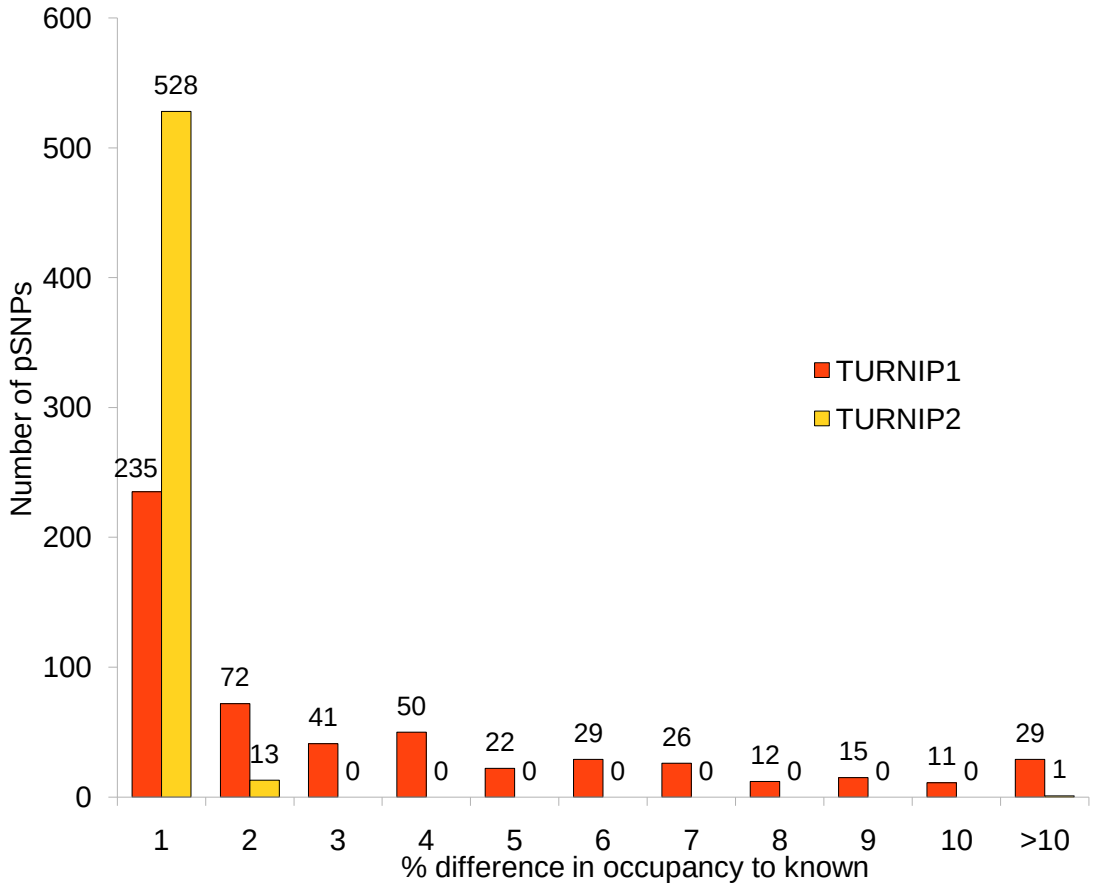

Figure 2.6.: The percentage difference between the expected generated data to default TURNIP output (TURNIP 1 - red), and to non-default BLAST parameter TURNIP output (TURNIP 2 - yellow). TURNIP 2 results still differ from the generated data but the majority are within 1% occupancy of the expected values. TURNIP 1 output has a large number of pSNPs within 1%, but still has some pSNPs with a difference of over 10%.

With the majority of the TURNIP 2 results being within 1% of the expected values, it would be imagined that there would be fewer differences between its results and those of the generated data when put into bins, as shown in figure 2.5. The differences between the number of pSNPs called in this run and those expected could therefore arise when the 1% change in occupancy resulted in the pSNP being reclassified into the bin above or below, when the expected occupancy

is near the bin boundary.

These results illustrate that TURNIP , with carefully chosen parameter settings, is correctly identifying variation, as the SNPs and pSNPs are called in the positions expected, and at very similar occupancies to those expected. The experimental run has also highlighted that the default BLAST parameters used in TURNIP could be changed to slightly improve the accuracy of the pSNP occupancy calling, and may allow inclusion of a few more low occupancy pSNPs. However, although the difference in accuracy of results between the default and altered BLAST parameter runs is notable, small inconsistencies can still occur in the non-default BLAST runs when compared to the expected values.

2.4. Secondary Analysis using TURNIP: Identifying Contaminated Data

After the successful bug fixes and validation testing of TURNIP, the SGRP datasets were run through the new version of TURNIP (version 1.3_20110323), with the BLAST parameters changed to those noted in the previous section (TURNIP 2) to enable more accurate pSNP calling. In addition, the SGRP data was downloaded and the full, unclipped dataset (i.e. all reads, not only those assigned to the rDNA region) used for analysis with TURNIP. The unclipped dataset provides a comparison to the filtered dataset, and highlights any effects of pre-filtering. For example, TURNIP does not require data to be filtered before use, so it will be helpful to assess the effect of doing so on variation discovery.

A number of SGRP strains were not included in these, and subsequent, analyses. In *S. cerevisiae* strain YGPM was excluded due to reasons provided in the SGRP handbook, as it had an unknown origin and odd characteristics including unusual read lengths and quality scores. In addition, strains YS2 and L_1528 were excluded from our analyses, and from the James *et al* analysis (James et al., 2009), due to previously discovered contamination of the sequence reads (personal communication with Dr James). In L_1528 this was visible in the results of a TURNIP run as a few reads consistently possessed variation across the rDNA, which could be identified as *S. paradoxus* contamination. In *S. paradoxus*, 26 of the 27 SGRP ABI sequenced strains were analysed, with strain UWOPS91-917.1, a Hawaiian strain, excluded due to contamination. In this case the contamination

was believed to have originated from an *S. cerevisiae* strain, making removal difficult due to many similarities between the rDNA of the two species. This strain exhibited a large amount of variation when run through TURNIP(304 pSNPs, 60 SNPs and 61 indels), which suggested a large quantity of contaminated reads was present.

Additionally, to increase confidence in the final results, the pSNP calls were manually checked and were only included if more than 2 reads were involved in a polymorphism identification. Therefore if only one variant read was identified, the corresponding polymorphism was removed from the results, as the likelihood that it was derived from sequencing error was deemed to be high.

2.4.1. TURNIP results

Sequence reads for the remaining 34 *S. cerevisiae* and 26 *S. paradoxus* strains were run through the newest version of TURNIP (version 1.3_20110323). Three analyses were performed on the *S. cerevisiae* reads, and two on the *S. paradoxus* dataset. The first run in *S. cerevisiae* used the clipped, pre-filtered SGRP dataset used in the James *et al* analysis (James et al., 2009), with the aim that our results could be directly compared to this previous work. The second run was also performed on the clipped dataset but with the default BLAST parameters in TURNIP changed to have parameter B, the maximum number of reads aligned to the consensus sequence at each position, set to 800, as suggested by the validation results in the previous section. Lastly the unclipped sequence data were run through TURNIP with the improved BLAST parameters. The results of these three runs are shown in table 2.6. The *S. paradoxus* initial run used the default TURNIP BLAST parameters, and the second run had parameter B changed to 800, a summary of the results being shown in table 2.7. Clipped datasets for *S. paradoxus* were not available from the SGRP site.

In *S. cerevisiae*, very few differences between run 1 and run 2a were observed, where BLAST parameters were altered. However, far more variation was predicted when the unclipped dataset was analysed, presumably due to the presence of poor, non-rDNA matches to the consensus sequence. This was particularly noticeable for pSNPs, where the total number identified increased by 176 between runs 2a and 2b. This contrasted with the *S. paradoxus* data where large differences in polymorphism counts could be attributed to changes to the BLAST parameters.

In this case, more pSNPs and INDELS were identified when the parameters were changed, particularly for pSNPs where the number identified increased from 973 to 1351.

Strain	Run 1 pSNP	Run 2a pSNP	Run 2b pSNP	Run 1 SNP	Run 2a SNP	Run 2b SNP	Run 1 INS	Run 2a INS	Run 2b INS	Run 1 DEL	Run 2a DEL	Run 2b DEL
273614X	25	25	31	4	4	4	3	3	6	19	19	20
322134S	15	15	14	5	5	5	3	3	3	17	17	13
378604X	20	20	28	0	0	0	4	4	4	20	20	21
BC187	7	7	8	7	7	7	1	1	1	17	17	17
DBVPG1106	1	1	1	8	8	7	0	0	0	11	11	11
DBVPG6040	26	26	40	0	0	0	3	3	3	15	15	17
K11	11	11	15	22	22	22	5	5	6	12	12	15
NCYC110	5	5	12	13	13	12	1	1	3	12	12	12
NCYC361	24	24	31	0	0	0	4	4	4	18	18	22
S288c	13	13	17	0	0	0	1	1	3	12	12	12
DBVPG1853	26	26	26	12	12	12	13	13	9	23	23	23
DBVPG6765	11	11	43	14	14	14	0	0	0	13	13	17
DBVPG1373	9	9	19	6	6	6	1	1	1	14	14	19
DBVPG1788	1	1	7	8	8	8	0	0	0	13	13	15
SK1	12	13	22	15	15	16	2	1	0	12	12	15
L_1374	4	4	4	8	8	6	0	0	0	9	8	10
DBVPG6044	10	10	17	14	14	14	1	1	5	12	12	17
UW03_461_4	8	8	9	21	21	24	0	0	0	16	17	17
UW05_217_3	25	25	32	7	7	6	0	0	0	16	17	19
UW05_227_2	10	10	10	19	19	21	0	0	0	14	14	13
UW83_787_3	37	37	39	5	5	6	1	1	1	22	22	21
UW87_2421	4	4	9	13	13	14	0	0	2	15	15	15
W303	2	9	15	0	0	0	0	0	2	0	1	5
Y9	10	10	12	8	8	8	5	5	5	15	14	18
Y12	14	14	16	8	8	8	5	5	5	18	19	19
Y55	4	12	13	15	15	14	1	2	3	13	12	13
YIIc17_E5	23	23	26	5	5	7	2	2	4	19	18	22
YJM975	4	4	7	6	6	6	0	0	0	15	15	16
YJM978	2	2	11	6	6	8	0	0	0	14	14	15
YJM981	5	5	7	9	9	6	0	0	1	17	16	16
YPS128	0	0	5	14	14	14	0	0	0	12	12	12
YPS606	3	3	3	13	13	13	0	0	0	12	12	12
YS4	30	30	42	9	9	9	6	6	7	19	19	22
YS9	27	27	29	1	1	4	5	5	4	16	16	17
Total	**428**	**444**	**620**	**295**	**295**	**301**	**67**	**67**	**82**	**502**	**501**	**548**

Table 2.6.: Summary of TURNIP output for *S. cerevisiae*. The results of run 1 (clipped data, default BLAST) can be compared to those of run 2 (clipped data, non-default BLAST) for each polymorphism type

However, on closer examination of individual *S. paradoxus* strains, variation inconsistencies were noted. For example, CBS432 is the type strain for *S. paradoxus*, and the previously published rDNA sequence for this strain was used as the consensus sequence upon which to align all of the *S. paradoxus* reads. However,

Strain	Run 1 pSNP	Run 2 pSNP	Run 1 SNP	Run 2 SNP	Run 1 INS	Run 2 INS	Run 1 DEL	Run 2 DEL
A4	16	26	88	88	14	14	11	12
A12	38	43	70	71	13	13	12	15
CBS432	5	101	5	3	2	8	5	15
CBS5829	3	21	6	4	2	3	3	1
DBVPG4650	5	14	2	2	3	2	4	4
DBVPG6304	20	37	95	93	8	8	10	11
IFO1804	37	36	37	37	4	8	9	12
KPN3828	37	46	5	5	8	6	6	5
KPN3829	16	20	6	6	4	4	5	6
N_17	25	41	0	0	5	5	7	6
N_43	32	56	39	39	6	6	11	13
N_44	35	32	36	36	5	4	9	10
N_45	51	83	2	2	5	6	8	8
Q32_3	1	11	0	0	1	1	1	1
Q59_1	5	22	0	0	3	9	1	1
Q62_5	15	31	2	2	2	9	1	1
Q89_8	0	12	0	0	1	1	1	2
Q95_3	1	18	0	0	2	4	1	2
S36_7	10	10	0	0	2	2	1	1
T21_4	22	35	0	0	3	2	3	4
UFRJ50791	18	18	95	95	15	15	14	14
UFRJ50816	181	194	62	62	14	15	19	22
UWOPS91_917.1	345	367	57	57	37	35	33	36
Y6_5	18	34	0	0	2	4	2	2
Y7	0	8	1	1	1	1	1	4
YPS138	19	15	89	88	12	15	7	6
Z1_1	18	20	1	1	3	4	2	3
Total	**973**	**1351**	**698**	**692**	**177**	**204**	**187**	**217**

Table 2.7.: Summary of TURNIP output for *S. paradoxus*. The results of run 1 (default BLAST) can be compared with those of run 2 (non-default BLAST) for each polymorphism type

when the BLAST parameters were changed, this strain had one of the largest numbers of pSNPs (101), including a number within the highly conserved 18S region. A number of other strains also achieved large levels of variation, for example strain UFRJ50816 (over 180 pSNPs in both TURNIP runs).

To check for possible contamination of individual strains, or to further support the discovered variation, CBS432 reads were examined more closely. For example, 9 pSNPs were identified within region 7752 to 7777, one of which was identified in the earlier run with stricter BLAST parameters (run1). One of the reads which contained a pSNP identified within this region, after checking the TURNIP results directory for file 7760_results.txt, was 'CBS432-25b09.q1k'. Looking at the corresponding blast output directory file (7760-tmp_blast.out), this read had 87 out of 100 matches to the consensus position. To check the quality of the read, the trace was checked at the NCBI trace archive (www.ncbi.nlm.nih.gov/Traces), using the query **TRACE_NAME IN ('CBS432-25b09.q1k')**, but all peaks seemed distinct and of good quality at this position. The read was then BLASTed using a standard nucleotide blast (blastn) against the NCBI BLAST nucleotide collection database, which yielded a 95% maximum identity match, of 1235/1294 hit for *Plasmodium falciparum* 3D7 chromosome II (Sequence id: gb|AE014186.2|). This process therefore identified contamination of this strain with *Plasmodium falciparum*. To check that the reads which mapped to this position in this strain were derived from CBS432, this method was repeated with an identified read which did not contain pSNPs or other polymorphisms (CBS432-10b02.p1k), which had a top hit of *S. cerevisiae* 18S rDNA gene, (Sequence ID: dbj|AB594475.1|) with a 93% identity and 1063/1140 match. Previous runs which had used the original default parameters were checked for the presence of these reads which were identified as contaminants, and the contaminant reads were also found in the earlier runs. This indicated that the data needed to be cleansed of possible contamination before running through TURNIP. Furthermore, a thorough method to identify contamination and poor matching reads was needed, as well as a testing procedure to ensure that identified contaminants were being removed from the relevant dataset.

2.4.2. Identifying Contamination

The extent of spurious polymorphisms identified within the yeast strain dataset, potentially the result of contamination, was investigated. The role of potential

contaminants in inflating estimates of polymorphisms was hinted at in our earlier discovery of large numbers of polymorphisms in highly conserved coding regions, areas of the rDNA unlikely to exhibit large quantities of variation. This presented difficulties in locating other contaminants in regions of the rDNA locus that are expected to possess more variants, such as spacer regions. A systematic approach to checking variant reads (within strain CBS432) was needed, to look at the scope of the problem in a way that could detect all contaminants.

Initially only strain CBS432 was examined. The TURNIP output was manually checked for low frequency polymorphisms in each text file, and the read i.d.'s corresponding to these variants were collated, resulting in a set of 387 redundant reads being identified, which equated to 36 unique read i.d.'s. For each of these unique i.d.'s the corresponding sequence was found from the fasta input file, and then blasted on the NCBI server against the nucleotide collection database using the Mega BLAST search method, with the results of the top hit for each BLAST analysis shown in table 2.8. Due to the poor similarity of some reads to the database after using Mega BLAST (for example reads CBS432-11d22.p1k and CBS432-171a16.p1k), blastn was used for comparison as this latter algorithm would be expected to find matches with lower similarity to the query sequence.

Of the 36 unique reads, 21 aligned best to *Plasmodium falciparum*, 5 aligned well to *S. paradoxus* or *S. cerevisiae*, and 10 matched best to *S. paradoxus* or *S. cerevisiae* but were either very short or had hits to other chromosomes (4 of the 10 appear to match to the right chromosome and species, but 6 are on the wrong chromosome or match to another strain best). Of the 21 reads that matched well to *P. falciparum*, 15 had poor matches to *S. paradoxus*, and the remaining 6 did not have any hits to *S. paradoxus* via this method. Furthermore, when these reads did match *S. paradoxus*, they were very small hits to the rDNA region in CBS432, with a high expect value, explaining their inclusion in the subsequent TURNIP analysis. Although the *P. falciparum* and *S. paradoxus* rDNA sequences are not highly similar, more highly conserved coding regions will possess small regions of sequence similarity that would result in a small number of reads from *P. falciparum* aligning to *S. paradoxus*. This suggests that filtering sequence read matches to the consensus sequence by length should be included as part of a standard TURNIP analysis.

Additional strains, both from *S. paradoxus* and *S. cerevisiae*, were then checked briefly using the same method as for CBS432, for any obvious signs of contamination, or of potential false positives from hits to the wrong chromosome,

Unique reads	Top blast match	Identity	e-value	Pass
CBS432-11d22.p1k	*Saccharomyces cerevisiae*	699/961	8.00E-137	?
CBS432-14b06.q1k	*Saccharomyces cerevisiae*	309/380	1.00E-073	?
CBS432-171a16.p1k	*Saccharomyces paradoxus*	97/112	3.00E-027	?
CBS432-175h06.p1k	*Saccharomyces cerevisiae*	628/857	7.00E-119	?
CBS432-19h24.p1k	*Saccharomyces cerevisiae*	408/524	5.00E-101	?
CBS432-19i01.p1k	*Saccharomyces cerevisiae*	408/524	5.00E-101	?
CBS432-44m17.p1k	*Saccharomyces cerevisiae*	201/261	2.00E-037	?
CBS432-67d17.q1k	*Saccharomyces cerevisiae*	185/230	1.00E-034	?
CBS432-79d05.p1k	*Saccharomyces cerevisiae*	428/512	8.00E-125	?
CBS432-25a03.q1k	*Plasmodium falciparum*	1087/1227	0.0	no
CBS432-25a19.q1k	*Plasmodium falciparum*	1065/1159	0.0	no
CBS432-25a24.p1k	*Plasmodium falciparum*	463/476	0.0	no
CBS432-25b09.q1k	*Plasmodium falciparum*	1147/1175	0.0	no
CBS432-25e19.q1k	*Plasmodium falciparum*	1086/1197	0.0	no
CBS432-25g05.q1k	*Plasmodium falciparum*	968/994	0.0	no
CBS432-27a23.q1k	*Plasmodium falciparum*	911/982	0.0	no
CBS432-27b11.p1k	*Plasmodium falciparum*	782/808	0.0	no
CBS432-27d11.p1k	*Plasmodium falciparum*	534/577	0.0	no
CBS432-27o05.q1k	*Plasmodium falciparum*	919/924	0.0	no
CBS432-29f07.p1k	*Plasmodium falciparum*	1117/1160	0.0	no
CBS432-29f16.q1k	*Plasmodium falciparum*	519/540	0.0	no
CBS432-29f19.q1k	*Plasmodium falciparum*	940/967	0.0	no
CBS432-29g13.p1k	*Plasmodium falciparum*	1046/1065	0.0	no
CBS432-29g18.p1k	*Plasmodium falciparum*	1186/1269	0.0	no
CBS432-29h21.p1k	*Plasmodium falciparum*	1122/1159	0.0	no
CBS432-29i20.p1k	*Plasmodium falciparum*	794/827	0.0	no
CBS432-29j21.q1k	*Plasmodium falciparum*	858/890	0.0	no
CBS432-29n06.p1k	*Plasmodium falciparum*	1103/1160	0.0	no
CBS432-29n11.p1k	*Plasmodium falciparum*	665/688	0.0	no
CBS432-29n11.q1k	*Plasmodium falciparum*	546/557	0.0	no
CBS432-180m21.q1k	*Saccharomyces paradoxus*	834/839	0	yes
CBS432-170d19.p1k	*Saccharomyces cerevisiae*	852/874	0.0	yes
CBS432-185f22.q1k	*Saccharomyces cerevisiae*	491/593	1.00E-137	yes
CBS432-35d17.q1k	*Saccharomyces cerevisiae*	767/946	0.0	yes
CBS432-181d19.p1k	*Saccharomyces paradoxus*	146/164	1.00E-047	?
CBS432-94c14.q1k	*Saccharomyces cerevisiae*	974/1222	0.0	yes

Table 2.8.: *S. paradoxus* strain CBS432 low frequency variation read check. Those with question marks were low complexity, or short reads, that did not match well, and so were classified as uncertain

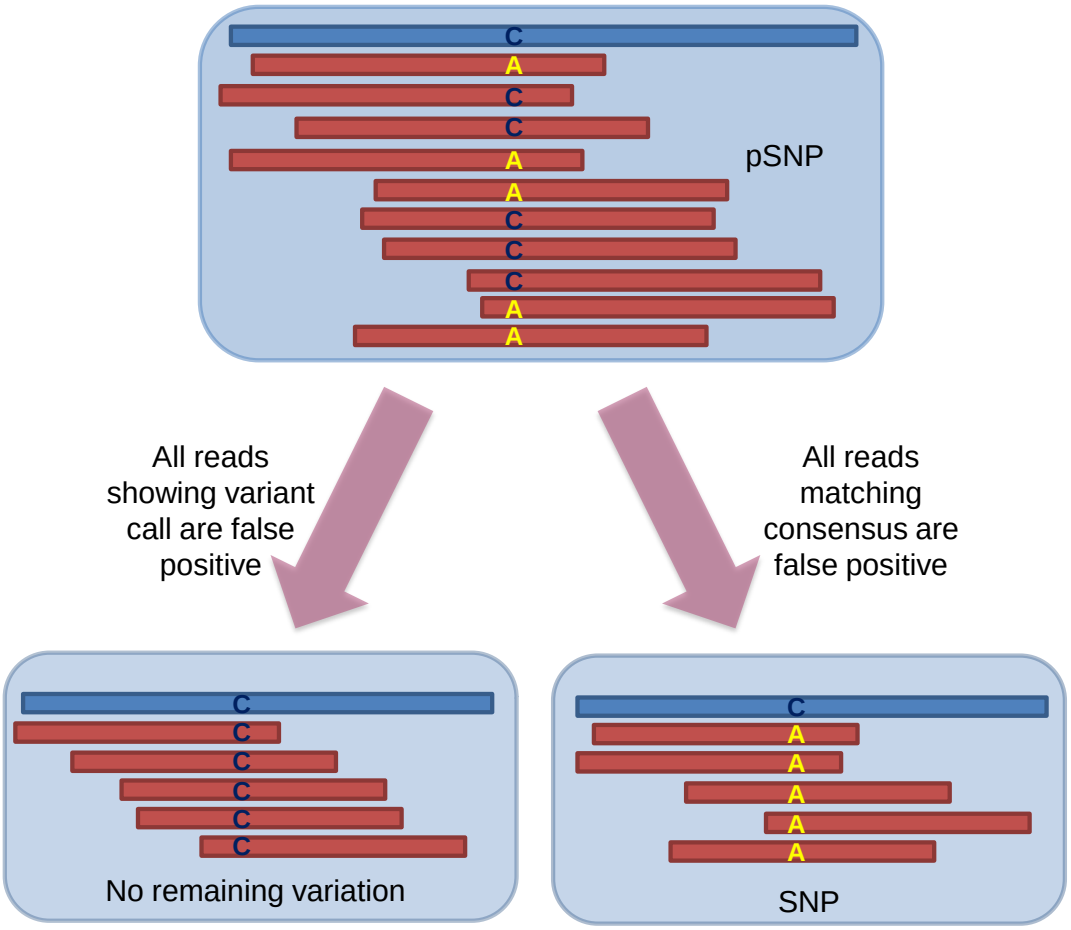

Figure 2.7.: A representation of a pSNP is shown in the top box, with a consensus sequence in blue, reads in red, with a C to A pSNP. If all reads matching to the consensus are false, variation becomes a SNP. If all reads possessing the variant nucleotide are false, no variation remains.

with results shown in table 2.9. 126 pSNP positions from 9 strains were analysed, and were identified as either true pSNPs, false positives (by variant reads matching better to a different chromosome) or SNPs (if the reads matching the consensus sequence at the polymorphism site were actually false positives leaving only variant reads), see figure 2.7. None of these strains showed any contamination with *P. falciparum* or any other species. Of the 126 positions checked, only 18 were found to be true pSNPs, and only in two strains (Y9 and Y55). 12 were reclassified as SNPs, and the remaining 96 were false positives, with the reads contributing to the original variation call matching well to other areas of the genome.

The results contain a non-negligible number of false positive pSNP polymorphisms

Strain	Positions tested	False positives	True pSNPs	SNPs
Y7	8	8	0	0
S36_7	10	10	0	0
YPS138	15	8	0	7
Q32_3	11	11	0	0
UFRJ50791	18	15	0	3
Z1_1	20	20	0	0
KPN3829	20	19	0	1
Y9	11	1	10	0
Y55	13	4	8	1
Total	**126**	**96**	**18**	**12**

Table 2.9.: Detailed analysis of 126 potentially false pSNPs in 7 *S. paradoxus* and 2 *S. cerevisiae* strains.

(76% of the 126 tested were false positives, with another 9.5% reclassified as SNPs, leaving only 14.5% correctly identified as pSNPs). Although such a method could be employed on the entire dataset, it would be too time consuming to be feasible for this analysis or for analysis of other datasets in the future. Therefore clipping the data to reads hitting only the rDNA unit, and filtering them to include only long, high-quality matches, before running through TURNIP was deemed to be the most pragmatic approach, and a script to filter the data was identified as a requirement.

2.4.3. Filtering the data: Methodology and Script

A custom Perl script (filter_reads_v3.pl), employing BioPerl modules ((Stajich et al., 2002) version 1.6.9, Perl version 5.12.3), was used to filter the dataset more stringently. As part of the script, sequencing reads in fasta format from each strain were aligned to the *S. paradoxus* or *S. cerevisiae* consensus sequence (extended on each side with 600bp duplicated sequence from the other end of the rDNA unit, to account for reads hitting the overlap between adjacent rDNA units) using blastall (BLAST version 2.2.27+, and BioPerl module Bio::Tools ::Run::StandAloneBlast were used). Conditions for reads to pass the filter comprised a minimum read length of 150 bp, minimum identity of 75%, and minimum percentage of the original read involved in a High-scoring Segment Pair of 75%. Blast parameters included an E-value of 1 x 10^{-10}, gap opening penalty of 3, gap extension penalty of 1 and a nucleotide mismatch penalty of -1. In the final stage of the script, reads that passed the filter were then converted into a BLAST searchable database using formatdb, ready for use with TURNIP.

This clipping and filtering process resulted in a total of 36,522 and 44,479 rDNA-specific sequencing reads for the 26 *S. paradoxus* and 34 *S. cerevisiae* strains respectively, with the number of reads for individual strains shown in table 2.10 and table 2.11.

Strain	Original reads	Filtered reads
273614N	11881	834
322134S	12682	1045
378604X	13372	872
BC187	10512	532
DBVPG1106	9123	678
DBVPG1373	19404	1061
DBVPG1788	18549	908
DBVPG1853	15075	1608
DBVPG6040	11476	1136
DBVPG6044	22691	1736
DBVPG6765	55691	2557
K11	11428	431
L_1374	19057	703
NCYC110	11448	1389
NCYC361	9678	1249
Q32_3	21325	1070
Q89_8	16734	820
S288c	21287	1570
SK1	61957	2931
UWOPS03_461_4	12795	853
UWOPS05_217_3	12691	1260
UWOPS05_227_2	13491	705
UWOPS83_787_3	12298	593
UWOPS87_2421	12160	518
W303	32270	4425
Y9	10205	601
Y12	11102	660
Y55	67120	3204
YIIc17_E5	13089	794
YJM975	13314	657
YJM978	13614	651
YJM981	10899	2886
YPS128	19543	863
YPS606	24748	1212
YS4	13653	901
YS9	13505	566
Total	679867	44479

Table 2.10.: Numbers of *S. cerevisiae* reads before and after filtering

Strain	Original reads	Filtered reads
A4	21440	967
A12	22303	666
CBS432	56396	2863
CBS5829	45885	2689
DBVPG4650	28088	1514
DBVPG6304	29693	977
IFO1804	11659	822
KPN3828	11669	710
KPN3829	11412	660
N_17	46055	2569
N_43	25213	1114
N_44	21939	782
N_45	58835	2559
Q32_3	21325	1070
Q59_1	25390	822
Q62_5	24283	1080
Q89_8	16734	820
Q95_3	25189	779
S36_7	14552	486
T21_4	25173	1084
UFRJ50791	10068	477
UFRJ50816	23243	1064
UWOPS91_917_1	27080	1597
W303	32270	4425
Y6_5	17416	747
Y7	23673	1139
YPS138	21792	1137
Z1_1	17541	903
Total	716316	36522

Table 2.11.: Numbers of *S. paradoxus* reads before and after filtering

2.4.4. The Final TURNIP Analysis

Polymorphisms, comprising SNPs, pSNPs and indels, were identified within the two filtered strain datasets using TURNIP. Default parameters were used within the configuration file, with a minimum quality score of 38, and an allowed shortness of 38. As suggested by our previous analyses (Section 2.4.1), the BLAST parameters -b and -v within TURNIP were set to 800, higher than the default values, to allow all reads aligning to specific rDNA regions to be stored and analysed. To ensure confidence in pSNP discovery, the additional criterion that pSNPs should be present in more than a single read was asserted. The output was then inspected visually for complex mutations i.e. nucleotide positions in which there is more than one type of variation (James et al., 2009). These positions were then annotated manually.

The results of this modified filter were compared to the previous results. Reads that were involved in a pSNP in the runs from the unclipped, unfiltered data, but not in those from the more stringent modified filter, and a random selection of reads were then manually checked, confirming that the final results were highly likely to retain all true hits to the rDNA sequence while removing false positive matches. The i.d. of reads used in this check, the top hits found in the NCBI database for each read, and other information including whether the pSNP was assumed to be genuine or what the read matched to, are displayed in the appendix in tables A.1 and A.2 for *S. paradoxus*, and tables A.3 and A.4 for *S. cerevisiae*.

After filtering the *S. paradoxus* data, 29 reads that were involved in pSNPs across 16 of the 26 strains were investigated to see if they were true positives. The remaining 10 strains could not be checked as they had no pSNPs remaining after filtering. All 29 appear to be genuine pSNPs, with a close match to the rDNA region of *S. paradoxus* or *S. cerevisiae* in each case, (table A.2). Note that in a few strains, such as YPS138 and KPN3828, a number of pSNPs were scored as SNPs after filtering. In contrast, 59 reads from the *S. paradoxus* strains which were implicated in polymorphisms before filtering, but which were no longer present after filtering, were also checked (table A.1). All 59 of these reads were deemed to have been correctly removed during filtering, as they matched poorly to the *S. paradoxus* rDNA sequence, and were either contaminants, or mapped well to other regions of the genome. To highlight some specific examples, strain YPS138 position 2617 was previously involved in a pSNP. When BLASTed against the NCBI database only 268 bases of the query were involved in the top hit, so

it would not have passed the filtering criterion for length. The BLAST hit is poor, but it seems to match to many 26S regions in yeasts at approximately 33% (268/269 match, out of total read length of 812 bases). It is possible that this read could be part of the rDNA sequence but potentially at the flanking regions where the sequence is thought to degrade. However, further work would be needed to check whether or not this was the case, and for now it must be assumed to be a false positive. Other strains with pSNP positions lost after filtering include Y6_5, position 4050. In this case a previously identified pSNP has been reclassified as a SNP, as the consensus read matched the wrong chromosome. In N_44, position 5846 is also an example where the read which previously aligned to the consensus sequence was mapped elsewhere in the genome (in this case to Sec10p), thereby reclassifying the pSNP as a SNP after filtering.

Similarly 74 *S. cerevisiae* reads involved in pSNPs that remained after filtering were checked and all appear to be genuine, high quality matches to the rDNA sequence (table A.4). For those reads lost after filtering, 60 were checked (out of 279 pSNPs which were lost in total) (table A.3) and all appear to have been false positives. Of note, in strain YJM975 at position 4484, two reads were checked, one of which matched well to rDNA, the other to another chromosome. This resulted in this pSNP position being lost after filtering, as the resulting variation then fell below the threshold of more than one read, and could no longer be counted as genuine. In strain W303 at position 4523 a read was found to hit to the right area of the chromosome, but matching to the three 5S repeats that are just outside of the rDNA array. This region is not part of the rDNA array itself and the resulting variation is now discounted as a false positive. In future, TURNIP could be extended to work with Next-Generation Sequence data where reads will be shorter in length. It is important to consider that the reads matching to the 5S regions outside of the rDNA array could potentially erroneously pass a filtering step if shorter length thresholds are used to match to the consensus, in addition to some of the other false positives identified here.

As the filtering appears to have removed a sizeable number of false positives whilst identifying much of the genuine variation, the resulting polymorphism data are believed to be of good quality, and are analysed further in the next chapter.

2.5. Conclusions and Chapter Summary

After TURNIP installation and initial runs highlighted possible bugs and inconsistencies, a thorough validation of the TURNIP suite and this methodology was undertaken. A number of bugs were removed, contributing to a new release of the TURNIP software, and optimised parameters in order to get the most information from the SGRP dataset. A script was written to produce data with known variation to test the confidence in results from the methodology, and after manually checking the results, issues with confidence in the data were discovered. To address this further scripts were written to filter possible contamination and reduce the dataset to rDNA specific reads, resulting in more confident identification of rDNA variation. This process has shown the need to check results manually at the end of an automated process, as this can lead to the discovery of unusual results or show possible weaknesses in a methodology. This process has resulted in a good quality dataset to analyse further, as presented in the following chapter.

3. Analysis of rDNA Variation

Chapter Abstract

The SGRP dataset was analysed using methods described in the previous chapter, identifying 978 and 1,168 SNPs, pSNPs and indels within the ribosomal DNA region of 26 *S. paradoxus* and 34 *S.cerevisiae* strains respectively. Although both species exhibit high levels of within-strain sequence heterogeneity, there is a difference in the structure of this variation which can be related to their differing evolutionary dynamics. The variation also allows discrimination of individual strains, and demonstrates that rDNA datasets can be used as an evolutionary proxy for the whole genome in terms of strain divergence. As part of this analysis two *S. paradoxus* strains were identified as having undergone putative hybridisation events, and additional levels of genome mosaicism in the *S. cerevisiae* dataset were identified. We discuss how patterns of rDNA variation could give insights into the dynamics of concerted evolution, providing a snapshot of the process which will be examined further in later chapters.

3.1. Quantifying Variation

3.1.1. Variation within the overall dataset

Venn diagrams of pSNP and SNP variation within selected strain groups were created using Venny (Oliveros, 2007). Regression and correlation analysis of variation with strain information were carried out in R (version 2.15.2) (R Development Core Team, 2011), using standard and MASS libraries (version 7.3-22). The level of variation identified in each of the 26 *S. paradoxus* rDNA arrays was found to vary markedly between strains, ranging from a single polymorphism in Q89.8 (European strain) to 114 polymorphisms in DBVPG 6304 (American strain) (see Table 2.2 for strain information). In total, 978 polymorphisms were identified, comprising SNPs, pSNPs, insertions, deletions and complex mutations.

Table 3.1 presents a breakdown of the mutations (type and total) found in each strain, with single nucleotide substitutions (either fully resolved as SNPs, or partially resolved as pSNPs) representing the most abundant form of mutation (79.6%). When ordered according to the total number of polymorphisms found in each rDNA array (Table 3.1), the 26 *S. paradoxus* strains could be readily split into the three distinct geographic populations as previously defined by Liti *et al.* (Liti et al., 2009), namely American, European and Far Eastern (Figure 3.1). When comparisons were made with the reference strain CBS 432 (the *S. paradoxus* type strain), the 16 European strains were found to have the fewest number of polymorphisms (129; 8.1 per strain), while the four Far Eastern strains had over six times as many per strain (196; 49.0 per strain), and the six American strains were the most diverse with over thirteen times as many polymorphisms per strain (653; 108.8 per strain).

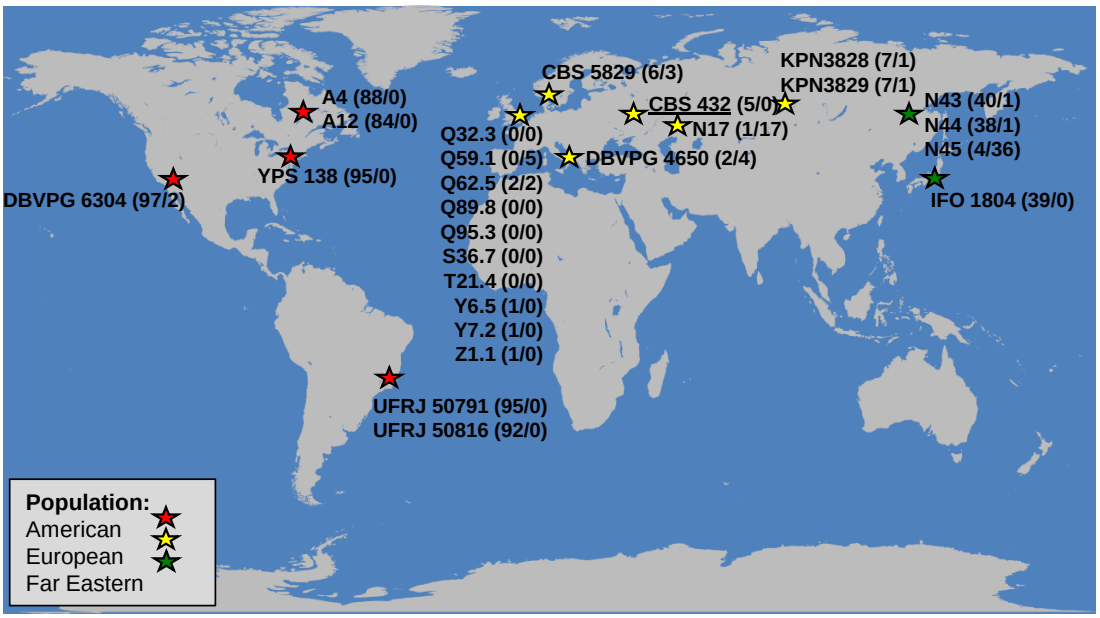

Figure 3.1.: World map with the location of the collection sites for the *S. paradoxus* strains indicated by stars. Stars are coloured by population type. In brackets following each strain are the number of SNPs and the number of pSNPs identified for that strain in this study. Used with kind permission from Dr Steve James.

In the original analysis of the SGRP *S. cerevisiae* strain set (James et al., 2009) the term complex mutation was used to refer to any nucleotide position within the rDNA array at which different *S. cerevisiae* strains exhibited different types of base substitution (i.e. a transition in one strain as opposed to a transversion in another). This definition has been revised to apply to any site, either a single nucleotide position or small region, where two or more different mutations have occurred in separate repeats of the same rDNA array (i.e. present at the same

location on different covering reads). Only two complex mutations were detected in the entire *S. paradoxus* dataset, and these were both found in Far Eastern strains, namely IFO 1804 (from Japan) and N-45 (from Russia). In both cases the

Strain	Population	SNP	pSNP	Deletion	Insertion	Complex	Total	Copy Number
Q89.8	European	0	0	0	1	0	1	81
Q32.3	European	0	0	1	1	0	2	74
S36.7	European	0	0	1	1	0	2	57
Q95.3	European	0	0	1	2	0	3	46
Y7.2	European	1	0	1	1	0	3	78
Z1.1	European	1	0	1	1	0	3	83
T21.4	European	0	0	2	3	0	5	66
Y6.5	European	1	0	2	2	0	5	65
Q62.5	European	2	2	1	1	0	6	68
Q59.1	European	0	5	1	4	0	10	52
CBS432 (T)	European	5	0	4	2	0	11	68
DBVPG 4650	European	2	4	3	2	0	11	87
CBS 5829	European	6	3	1	2	0	12	88
KPN 3828	European	7	1	3	2	0	13	82
KPN 3829	European	7	1	3	2	0	13	79
N-17	European	1	17	7	4	0	29	78
IFO 1804	Far Eastern	39	0	4	2	1	46	96
N-44	Far Eastern	38	1	5	3	0	47	52
N-43	Far Eastern	40	1	5	4	0	50	64
N-45	Far Eastern	4	36	8	4	1	53	65
A12	American	84	0	10	9	0	103	45
A4	American	88	0	6	11	0	105	66
UFRJ 50816	American	92	0	10	6	0	108	72
YPS138	American	95	0	7	8	0	110	76
UFRJ 50791	American	95	0	8	10	0	113	64
DBVPG 6304	American	97	2	7	8	0	114	52
Total		705	73	102	96	2	978	

Table 3.1.: Table of variation for each *S. paradoxus* strain, compared to the reference strain CBS 432, as identified using the TURNIP software. For each strain, the population and estimated rDNA copy number are also given. Ordering the strains by total polymorphism count results in the strains being split into their population groups.

mutation was found at the same location within the IGS1 region (base positions 3,929 to 3,937), and involved a homopolymeric polyT tract. In the reference strain, this tract comprises of 9 T residues. However, in both IFO 1804 and N-45 this tract appears to be variable in length. For example, in N-45 three different variants were detected, one identical in length to the reference strain, and two significantly longer (32 Ts and 33 T s), with the longest length variant (33 Ts) found in the majority of covering reads (28/33) (Figure 3.2 and Table 3.2). In contrast, although significantly longer than the reference strain, this tract appears to be of fixed length in the other two Far Eastern strains (N-43, 23 Ts; N-44, 26 Ts).

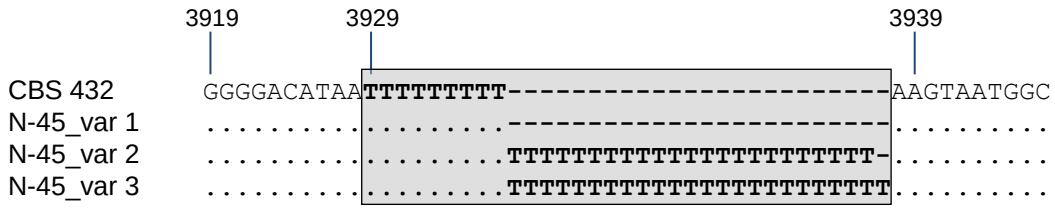

Figure 3.2.: Variable length homopolymeric polyT tract found in the *S. paradoxus* N-45 IGS1 region (TURNIP alignment positions 3929 to 3937)

Sequence Type	Tract length	Trace ID (TI #)	Frequency
N-45_var 1	9T	1254238616	6.1% (2 reads)
N-45_var 2	32T	1254241747	9.1% (3 reads)
N-45_var 3	33T	1254229529	84.8% (28 reads)

Table 3.2.: Variable length homopolymeric polyT tract found in the *S. paradoxus* N-45 IGS1 region (TURNIP alignment positions 3929 to 3937)

In addition to the varying levels of sequence variation found in the individual *S. paradoxus* rDNA array datasets, it was discovered that the detected variation was not distributed evenly over the rDNA repeat (Figure 3.3a and Table 3.3). Most of the identified polymorphisms were found in the non-coding ETS2, IGS1 and IGS2 regions, between positions 3500 and 6500 (Figure 3.3a). In contrast, and perhaps not surprisingly in view of functional constraints, very few of the 778 SNP and pSNP mutations were found in the rRNA-encoding genes. For instance, none were detected in either of the highly conserved 5S or 5.8S rRNA genes whilst eight were found in the 26S rRNA gene. Seven of these polymorphisms were found to be SNPs, six of which are specific to the American strains (base position 248), with the remaining variant identified as a low occupancy pSNP (2%) at base position 1174 in the Far Eastern strain N-45. Two additional SNPs were found in the

Region	*S. paradoxus*						*S. cerevisiae*					
	pSNP	SNP	DEL	INS	CX	Total	pSNP	SNP	DEL	INS	CX	Total
26S	1	7	0	0	0	8	18	4	1	2	0	25
ETS2	8	77	34	26	0	145	12	4	23	0	0	39
IGS1	30	237	59	66	2	394	121	110	304	27	6	568
5S	0	0	1	0	0	1	0	0	0	0	0	0
IGS2	19	317	6	4	0	346	111	178	23	11	2	325
ETS1	9	48	1	0	0	58	21	24	79	0	0	124
18S	0	2	0	0	0	2	9	0	1	0	0	10
ITS1	5	15	0	0	0	20	19	19	0	3	2	43
5.8S	0	0	0	0	0	0	0	0	0	0	0	0
ITS2	1	2	1	0	0	4	4	0	24	6	0	34
Total	73	705	102	96	2	978	315	339	455	49	10	1168

Table 3.3.: The number of polymorphisms of each type split according to different regions of the rDNA unit for *S. paradoxus* and *S. cerevisiae*. DEL corresponds to deleted positions, INS to inserted, and CX to complex mutations.

18S rRNA gene (position 6742), and these were C to T transitions specific to the two Brazilian strains UFRJ 50791 and UFRJ 50816 (previously classified as *S. cariocanus* (Naumov et al., 2000)). All but one of the 198 insertions and deletions were only found in non-coding regions, with the majority located in the ETS2 and IGS1 regions (positions 3397 to 4502).

The variation uncovered in our new TURNIP analysis of the 34 *S. cerevisiae* strains (see Table 2.1 for strain information) is summarised in Table 3.4. In total, 1,168 polymorphisms were identified, with pSNPs and SNPs collectively representing 56% of the uncovered variation. Table 3.4 shows some large differences in identified polymorphisms from the previous study (James et al., 2009), potentially due to the different software used to uncover them. While many of these differences were a general decrease in the number of identified pSNPs, for a few strains this was accompanied by an increase in the number of identified SNPs. However, the remaining variation was still high, with the number of polymorphisms varying significantly across the strains, ranging from 4 in the W303 strain to 63 in DBVPG 1853, though the range of variation and the variance in mutation number per strain was not so large as that seen in the *S. paradoxus* dataset.

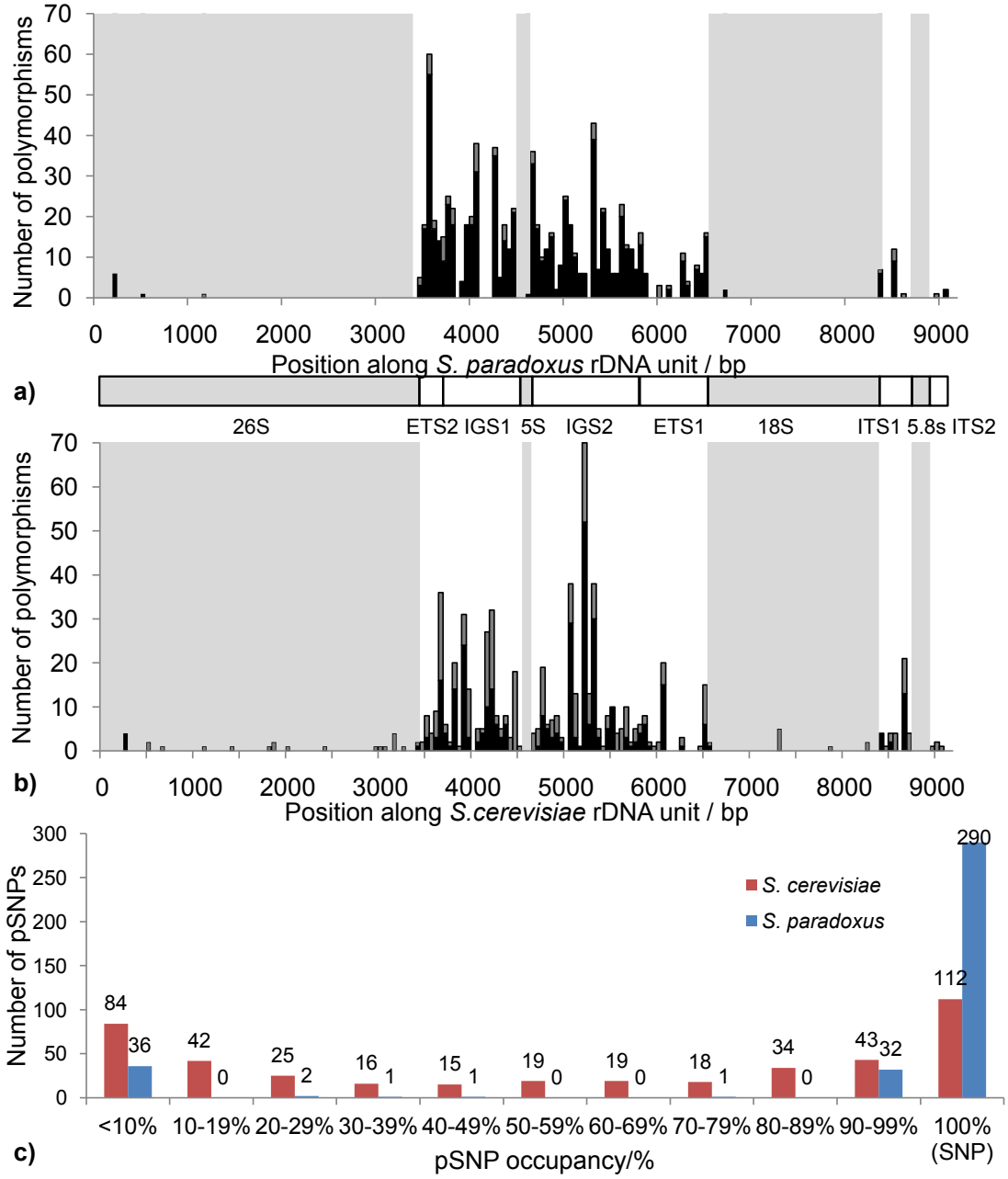

Figure 3.3.: The distribution of pSNP and SNP variants within the rDNA unit and their occupancies along the tandem array. a) pSNPs and SNPs within the *S. paradoxus* dataset, pSNPs are shown as dark grey bars, SNPs as black bars, with the boxed areas in light grey highlighting coding rRNA regions. Representation of an rDNA unit is shown below. b) pSNPs and SNPs within the *S. cerevisiae* dataset, pSNPs are shown as dark grey bars, SNPs as black bars, with the boxed areas in light grey highlighting coding RNA regions. c) Bar chart showing unit occupancies of pSNPs in the *S. paradoxus* and *S. cerevisiae* datasets, in occupancy bins of size 10%. For each species group, pSNP and SNP variants were recoded as changes from the putative ancestral base, instead of from the base(s) possessed by the reference strain.

[!htb]

Strain	Genome Type	Modified Genome Type	SNP	pSNP	Deletion	Insertion	Complex	Total	Copy Number
W303	Mosaic	Mosaic	0	3	1	0	0	4	182
L_1374	Structured	Structured mosaic	6	2	8	0	0	16	60
DBVPG 1106	Structured	Structured mosaic	7	1	9	0	0	17	98
DBVPG 1788	Structured	Structured mosaic	8	0	11	0	0	19	67
YJM975	Structured	Structured mosaic	6	4	12	0	0	22	65
YJM978	Structured	Structured mosaic	6	4	12	0	0	22	65
YPS128	Structured	Structured clean	14	0	10	0	0	24	62
YJM981	Structured	Structured mosaic	6	3	16	0	1	26	354
S288c	Mosaic	Mosaic	0	14	12	1	0	27	111
NCYC 110	Structured	Structured clean	15	2	9	2	0	28	163
YPS606	Structured	Structured clean	14	2	12	0	0	28	67
BC187	Structured	Structured mosaic	7	7	14	1	0	29	71
DBVPG 6765	Structured	Structured mosaic	13	3	13	0	0	29	70
DBVPG 6044	Structured	Structured clean	15	2	11	2	0	30	107
SK1	Mosaic	Mosaic	16	3	11	0	0	30	72
DBVPG 1373	Structured	Structured mosaic	8	7	15	1	0	31	75
UWOPS87-2421	Mosaic	Mosaic	14	4	13	0	0	31	57
322134S	Mosaic	Mosaic	6	12	14	2	0	34	109
Y55	Mosaic	Mosaic	15	7	12	1	0	35	78
27361N	Mosaic	Mosaic	4	15	14	3	0	36	93
Y9	Structured	Structured mosaic	8	10	14	3	1	36	72

Strain	Genome Type	Modified Genome Type	SNP	pSNP	Deletion	Insertion	Complex	Total	Copy Number
Y12	Structured	Structured mosaic	9	11	15	3	2	40	79
UWOPS05-227-2	Structured	Structured clean	24	7	11	0	0	42	70
378604X	Mosaic	Mosaic	0	20	19	4	0	43	87
K11	Structured	Structured mosaic	23	2	13	5	0	43	50
UWOPS03-461-4	Structured	Structured clean	29	0	15	0	0	44	89
DBVPG 6040	Mosaic	Mosaic	0	27	16	2	0	45	132
UWOPS05-217-3	Structured	Structured clean	27	3	15	0	0	45	133
YS9	Mosaic	Mosaic	1	27	14	2	2	46	56
YIIc17_E5	Mosaic	Mosaic	7	18	18	4	0	47	80
UWOPS83-787-3	Mosaic	Mosaic	8	21	19	1	0	49	64
NCYC 361	Mosaic	Mosaic	0	27	20	2	2	51	189
YS4	Mosaic	Mosaic	9	24	18	4	1	56	88
DBVPG 1853	Mosaic	Mosaic	14	23	19	6	1	63	144
Total			339	315	455	49	10	1168	

Table 3.4.: Table of variation for each *S. cerevisiae* strain, compared to the reference strain S288c, as identified using the TURNIP software. For each strain, the genome type (mosaic or structured), the modified genome type (mosaic, structured clean and structure mosaic) determined in this study, and the estimated rDNA copy number are also given.

In addition, this new analysis has uncovered significant numbers of insertion and deletion polymorphisms, which account for nearly 33% of all the variation detected in the two *Saccharomyces* species. Indeed, one of the striking differences between the variation identified in the two species is the large number of deletions found in *S. cerevisiae*. In fact 38.9% of all the detected variation in *S. cerevisiae* is due to deletions, compared to only 10.4% in *S. paradoxus* (Figure 3.4). A

closer inspection reveals the majority of *S. cerevisiae* deletions (67%) are found in the non-coding IGS1 region, with 75% of all IGS1 deletions specific to just five small regions, all of which are homopolymeric poly(dA).poly(dT) tracts. In the *S. cerevisiae* reference strain S288c, these five tracts range from 8 to 29 residues in length (Table 3.5). Some of these tract-specific deletions are found on all covering reads (i.e. are fully resolved) while others are found on only a proportion of all covering reads (i.e. are partially resolved). For example, in S288c a 16 residue poly(dT) tract was observed between base positions 3627 to 3642. In the soil strain DBVPG 1788, this same tract is shorter at only 13 residues in length, whereas in the beer spoilage strain NCYC 361, it exists in two variant forms, one identical in length to S288c (16 T residues) and present on the majority of covering reads, and a shorter variant (12 T residues) present on only six covering reads. While Ganley and Kobayashi (Ganley and Kobayashi, 2007) noted that a high number of deletions may be indicative of genome size reduction (Loftus et al., 2005), our observation that these mutations tend to occur within a small genomic area makes this phenomenon less likely in this case.

Tract type	**Location**		**Length**	
	S288c	**CBS 432**	**S288c**	**CBS 432**
polyT	3627-3642	3638-3653	16	16
polyA	3834-3841	3856-3861	8	6
polyT	3914-3935	3930-3938	22	9
polyT	4300-4316	Absent	17	0
polyA	4487-4515	4479-4495	29	17

Table 3.5.: Location and size of the five largest IGS1 poly(dA).(dT) tracts in *S. cerevisiae* (S288c) and their equivalent counterparts in *S. paradoxus* (CBS 432)

These results indicate that not only can homopolymeric tracts vary in length between different rDNA arrays of the same species, but they can also vary in length between individual repeats of the same rDNA array (e.g. NCYC 361). In contrast in the *S. paradoxus* reference strain (CBS 432^{T}), there are only four equivalent poly(dA).poly(dT) tracts in the IGS1 region, and two of these are significantly shorter in length than their *S. cerevisiae* counterparts (Table 3.5). These differences, both in tract number and tract size, appear to be a significant contributory factor as to why far more deletions are found in *S. cerevisiae*, and most notably in the non-coding IGS1 region. Long homopolymeric tracts, particularly poly(dA).poly(dT) tracts, are known to be unstable and prone to slip-strand replication errors, which in turn can give rise to (tract) length variation (Strand

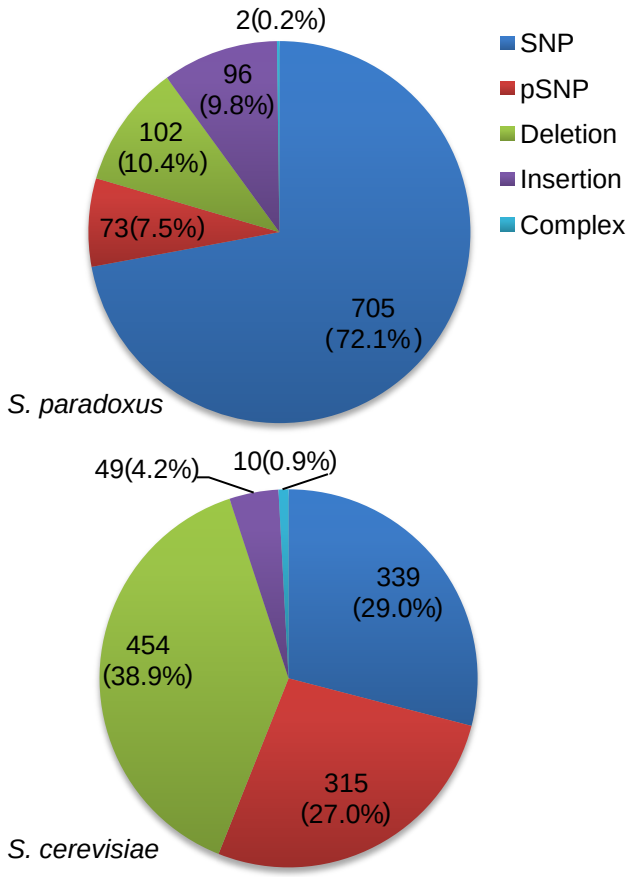

Figure 3.4.: Pie charts of number of each type of polymorphism in the *S. paradoxus* and *S. cerevisiae* datasets. Numbers of each type are shown, with the percentage of each polymorphism as part of the entire dataset given in brackets.

et al., 1993).

The other major differences in the mutational profiles, the relative proportions of each mutation type, of the two species groups (Figure 3.4 and 3.5) are the high number of SNPs in *S. paradoxus* and the high number of pSNPs in *S. cerevisiae* compared to *S. paradoxus*. In general, the mutational profiles show that certain types of polymorphism are favoured in each species and furthermore that these differ between species. In their earlier analysis, Ganley and Kobayashi (Ganley and Kobayashi, 2007) also found a biased mutational profile in *S. paradoxus*, though they were not able to establish one in *S. cerevisiae* due to a lack of identified mutations.

The spread of variation across the rDNA unit was found to be uneven in *S. cerevisiae*, as was also evident in *S. paradoxus* (Figure 3.3b and Table 3.3). While high, though differing, numbers of polymorphisms were observed for both species

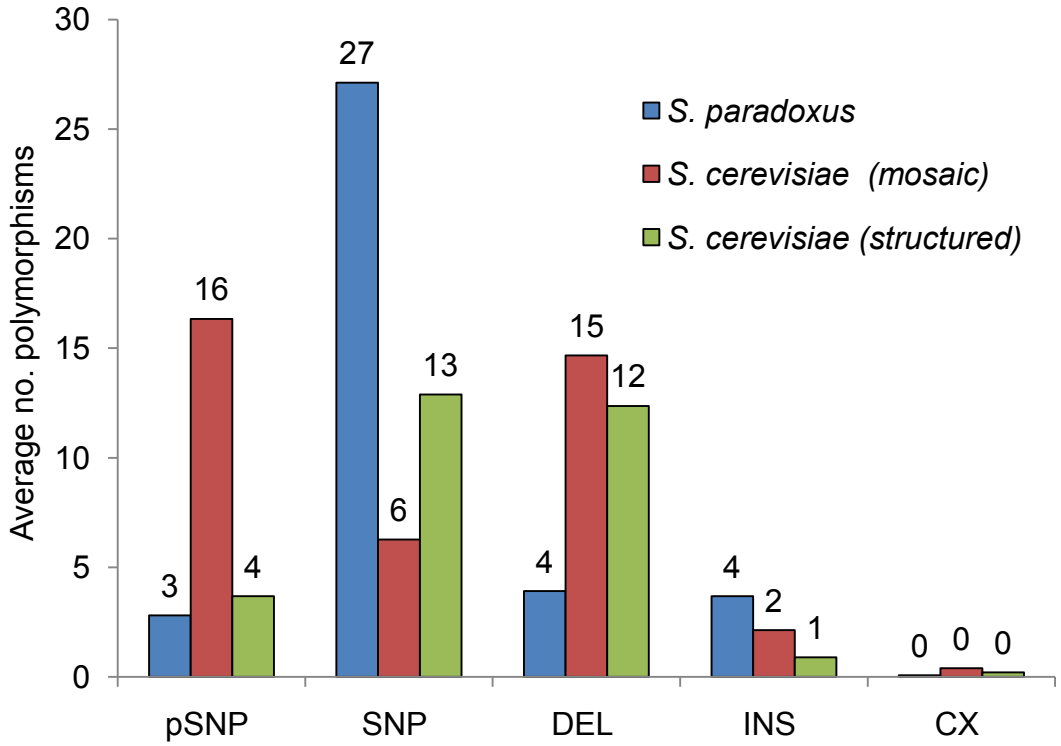

Figure 3.5.: Average number of polymorphisms per strain, split into *S. paradoxus* strains, *S. cerevisiae* mosaic strains, and *S. cerevisiae* structured strains. Number above the coloured bars are rounded to the nearest integer.

within IGS1 (48.6% in *S. cerevisiae* and 40.3% in *S. paradoxus*) and IGS2 (27.8% in *S. cerevisiae* and 35.4% in *S. paradoxus*), greater fold differences were observed in the numbers observed within the ETS1 (10.6% in *S. cerevisiae* and 5.9% in *S. paradoxus*) and ETS2 (3.3% in *S. cerevisiae* and 14.8% in *S. paradoxus*) regions. Furthermore, the mutation types contributing to these regional proportions differed markedly between species (e.g. the majority of IGS1 mutations in *S. paradoxus* were SNPs but in *S. cerevisiae* were deletions). Within coding regions, *S. cerevisiae* exhibited a higher number of polymorphisms than for *S. paradoxus* (35 compared to 11), most of which were pSNPs and most within the 26S rRNA gene.

A total of 10 complex mutations were identified (<0.9% of all detected variation) in *S. cerevisiae*. This rare type of mutation was detected in three strains with structured genomes (YJM981, Y9 and Y12) and four with mosaic-like genomes (DBVPG 1853, NCYC 361, YS4 and YS9). A closer inspection of the data revealed these mutations to be specific to just four sites within non-coding regions of the rDNA array; two in IGS1, one in IGS2 and one in ITS1. In addition, the type of complex mutation was also found to differ depending upon its location within the rDNA array. In the IGS1 region, two types of complex mutation were detected; a

complex insertion (of two or more sequences) between positions 3625 and 3631, and a complex substitution (A→G; A→T) at position 4484. A hexanucleotide (TTCCGC) tandem repeat of variable size (3 to 7 copies) was also identified in the IGS2 region between positions 5632 and 5649, and a variable length polyT tract (7 to 13Ts) was discovered in the ITS1 region between positions 8413 and 8419. The IGS1 complex insertion occurred most frequently, and was detected in five of the seven strains (DBVPG 1853, YS4, YS9, Y9 and Y12), although the actual insertions differed in sequence and size as well as number according to strain. The highest number of insertions was found in the baking strain YS4 which had three, and these ranged from two to 12 nucleotides in length, possibly reflecting the hybrid origin of this industrial strain and its resulting mosaic-like genome (Liti et al., 2009). Overall, more complex mutations were detected in mosaic strains (6) compared to structured strains (4), although of the latter, two are fermentation strains (Y9 and Y12) and so conceivably may also be hybrid in origin.

partial Single Nucleotide Polymorphisms

In total, 73 pSNPs were identified in the *S. paradoxus* dataset, an average of 2.81 pSNPs per strain. Over half the strains (15/26 strains) were found to have no pSNPs in their rDNA arrays, with a further six strains having no more than two pSNPs. Consequently, the previous identification of 8 pSNPs in a single strain (Ganley and Kobayashi, 2007) is consistent with our findings, falling at the upper end of our variation range. Notably the majority of the *S. paradoxus* pSNPs (72.6%) were detected in just two strains, namely N-17 (European strain; 23.3%) and N-45 (Far Eastern strain; 49.3%).

In N-17, isolated in Russia (Tatarstan), 16 of the 17 identified pSNPs had less than 4% occupancy and none of these 16 were shared with other strains within the European group. The remaining pSNP was found in the ETS1 region (position 6045), and had a 98.6% occupancy. Furthermore, this pSNP was found to be shared, as a pSNP, with a second European strain isolated in the UK (Q59.1) (Figure 3.6a).

The other *S. paradoxus* strain exhibiting a high pSNP count, the Far Eastern strain N-45, like N-17 was isolated from Russia (albeit from Terney, on the Russian coast of the Sea of Japan). In contrast to N-17, 32 of the 36 pSNPs identified in this strain, all with greater than 90% occupancy, were found at the same positions

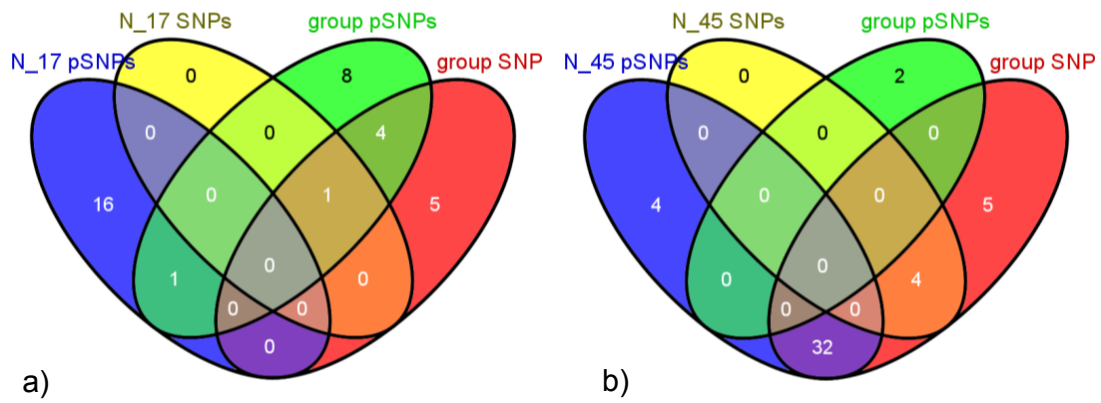

Figure 3.6.: a) Venn diagram of the pSNP and SNP locations in strain N-17 when compared to other SNPs and pSNPs in the European population group. 16 of the pSNPs in N-17 are not sites of variation in the other European strains. b) Venn diagram of the pSNP and SNP locations in strain N-45 when compared to other SNPs and pSNPs in the Far Eastern population group. 32 of the 36 pSNPs in N-45 are characterised as SNPs in other strains in the Far Eastern group.

as fully resolved SNPs in the other three Far Eastern strains (see Figure 3.6b). The remaining four pSNPs were not shared with other strains from the Far Eastern group. Three of these four pSNPs had low occupancy (1-2%) and were found at positions 1174 (26S rRNA-encoding gene), 5817 (IGS2) and 6045 (ETS1), while the fourth pSNP had a high occupancy (90.1%) and was found at position 5825 (IGS2). Although N-45 is the only Far Eastern strain to possess the 6045 pSNP, this A to G base substitution was also found (again as a pSNP) in two European strains, including N-17 as mentioned previously.

In general, and as illustrated in Figure 3.3c, the *S. paradoxus* pSNPs could be subdivided into two categories, those with very low occupancy (fewer than 10% of reads carrying a SNP), and those with very high occupancy (greater than 90% of reads carrying a SNP). In total, 36 pSNPs had less than 10% occupancy, 32 had more than 90%, while just five had intermediate occupancy (28.8 to 43.6%).

In a previous study (James et al., 2009), pSNPs were shown to be a prevalent type of variation in the rDNA region of *S. cerevisiae*. As noted above, the greater stringency used in our present study has removed or reclassified some previously identified polymorphisms. However, the quantity of pSNPs in strains of this species is still considerable. In this study, 315 pSNPs were detected in the *S. cerevisiae* dataset, an average of 9.26 per strain. Although this number of mutations is somewhat higher than that uncovered by Ganley and Kobayashi in their analysis of the RM11-1A strain (Ganley and Kobayashi, 2007), it is of note that this strain

is of a structured type and would therefore be expected to possess a low number of pSNPs.

In addition to a much higher pSNP frequency per strain than for *S. paradoxus*, this type of mutation was found in most of the *S. cerevisiae* strains analysed (31/34 strains). Unlike in *S. paradoxus*, where two strains possess the majority of pSNPs, a continuum of pSNP quantities was observed from the lowest strains (0 pSNPs in the structured strains DBVPG 1788, YPS128 and UWOPS03-461-4) to the highest (27 pSNPs in the mosaic strains DBVPG6040, NCYC361 and YS9). Furthermore, the occupancy distribution for *S. cerevisiae* pSNPs is different from that seen for *S. paradoxus* (Figure 3.3c). While there are some similarities between the two distributions, with both following a U-shaped curve, the *S. cerevisiae* distribution is much flatter, with a considerably greater number of pSNPs with occupancies ranging between 10% and 90%.

3.2. Phylogenetic Analysis

3.2.1. Method

Within-species phylogenetic trees were estimated from the combined pSNP/SNP datasets, rooted with a selected strain from the other species. For the *S. paradoxus* tree, the *S. cerevisiae* reference strain S288c was filtered and run through TURNIP, as described in the previous chapter, against the *S. paradoxus* CBS 432 consensus sequence. This scored variation between the S288c strain and similar regions within CBS 432. Variation output from TURNIP for the 26 *S. paradoxus* strains and the S288c *S. cerevisiae* strain were then processed using a custom Perl script (var_matrix_v3.pl) to construct a variation matrix in Phylip format. Within this process, each site in the rDNA consequence sequence at which a pSNP or SNP was found to occur in one or more strains was analysed. The frequency of each nucleotide base across the 27 strains at each varying site was calculated and written to the variation matrix, with examples of the output shown in figure 3.7.

The resulting variation matrix was then used as input to selected programs within the Phylip phylogenetic analysis suite (version 3.69) (Felsenstein, 2004), with an overview of the method used illustrated in figure 3.8. Specifically, a distance matrix was produced from the variation matrix using GENDIST with the

Cavalli-Sforza and Edwards Chord distance (Cavalli-Sforza and Edwards, 1967). A neighbor-joining tree (Saitou and Nei, 1987) was generated from this matrix using the NEIGHBOR program. 1,000 bootstrap datasets (Felsenstein, 1985) for the tree were then produced from the variation matrix using SEQBOOT and subsequently analysed using GENDIST and NEIGHBOR. The bootstrap trees were then mapped to the original tree using RAxML (version 7.3.0) (Stamatakis, 2006). This resulted in a bootstrapped *S. paradoxus* tree using the *S. cerevisiae* reference strain S288c as the nominated root. This tree was then visualised using MEGA 5 (Tamura et al., 2011), with only those bootstrap values greater than 50%

a)

Pos:		1174				248			
Base:		A	C	T	G	A	C	T	G
Consensus		0.000	1.000	0.000	0.000	1.000	0.000	0.000	0.000
A12	0.000	1.000	0.000	0.000	0.000	0.000	1.000	0.000	1.000
A4	0.000	1.000	0.000	0.000	0.000	0.000	1.000	0.000	1.000
CBS432	0.000	1.000	0.000	0.000	1.000	0.000	0.000	0.000	1.000
CBS5829	0.000	1.000	0.000	0.000	1.000	0.000	0.000	0.000	1.000

b)

3456			
A	C	T	G
1.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000
0.000	0.000	0.000	1.000
1.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000
0.984	0.000	0.000	0.016
0.000	0.000	0.000	1.000
0.000	0.000	0.000	1.000
0.031	0.000	0.000	0.969
1.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000
1.000	0.000	0.000	0.000

c)

```
27      411
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
A12             0.00 1.00 0.00 0.00 0.00 0.00 1.00 0.00
A4              0.00 1.00 0.00 0.00 0.00 0.00 1.00 0.00
CBS432          0.00 1.00 0.00 0.00 1.00 0.00 0.00 0.00
CBS5829         0.00 1.00 0.00 0.00 1.00 0.00 0.00 0.00
```

Figure 3.7.: a) example of output from the script var_matrix_v3.pl. Each position which has a pSNP or SNP in any strain is recorded, with the frequency of each base at that position shown. b) 2 pSNPs (highlighted in pink) and a SNP (in blue) represented at position 3456. c) example of variation matrix produced by var_matrix_v3.pl script, which is in a format compatible with Phylip. A row of 4s indicates the number of possible alleles at each position (one for each base), with the gray box highlighting one position

shown at branch nodes. These steps were repeated for the *S. cerevisiae* dataset, where *S. paradoxus* strain Q32_3 was used as the nominated root as the sequence of the reference/type strain CBS 432 was potentially contaminated (see Section 2.4.2). Phylogenetic networks for both datasets were produced using SplitsTree4 (version 4.12.3) (Huson and Bryant, 2006), using the Cavalli-Sforza and Edwards Chord distance matrices within the GENDIST output.

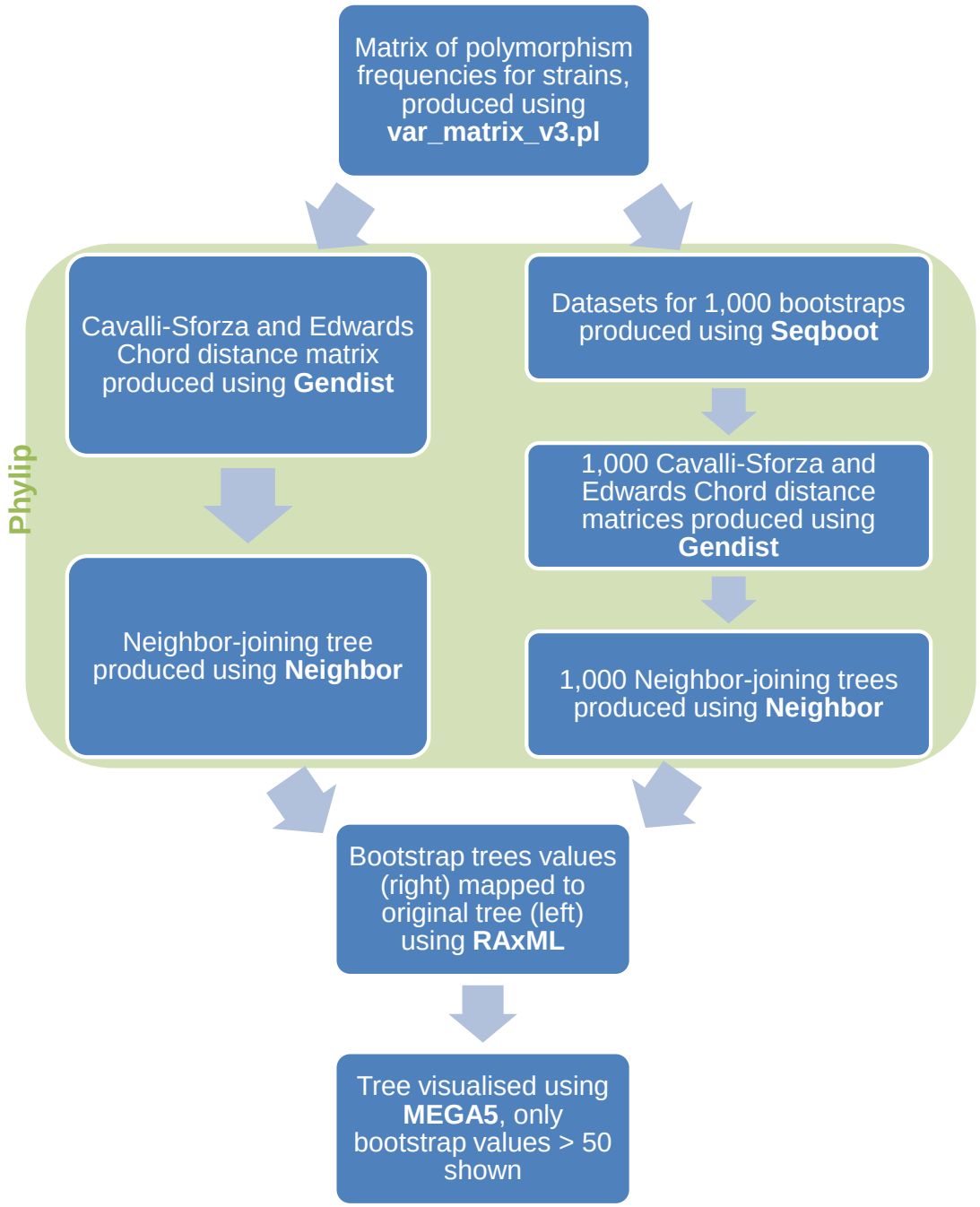

Figure 3.8.: Overview of the different programs used at different stages to produce the finished phylogenies, shown in figures 3.10 and 3.11, from the polymorphism frequency data. The programs which are part of the Phylip suite are shown within the green box.

The estimated trees were compared with phylogenies previously constructed from genome-wide SNP variation (Liti et al., 2009). *S. paradoxus* and *S. cerevisiae* distance matrices were downloaded from the Saccharomyces Genome Resequencing Project website (SGRP, 2013). The distance matrices were analysed using NEIGHBOR to estimate a Neighbor-Joining tree, strains additional to this analysis were removed using RETREE and the resulting trees were saved in Phylip format. Subsequent tree comparison was carried out using the software TOPD/FMTS (Puigbò et al., 2007) with the *disagree* option. The value of the resulting Split Distance statistic was compared to those calculated for 100 trees of the same strain set randomly generated by TOPD/FMTS. Distance matrix comparison was performed using a Mantel's test within the QIIME software (Caporaso et al., 2010).

3.2.2. Results

The SNP and pSNP polymorphisms identified in each of the 26 *S. paradoxus* strains were combined into a single dataset. SNPs and pSNPs were found to occur at 58 and 151 rDNA sites respectively, at 166 unique positions (i.e. 74.1% pSNPs occurred at sites where SNPs were also identified). The phylogenetic signal in the dataset appeared to be strong, with raw polymorphism counts highly correlated to geographical origin (American, European and Far Eastern; Pearson's $r = 0.987$) (Figure 3.9), and with the Far Eastern group of strains (IFO 1804, N-43, N-44 and N-45) exhibiting remarkably little variation in raw counts. The resulting rDNA-based phylogenetic tree mirrored this pattern, splitting into three well-supported groups that directly corresponded to geographical origins (Figure 3.10). The *S. paradoxus* phylogeny (Figure 3.10) showed little variation within the European group, particularly within the ten UK strains (Q95.3 to Q59.1). Likewise, the two Siberian strains (KPN3828 and KPN3829) were found to be highly similar to one another. In the Far Eastern group, most closely related to the European group, N-45 was found to be the most divergent of the four strains. The American strains proved to be most divergent as a group.

Notably, the new rDNA-based phylogeny was highly similar to that previously produced by Liti *et al.* (2009), generated from 623,287 SNPs spread across the nuclear genome. The grouping of strains into European, Far Eastern and American, and furthermore into UK and non-UK within the European group, was identical between the two trees. Minor differences in topology were seen within-group, with

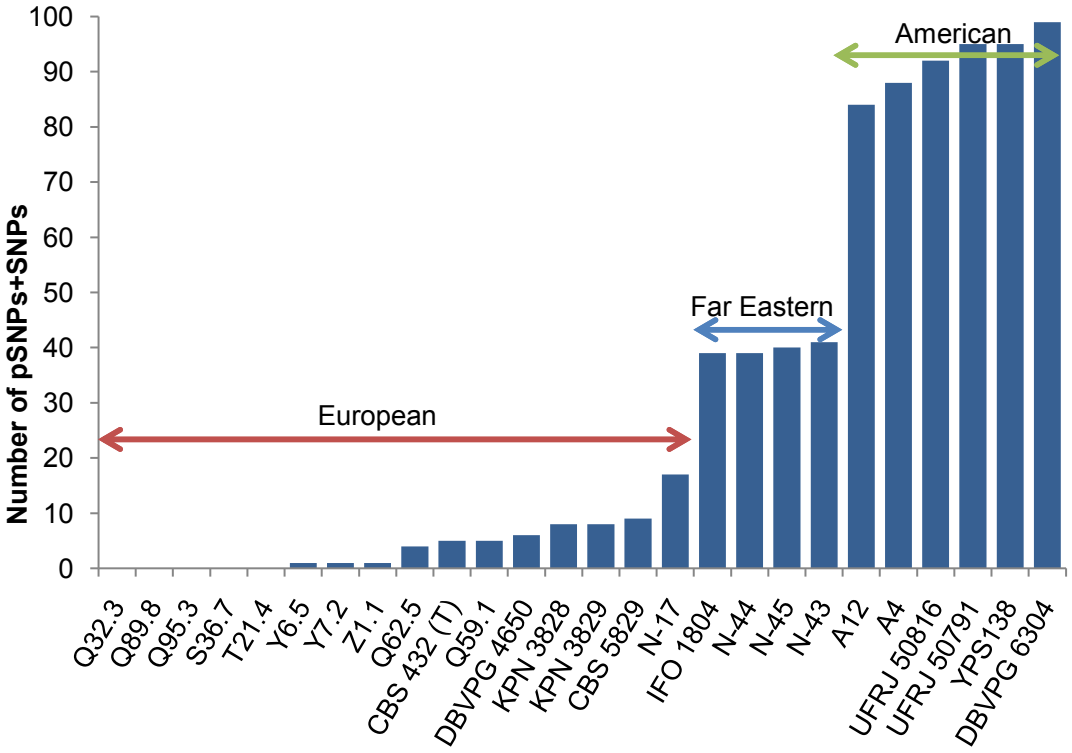

Figure 3.9.: Bar chart of pSNP plus SNP variation in each *S. paradoxus* strain, labelled to show the split into distinct populations. The strains are ordered by increasing number of pSNPs + SNPs, and naturally split into the three geographical locations.

N-17, CBS 432, N-45 and A12 the clearest examples.

In *S. cerevisiae*, SNPs and pSNPs were found to occur at 143 and 90 rDNA sites respectively, at 181 unique positions (i.e. 36.4% pSNPs occurred at sites where SNPs were also identified). In a previous *S. cerevisiae* phylogeny based on 235,127 SNPs distributed throughout the genome (Liti et al., 2009), the strains did not partition cleanly into distinct groups, but rather a subset of the strains grouped according to either geographic origin or industrial usage, with the remainder showing no strong grouping structure. Furthermore, the 19 highly-grouped strains were found to possess structured genomes, with the remaining 15 possessing mosaic genomes. As expected, phylogenetic analysis of such a dataset results in conflicting signals of inter-strain relationships.

Indeed, Figure 3.12 and Figure 3.13 show NeighborNets (Bryant and Moulton, 2004) estimated for the *S. paradoxus* and *S. cerevisiae* strain sets. It is clear from these networks that the *S. cerevisiae* dataset possesses a greater degree of phylogenetic conflict than that of *S. paradoxus*. Despite this issue, our new

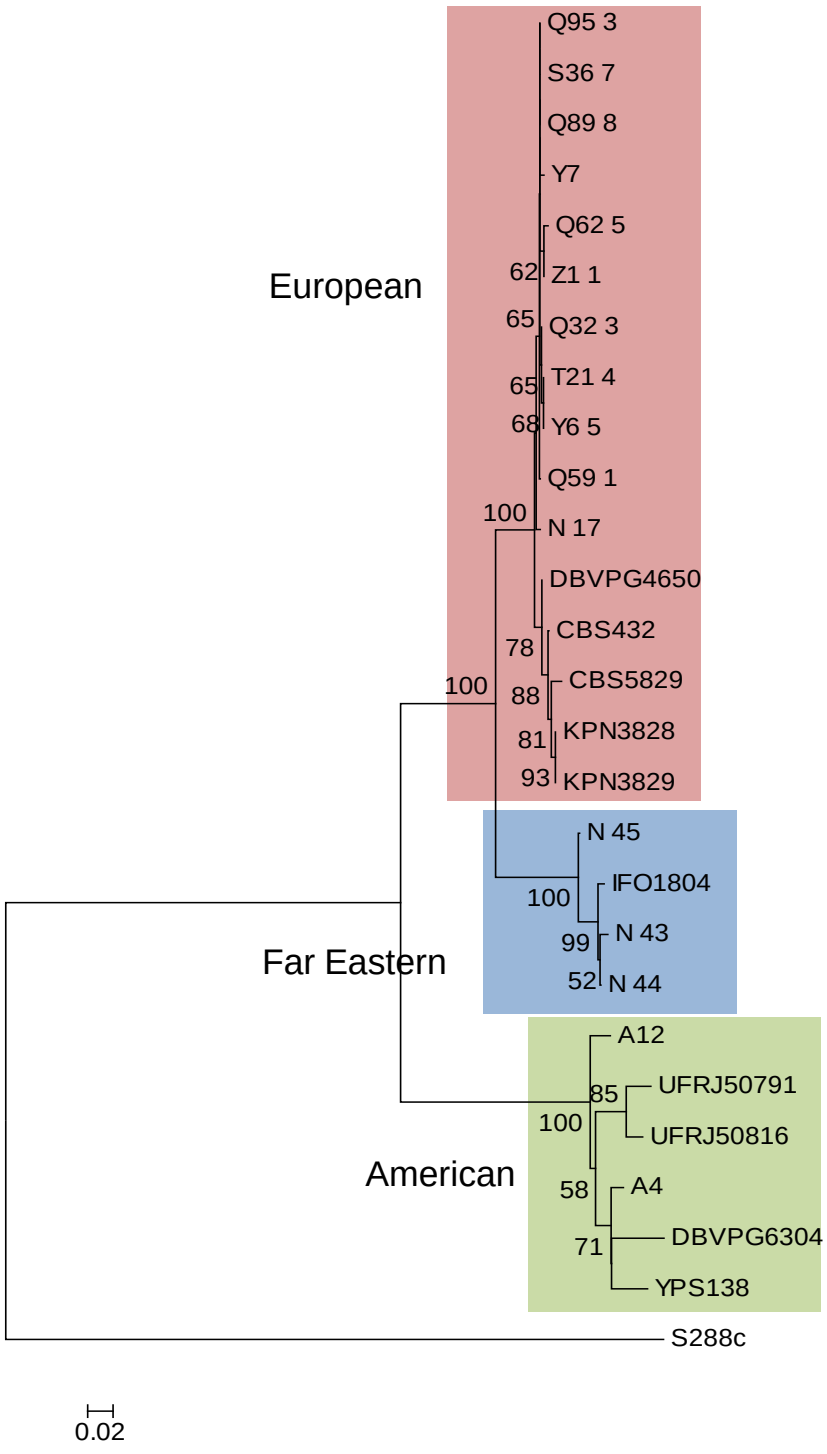

Figure 3.10.: *S. paradoxus* neighbor-joining tree with *S. cerevisiae* strain S288c as the nominated root. There is clear separation into groups according to the geographical location of the strain collection site. Only bootstrap support values greater than 50 are shown.

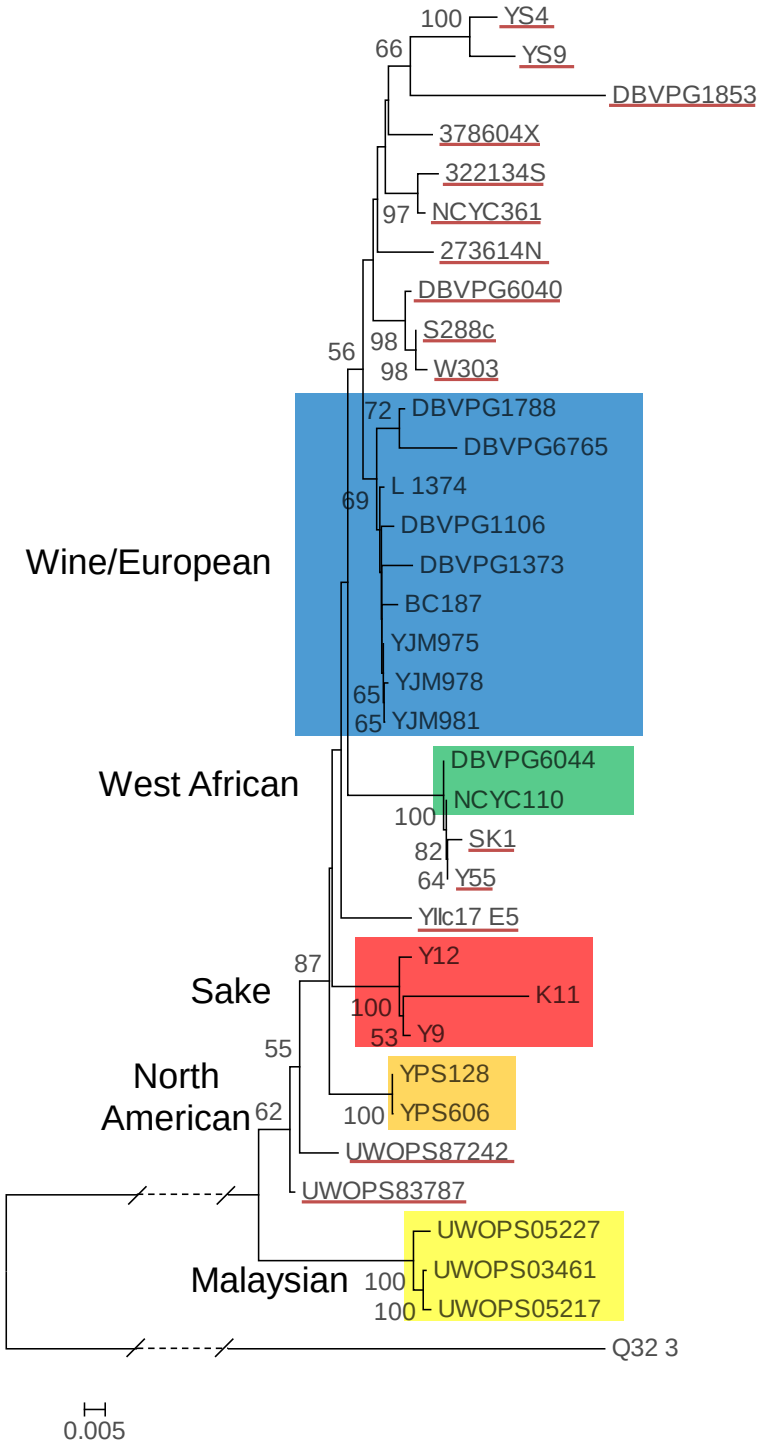

Figure 3.11.: *S. cerevisiae* neighbor-joining tree with *S. paradoxus* strain Q32.3 as the nominated root. Only bootstrap support values greater than 50 are shown. The dotted line is equivalent to a distance of 0.355. Groups of interest are shown as coloured boxes and mosaic strains are underlined in red.

rDNA-based phylogeny (Figure 3.11) is highly similar to that estimated by Liti *et al.* (Liti et al., 2009). For example, the new tree exhibits identical Malaysian, Sake, West-African and Wine/European groups (all consisting of structured strains) to the previous tree. Furthermore, there is an overall consistency in the relationships between the groups. The major difference between the two topologies is the location of the YIIc17_E5 strain. In the Liti *et al.* (2009) tree, this strain can be found amongst a loose group of mosaic strains adjacent to the Wine/European group while in our rDNA-based phylogeny, the strain is located closer to the West African and Sake groups. This difference could potentially be explained by the putative parentage of this mosaic genome, with different relative contributions of the parents within the genome-wide SNP and rDNA datasets. Indeed, on closer examination of the YIIc17_E5 pSNP/SNP polymorphisms, of the 25 rDNA sites at which this strain varies from the reference strain, two contrasting phylogenetic signals can be observed. One group of polymorphisms links YIIc17_E5 to the three Sake strains, while the other group links it to the set of mosaic strains close to the Wine/European group, in particular the 273614N, DBVPG6040 and S288c strains.

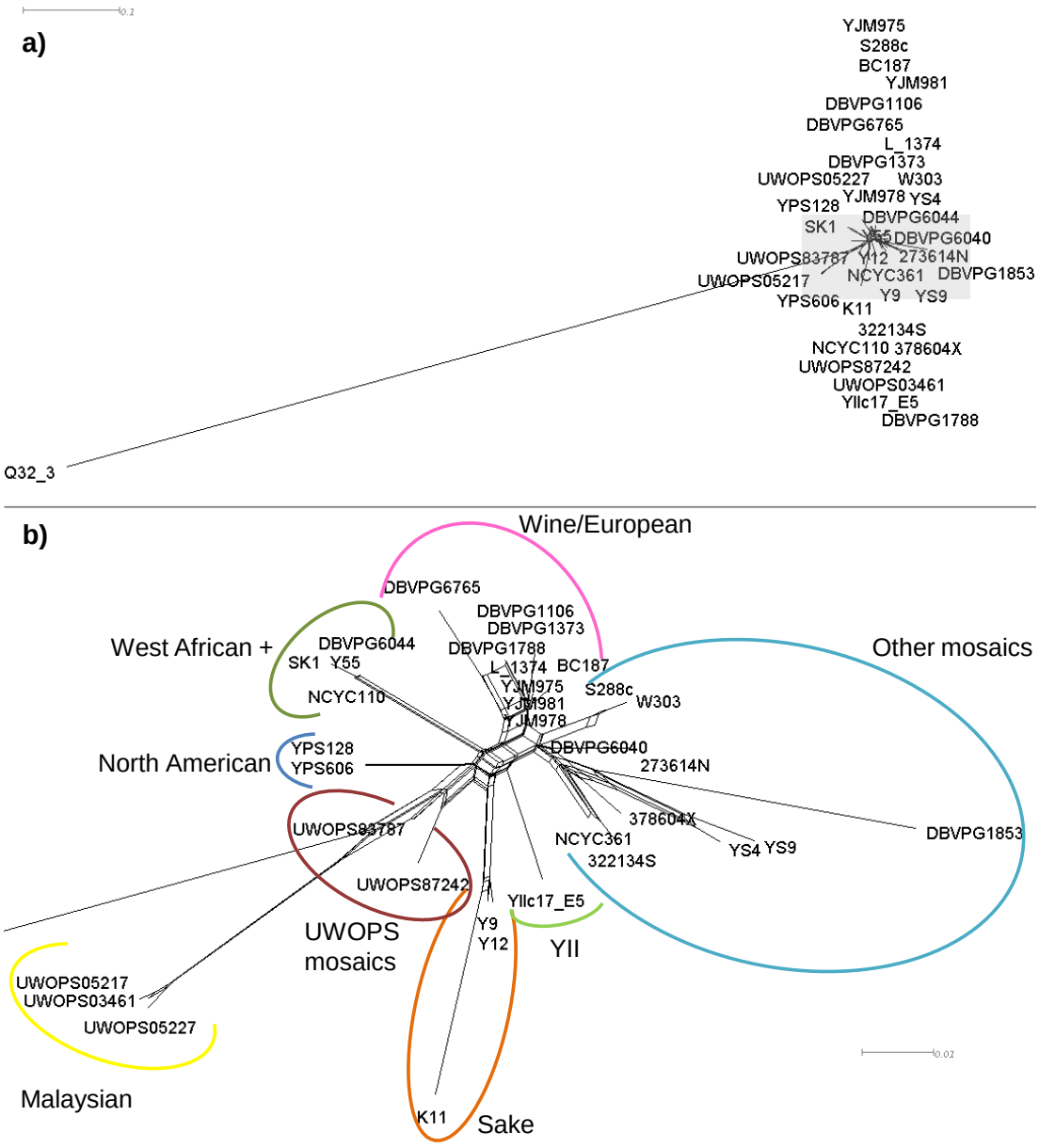

Figure 3.12.: a) The *S. cerevisiae* network shows a complex network structure, consistent with existing knowledge of this population. Overview of the whole network including outgroup. b) A close up of the main population structure in the network (highlighted in a) by the grey box), with different groups labelled and indicated with coloured lines.

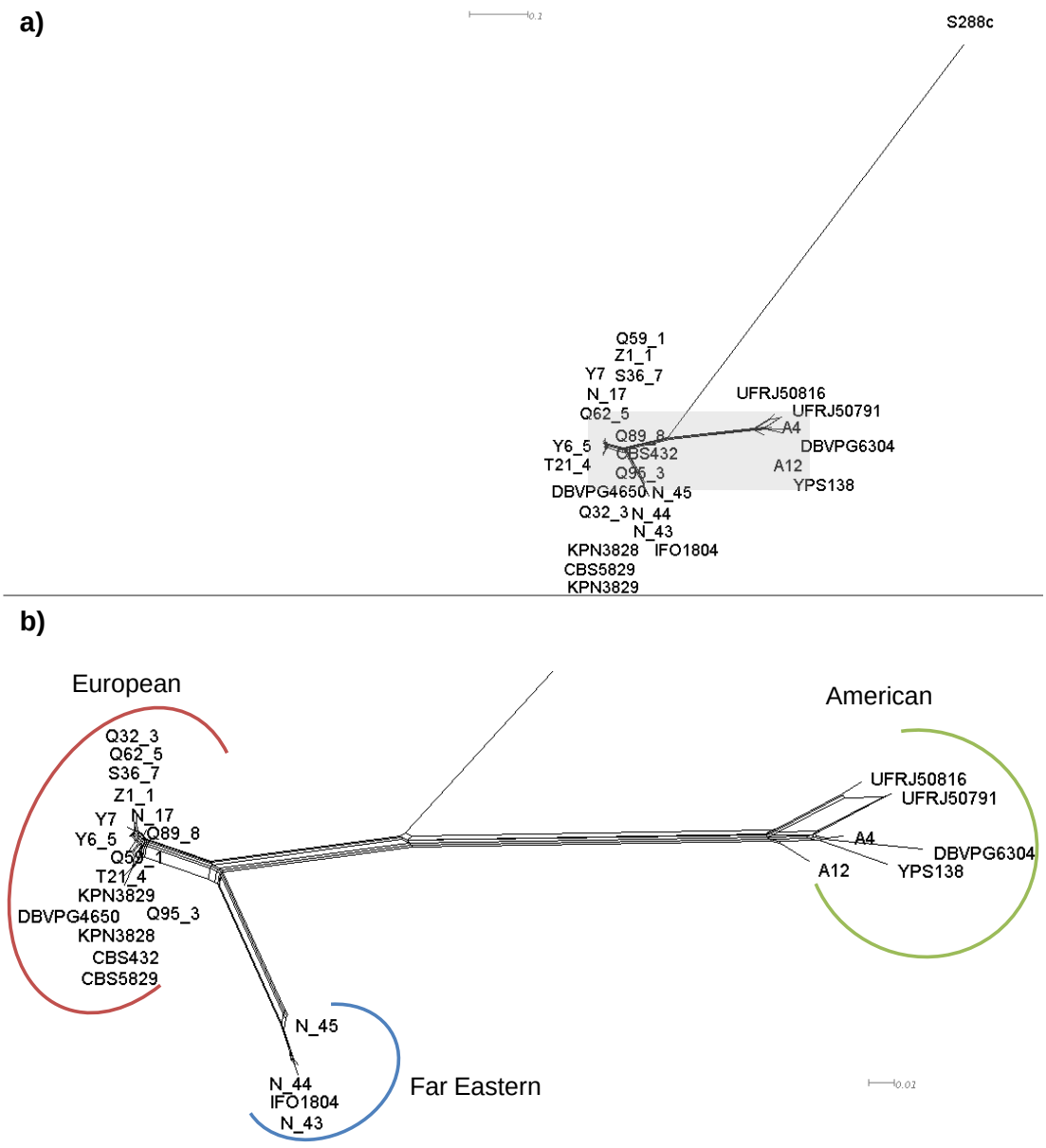

Figure 3.13.: a) The *S. paradoxus* network shows a clear separation of each geographic population. Overview of the whole network including outgroup. b) A close up of the main population structure in the network, with different geographical groups labelled and indicated with coloured lines.

3.2.3. Comparing the rDNA-based and genome wide SNP phylogenetic trees

The distance matrices and phylogenetic trees produced for both datasets were compared to those estimated for the whole genome SNP datasets in the SGRP project (Liti et al., 2009). The distance matrices for the SGRP data were obtained from the misc directory within the SGRP site (SGRP, 2013). The SGRP matrices are divergences between the strains expressed as polymorphic positions per 1000 nucleotides across the whole genomes of the strains. These positions were only counted when strains had a nucleotide present in an alignment, not if there was a gapped position.

Comparing the two *S. paradoxus* trees using the TOPD/FMTS software (Puigbò et al., 2007), the disagree statistic exhibited a Split Distance of 0.52 compared to a random Split Distance (using randomly generated topologies of the same strain set) of 0.99, reinforcing the closeness of the two phylogenies. When the two *S. cerevisiae* trees were compared, the disagree statistic exhibited a Split Distance of 0.65 compared to a random Split Distance (using randomly generated topologies of the same strain set) of 0.99. Although the two trees are not as close as for *S. paradoxus*, this result supports our observation that there is strong agreement between them. The SNP and pSNP Cavalli-Sforza and Edwards Chord distance matrices for each dataset, obtained during analysis with PHYLIP, were compared to the SGRP matrices using Mantel's test, as implemented by the `compare_distance_matrices.py` module in the QIIME program suite (Caporaso et al., 2010), version 1.6.0. Matrices were edited to be tab delimited with identical strain names (although not in the same order), and a header of each name added, with results shown in table 3.6. From these results both *S. cerevisiae* and *S. paradoxus* show a strong correlation between the two distance matrices, particularly for *S. paradoxus*. This suggests that our rDNA-based distances are highly similar to those estimated from the SGRP whole-genome SNP datasets.

Species	Mantel's r statistic	p-value
S. paradoxus	0.99029	0.001
S. cerevisiae	0.64133	0.001

Table 3.6.: Mantel's r statistic comparing distance matrices from the SGRP analysis and our rDNA-based pSNP and SNP distances.

3.2.4. The use of pSNPs in phylogenetic analysis

Sequence heterogeneity within the rDNA unit has long been a problem in phylogenetic analysis, with numerous studies citing this issue, in particular within the ITS region (Buckler et al., 1997; Kiss, 2012; Nilsson et al., 2008). Solutions have included creating consensus sequences suppressing the observed variation, but such workarounds are still far from ideal. Whole genome sequencing projects offer the potential to fully characterise the variation across the entire rDNA sequence. Indeed the only barrier to a full characterisation is the ability to sequence the whole locus (or loci, in the case of multi-locus rDNA systems). At present, rDNA sequences cannot be assembled into ordered tandemly arranged units. A consequence of this is that it is impossible to be sure that the target sequence has been uniformly sampled across all of its copies, and hence some variation may still be missing from recent studies. However, current sequencing technologies, where a sequences of interest can be sampled deeply, suggest that full characterisation is being approached. Furthermore, promised technological advances mean that assembly of repetitive sequences may soon be possible, and that the full allelic variation across the rDNA unit could be characterised.

It has been shown, for the first time, how a detailed characterisation of sequence variation within the rDNA unit can be coded as a form of allelic variation - in this case as and inter-connected systems of pSNPs and SNPs - and how this variation can be analysed with existing tools to estimate phylogenetic trees. In particular, the well-established Cavalli-Sforza and Edwards Chord distance (Cavalli-Sforza and Edwards, 1967), a natural choice for variation of this kind, has been used in conjunction with the Neighbor-Joining method. The resulting phylogenetic trees (Figures 3.10 and 3.11) are highly similar in topology to those estimated in previous analyses (James et al., 2009; Liti et al., 2009). For *S. paradoxus*, this is perhaps not so unexpected, as the majority of pSNPs either have a high occupancy (over 90%) where they will be treated similarly to SNPs, or low occupancy (less than 10%), where they will not contribute significantly to pairwise distances. However, there is good agreement between this new phylogeny and previously estimated trees for *S. cerevisiae*, where occupancy ranges are much different and network-like signals resulting from hybridisation events are known to be a problem.

Interestingly, a recent computational study of SGRP *S. cerevisiae* genomes (plus additional genome sequences from the Saccharomyces Genome Database (SGD) (SGD, 2013) for validation) showed that a minimal set of 13 specific genes can

capture the phylogenetic relationship inherent to these strains (Ramazzotti et al., 2012). The method was proposed as a simpler alternative to whole-genome sequencing, and is highly attractive when financial or analytical constraints are a factor. However, some major challenges were faced by this approach, in particular the inconsistency of gene content across strains. Conversely, our analysis has shown that a single, complex locus may satisfy many of the goals of this study while also being universal across and within species. However, developing datasets such as the one used in this study would be unachievable for many at the present time. It would be interesting to see whether future technologies could achieve full characterisation of the rDNA sequence without the need for whole-genome sequencing.

Perhaps uniquely, the rDNA unit offers the opportunity to capture sequence variation before it is fixed as a SNP (or conversely is lost), and therefore is ideal for understanding the relationships between members within a species, such as those analysed here. This point, together with the quality of the resulting trees, leads to the conclusion that the analysis of pSNP and SNP variation within the rDNA unit offers a valuable phylogenetic opportunity, particularly for fine-scaled evolutionary scenarios.

3.2.5. Population structure

The population structure of *S. paradoxus* observed in this analysis is consistent with that found in previous studies (Liti et al., 2009, 2006), where a split into three distinct geographical groups (American, European and Far Eastern) is clearly seen. The American group appears to be the most basal of the three groups investigated, and more distant to the remaining groups. In the European group, the UK and mainland Europe strains form two distinct subgroups (bootstrap value 68%), supporting a similar split seen in previous studies based on genome-wide SNP differences (Liti et al., 2009). Within this group, and apparently unusually within this species, strain N-17 displays signs of a putative inter-group hybridisation event (one of only two such events identified in this dataset, see below).

Compared to *S. paradoxus*, *S. cerevisiae* shows significantly lower inter-strain diversity. The *S. cerevisiae* population structure is also more difficult to infer, largely due to the different pattern of variation (more pSNPs and fewer SNPs) resulting from likely hybridisation between strains. Indeed, a NeighborNet analysis

of the pSNP/SNP distance matrix indicates a much stronger non-treelike signal in this species (Figure 3.12), as has been previously suggested (Liti et al., 2006).

Although the *S. paradoxus* strains possess many more SNPs than pSNPs, their distribution across the rDNA repeat unit shares some similarities to that of *S. cerevisiae*, as seen in Figures 3.3a and b. In both species, as expected, the vast majority of polymorphisms are found within the non-coding regions. However, there are a small number of pSNPs, albeit at very low occupancy, in both datasets within the 26S rRNA-encoding gene, as well as in the 18S rRNA-encoding gene in the *S. cerevisiae* dataset (Liti et al., 2006). Indeed the low occupancy of these polymorphisms is consistent with Ganley and Kobayashi's idea of a tolerance threshold (Ganley and Kobayashi, 2007), where a small number of mutations may be harboured within the rDNA regions without detrimental effect.

SNPs are also present in the variable D1/D2 region of the 26S rRNA-encoding gene in both datasets, a region of rDNA that is important for yeast identification (Fell et al., 2000; Kurtzman and Robnett, 1998). In *S. paradoxus*, the D1/D2 SNP (position 248) is present in all six American strains, and is the same nucleotide (T residue) as is present in *S. cerevisiae*, indicating that this position may have mutated more recently in the Far Eastern and European groups. In *S. cerevisiae*, the D1/D2 SNP (position 253) is present in the two West African strains (DBVPG 6044 and NCYC 110) as well as the laboratory strains SK1 and Y55. The latter two strains are believed to be derived from crosses between West African and European/Wine strains (Liti et al., 2009), and both possess the same D1/D2 polymorphism (A to G transition) as is found in DBVPG 6044 and NCYC 110.

3.2.6. pSNPs as a predictor of genomic mosaicism

In a previous study, a high pSNP count was observed in *S. cerevisiae* strains possessing mosaic genomes (i.e. resulting from a hybridisation event) (James et al., 2009). Comparing the two *Saccharomyces* species within the present study, it was observed that on average *S. cerevisiae* strains have 3.25 times more pSNPs in their rDNA arrays than *S. paradoxus* strains. The marked difference in these figures is principally due to the 15 *S. cerevisiae* mosaic strains, which account for over 78% of the pSNPs (245 out of 315) identified in this strain subset. In contrast, only 70 pSNPs were detected in the 19 *S. cerevisiae* structured strains, making this strain set comparable to *S. paradoxus* for that polymorphism type. This

results in the *S. cerevisiae* mosaic strains having 4.44 times more pSNPs in their rDNA arrays than the *S. cerevisiae* structured strains, which supports findings from our earlier analysis (previously measured as a 2.9 fold difference (James et al., 2009)). Furthermore, the Pearson's correlation coefficient was found to be $r = 0.713$ between pSNP count and population type (i.e. mosaic or structured) for this new dataset and $p = 5.15 \times 10^{-9}$ for the corresponding negative binomial regression, indicating the strong relationship between these two variables.

In addition to confirming the previous result, potential mosaic-like features were found in lineages previously identified as "clean". Based on pSNP occupancy, the five *S. cerevisiae* structured lineages identified by Liti *et al.* (2009) can be subdivided into two groups, subsequently referred to as *structured mosaic* and *structured clean* strains (Figure 3.14). In the original set of 15 *S. cerevisiae* mosaic strains, 60% of the detected pSNPs (145/245) were found to have occupancies greater than 10% but less than 90%. In theory, one scenario under which this type of pSNP could have arisen is if two parental strains from different populations/lineages, and with differing SNPs, crossed and produced a hybrid. Using the mid-occupancy class of pSNP as an indicator of genome mosaicism, the seven strains belonging to the Malaysian, North American and West African lineages were observed to have only two (out of 16) pSNPs with occupancies between 10% and 90%, classifying them as structured clean strains. In contrast, the majority of pSNPs (40 out of 54) in the 12 strains belonging to the Sake and Wine/European lineages have occupancies within the 10% to 90% range, showing mosaic-like behaviour and classifying them as structured mosaic strains (Figure 3.14).

Indeed, a re-examination of the Structure diagrams produced by Liti *et al.* (2009) and shown in figure 3.15, reveals that apparent genome mosaicism, albeit at a relatively low level, can be identified in some of the strains originally classified by these authors as having structured/clean genomes. For example, in the Sake lineage approximately 10% of the Y12 (palm wine strain) genome appears to have originated from three other lineages (i.e. Malaysian, West African and Wine/European). Eleven pSNPs were identified in this strain (Table 3.4), 10 of which have occupancy values of between 10% and 90%, supporting the possibility that this class of pSNP might prove useful as a potential indicator of cryptic genome mosaicism, perhaps the result of hybridisation events older than those leading to the standard mosaic class. As many of the structured mosaic strains have a fermentation origin (e.g. sake and wine), it is likely they have undergone some degree of hybridisation during their respective histories which has left a

residual signal within their genomes, including within their rDNA arrays.

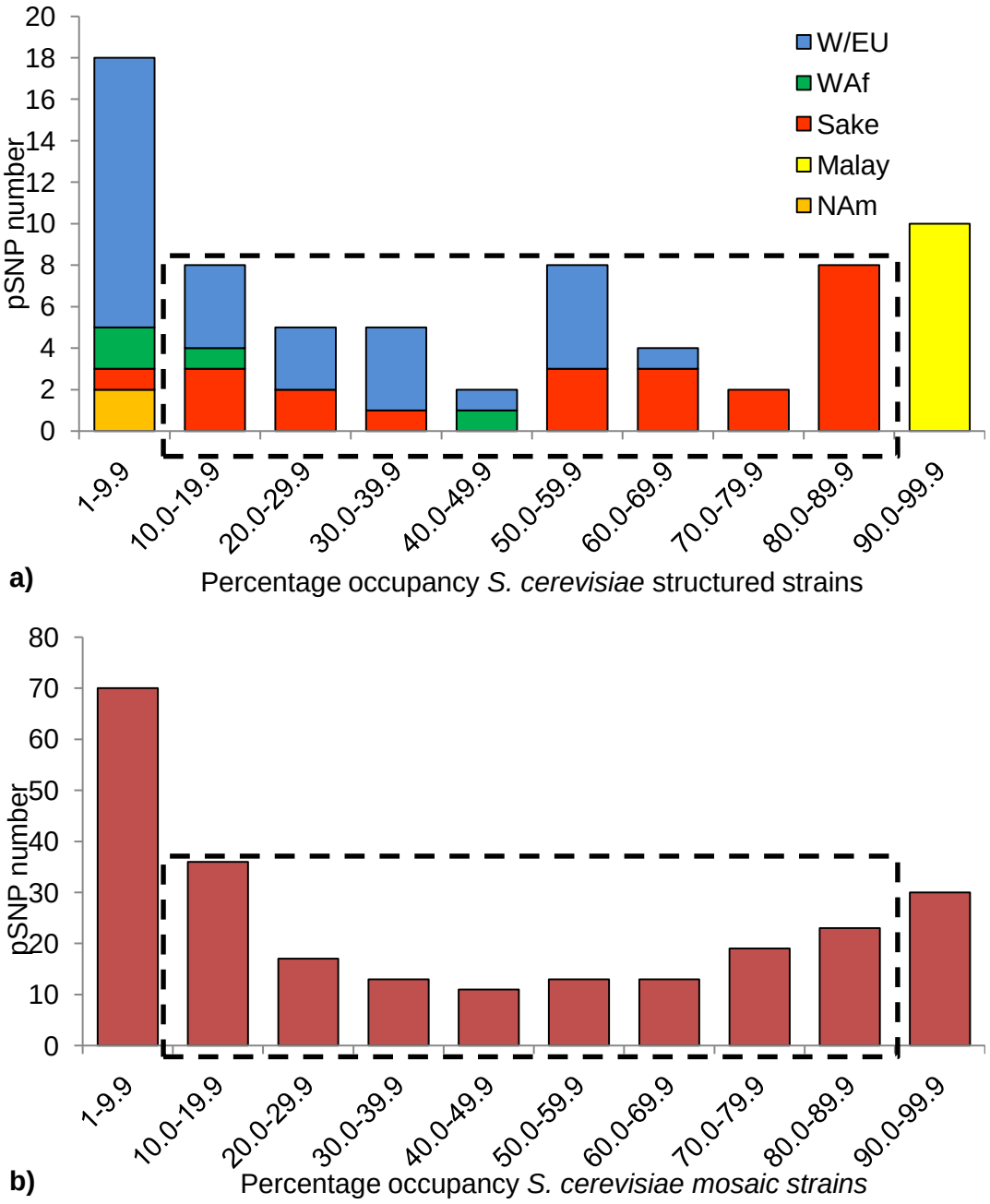

Figure 3.14.: a) Bar chart of the *S. cerevisiae* structured strains, with number of pSNPs against the pSNP occupancy. The boxed section highlights pSNPs with occupancies greater than 10% and less than 90%. The Malaysian, North American and West African strains have very few pSNPs within this boxed area, and these are denoted as clean structured strains. Those strains with a number of pSNPs within this boxed area show a degree of mosaicism, and are thus classified as being structured mosaic strains. b) Bar chart of *S. cerevisiae* mosaic strains, where there are a large number of pSNPs within the 10% to 90% occupancy range.

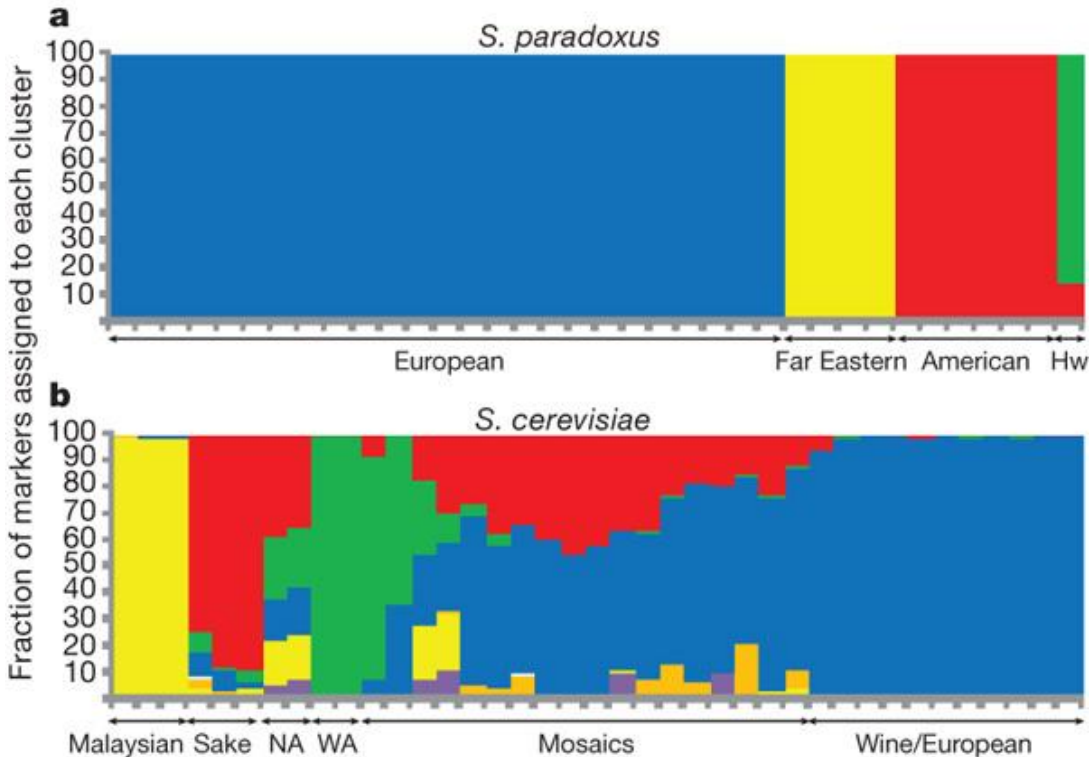

Figure 3.15.: Reprinted by permission from Macmillan Publishers Ltd: Nature (Liti et al., 2009), copyright 2009. a) Inference of population structure using the program Structure (version 2.1) on an *S. paradoxus* genome-wide SNP dataset. Each mark on the x axis represents one strain, and the blocks of colour represent the fraction of the genetic material in each strain assigned to each cluster. Hw, Hawaiian isolate, (not analysed in our study). b) Inference of population structure on *S. cerevisiae*. NA, North America; WA, West Africa.

3.2.7. rDNA Dynamics

A detailed characterisation of pSNP and SNP polymorphisms also provides insights into the dynamics of rDNA evolution and individual strain variation. 74.1% and 36.4% pSNPs were found to occur at sites of SNP variation in *S. paradoxus* and *S. cerevisiae* respectively, showing the clear relationship between the two variation types. Analysis of individual sites showed that variation could be seen "rippling" through a phylogeny, from regions of closely-related strains where the ancestral form was prevalent to more distant strains, where the variation could still be seen, but now either as a pSNP or fixed as a SNP. Indeed, the manner in which pSNPs can spread in this fashion through a group of related strains depends on several factors. One key factor is the size of the tandem array (i.e. the copy number), and another is the nature of relatedness between members of the group.

For the cases of *S. paradoxus* and *S. cerevisiae*, which differ both in copy number (averages 69 and 99 respectively per strain) and in relatedness of species members, major differences in features of pSNP and SNP variation would be expected, such as relative proportions of mutation types, and pSNP occupancy values. Indeed, this has been shown to be the case. In *S. paradoxus*, the majority of pSNPs (more than 90%) had an occupancy of either less than 10%, or greater than 90% (Figure 3.3c), and SNP variation is high within the species, suggesting that many previous pSNPs have become fixed. This is consistent with a species with a small copy number and treelike evolutionary structure, able to respond quickly to strong concerted evolutionary pressure. In *S. cerevisiae*, over half of all identified pSNPs (187/315) were found to have occupancies within the 10% to 90% range, and the number of pSNPs and SNPs were considerably higher and lower respectively than those seen in *S. paradoxus*. Indeed, this occupancy pattern would be consistent with a species with higher copy number where it had also been affected by significant quantities of hybridisation events.

Within both datasets, the distributions of related pSNP/SNP occupancies are found to be U-shaped (Figure 3.3c), though there are clear differences between the two curves. Indeed, this type of distribution is often observed in biological datasets, including both allele frequency (Chakraborty et al., 1980) and gene frequency (Haegeman and Weitz, 2012) datasets within populations, as predicted by mutation-drift theory. The datasets offer a fascinating snapshot of concerted evolution in action. For *S. paradoxus*, observing pSNP variation at a single point in time is a challenge, as the homogenisation process is rapid throughout this species. However, whole genome sequencing studies enable variation to be captured in low quantities. For *S. cerevisiae*, larger copy numbers and hybridisation between strains potentially increase mutation number and slow down the homogenisation process respectively, and so variation spreading across its strains can be captured more easily. However, other factors such as selection pressures can also affect the shape of the distribution.

Although some features of pSNP/SNP variation can readily be related to characteristics of their harbouring species, others are less obvious without a deeper understanding of strain origins and inter-relationships. From previous results on the consequences of genome mosaicism (James et al., 2009), *S. cerevisiae* strains with structured genomes could perhaps be expected to have pSNPs with similar occupancy patterns to the *S. paradoxus* dataset. However, the spread of pSNP occupancies is unexpectedly maintained when the *S. cerevisiae* results are split into mosaic and structured strains (Figure 3.14), as defined in previous

work (Liti et al., 2009). One explanation for this observation is the more variable nature of the structured *S. cerevisiae* strains, where each strain has been subject to differing levels of hybridisation, when compared to the wild *S. paradoxus* strain set. Indeed, as discussed in Section 3.2.6, the current classification of *S. cerevisiae* strains into mosaic and structured sets masks a wide range of variation, even within the latter grouping.

3.3. Coverage Across the rDNA Unit

3.3.1. Method

The coverage across the rDNA unit for each strain was calculated using a custom Perl script (coverage_v2.pl). This used the hit_series.out file from the TURNIP run for each strain, counting the number of reads which were hits in each 20 base pair window along the rDNA reference unit. These counts were written to a Microsoft Excel file and plotted as line charts, shown in Figure 3.16.

The rDNA copy number per strain was also estimated from these coverage results. The average read depth across the whole rDNA unit was calculated for each strain by averaging the number of reads in each 20 base pair window. This value was then divided by the genome sequencing depth for that strain in the SGRP user manual (SGRP, 2013), to give an estimate of copy number for each strain, shown in Tables 3.1 and 3.4.

Relationships between copy number and geographical or geographical/industrial group were assessed using correlation tests and linear models. In *S. paradoxus*, strains were grouped according to their geographical origin, with European, Far Eastern and American strains. In *S. cerevisiae*, strains were grouped according to their geographic/phylogenetic origin or their industrial usage, with mosaic strains W303, S288c, 322134S, 27361N, 378604X, DBVPG6040, YS9, NCYC361, YS4 and DBVPG1853 coded as *other mosaics*, structured West African and related mosaic strains NCYC110, DBVPG6044, SK1 and Y55 coded as *West African + related mosaics*, structured Malaysian strains UWOPS05-227-2, UWOPS03-461-4 and UWOPS05-217-3 coded as *Malaysian*, mosaic strain YIIc17E5 as itself, structured Sake strains Y9, Y12 and K11 coded as *Sake*, structured Wine/European strains L1374, DBVPG1106, DBVPG1788, YJM975, YJM978, YJM981, BC187,

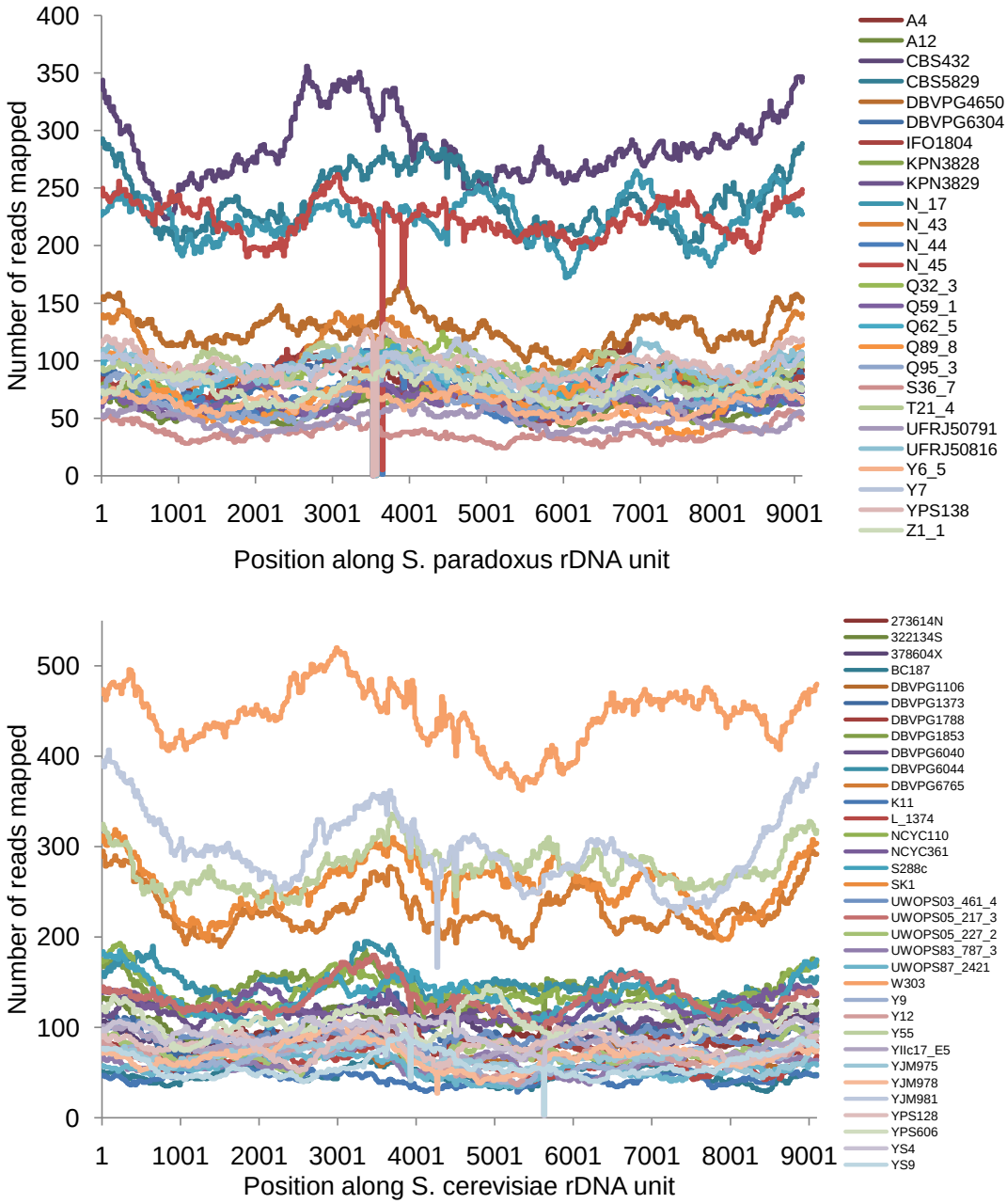

Figure 3.16.: The number of reads for each strain mapped to the representative rDNA unit. Top line chart refers to *S. paradoxus*, the lower to *S. cerevisiae*. In both datasets there are a small number of strains where there is no coverage, representing areas where there is either a great divergence from the consensus sequence, or an area of variation complexity.

DBVPG6765 and DBVPG1373 coded as *Wine/European*, structured North American strains YPS128 and YPS606 coded as *North American*, and mosaic strains UWOPS87-2421 and UWOPS83-787-3 coded as *UWOPS mosaics*.

3.3.2. Results

The sequence read coverage of each *S. paradoxus* strain, calculated by mapping reads to 20bp target windows of the relevant rDNA consensus sequence, was measured and found to range from 40 to 100 (see Figure 3.16), though with four strains (CBS432, CBS5829, N-17 and N-45) ranging from 180 to 360. Six American strains (A4, A12, DBVPG 6304, UFRJ 50791, UFRJ 50816 and YPS138) and two Far Eastern strains (N-44 and N-45) were found to possess small (up to 20 bp) sections within the ETS2/IGS1 region that were either not covered or very poorly covered. The ETS2 and IGS1 regions have been shown to display high quantities of polymorphism in *S. paradoxus* (Figure 3.3a and Table 3.3). When the 20 bp windows flanking these areas of poor/no coverage were examined in detail, it was discovered that all eight strains had SNPs, insertions and/or deletions on either side of these coverage anomalies (Table 3.7). This implied that any reads spanning these areas were either too dissimilar to the reference consensus to pass the BLAST or multiple alignment filters within the read mapping procedure, or else carried large deletions. For example, all six American strains have consistently no coverage over one specific area (ETS2 region, positions 3520 to 3540, and 3560 to 3580), which would appear to be a feature of this group and its diversification from the (European) type strain. A similar coverage analysis of the 34 *S. cerevisiae* strains was also carried out. Here, most strains fell within the range 40 to 200, although five strains (W303, Y55, YJM981, SK1 and DBVPG6765) were found to possess a sequence read coverage of between 200 and 480. Two strains (the closely related YS4 and YS9) both exhibited only two or three mapped reads in a single small area, positions 5620-5639.

Strain	Region	No. of reads	I.D of hits	20bp before	20bp after
A4	3560- 3579	0	n/a	4bp deletion and 8 SNPs	3 SNPs and 4bp deletion
A12	3560- 3579	2	A12-10i08.q1k, A12-8n21.q1k	a 5bp and 2bp deletion, and 2 SNPs	3 SNPs and a 5bp deletion

Strain	Region	No. of reads	I.D of hits	20bp before	20bp after
DBVPG6304	3560- 3579	3	DBVPG6304-22k14.q1k, DBVPG6304-4b20.q1k, DBVPG6304-36c11.q1k	5bp insertion and 8 SNPs	3 SNPs and a 5bp deletion
UFRJ50791	3560- 3579	2	UFRJ50791-1c04.p1k, UFRJ50791-6o15.q1k	5bp insertion and 8 SNPs	3 SNPs and a 3bp deletion
UFRJ50816	3560- 3579	0	n/a	5bp and a 2bp deletion, and 4 SNPs	3 SNPs and 4bp deletion
YPS138	3560- 3579	1	YPS138-32h06.p1k	5bp insertion and 8 SNPs	3 SNPs and a 5bp deletion
A4	3520- 3539	0	n/a	4bp insertion	4bp deletion and 8 SNPs
A12	3520- 3539	1	A12-13l16.p1k	2 pSNPs, partial ins (93% 4bp insertion and the other 7% a 4 bp and a 1 bp)	a 5bp and 2bp deletion, and 2 SNPs
DBVPG6304	3520- 3539	0	n/a	SNP and 4bp insertion	5bp insertion and 8 SNPs
UFRJ50791	3520- 3539	0	n/a	5bp insertion	5bp insertion and 8 SNPs
UFRJ50816	3520- 3539	4	UFRJ50816-25h24.q1k, UFRJ50816-6m03.p1k, UFRJ50816-28e01.q1k, UFRJ50816-22m15.q1k	4 pSNPs and 3 partial insertions	5bp and a 2bp deletion, and 4 SNPs
YPS138	3520- 3539	0	n/a	4bp insertion	5bp insertion and 8 SNPs

Strain	Region	No. of reads	I.D of hits	20bp before	20bp after
N_44	3640- 3659	1	N_44-19m04.p1k	9 pSNPs (most of which are complex), 16 bp deleted, have been split into 7 possible, again some of which are complex	none
N_45	3640- 3659	5	N_45-60a03.q1k, N_45-60f08.q1k, N_45-8i09.p1k, N_45-60g10.p1k, N_45-60d08.q1k	7 pSNPs, most of which are complex, 1 partial insertion, 16 bp deletion, split into 7 possible, most of which are complex.	none

Table 3.7.: *S. paradoxus* strains which had little or no coverage for small rDNA regions, and an analysis of the regions surrounding the anomalies

The number of rDNA repeats (copy number) in each *S. paradoxus* strain was estimated by comparing the coverage of the rDNA repeat consensus unit to the coverage of the whole genome. The estimated copy number was calculated for each strain and was found to range from 45 (American strain A12) to 96 copies (Far Eastern strain IFO 1804) (see Table 3.1), with an average of 69 copies per strain. These estimates were found to be lower and less variable than for *S. cerevisiae*, where estimated rDNA copy number ranged from 50 (strain K11) to 354 copies (strain YJM981) (see Table 3.4), with an average of 99 copies per strain. No significant correlation between rDNA copy number and geographical origin was identified in *S. paradoxus*, with Pearson's $r = -0.292$. Furthermore, a Negative Binomial regression of copy number on geographical origin did not give a significant result at the 5% level in the resulting z-tests on geographical factor levels or in a Chi-squared analysis of deviance test, with $p = 0.283$ for the latter (Figure 3.17a). In *S. cerevisiae*, although no significant correlation was initially

discovered between rDNA copy number and either geographical/industrial group or strain type (structured or mosaic), with $r = -0.257$ and $p = 0.063$ respectively, a clear relationship between copy number and geographical/industrial strain group could be observed (Figure 3.17b). On removing the outliers YM981 and DBVPG1106 (both Wine/European strains, with 354 and 98 copies respectively), the strong relationship between rDNA copy number and geographical/industrial strain group ($r = -0.634$) and between copy number and strain type became apparent ($r = 0.310$). Furthermore, a Negative Binomial regression of copy number on geographical/industrial group indicated that the group was an important factor in the model ($p = 1.81 \times 10^{-5}$) and that Sake, Wine/European, North American and UWOPS mosaics groups were significantly different from the other mosaics group (with $p = 0.001$, $p = 8.32 \times 10^{-6}$, $p = 0.003$ and $p = 0.001$ respectively, Figure 3.17b). A Negative Binomial regression of copy number on strain type also showed this factor to be significant, with $p = 0.041$. In conclusion, although *S. cerevisiae* mosaic genomes tend to possess a higher rDNA copy number than structured genomes, there are exceptions to this trend. For example, a low copy number was observed amongst the UWOPS mosaic strains and a high copy number amongst the West African structured strains. However, a strong relationship exists between phylogenetic grouping and copy number in *S. cerevisiae*, but not in *S. paradoxus*, and it would be interesting to determine the factors driving copy number evolution in future studies. In a very recent study (Long et al., 2013) "massive genomic variation" in 180 lines of the model dicot plant *Arabidopsis thaliana* was uncovered, ∼90% of which was attributed to copy number variation of repetitive sequence, with 45S rDNA the largest contributor by far. Interestingly, the observed variation was found to be strongly correlated to geographic pattern.

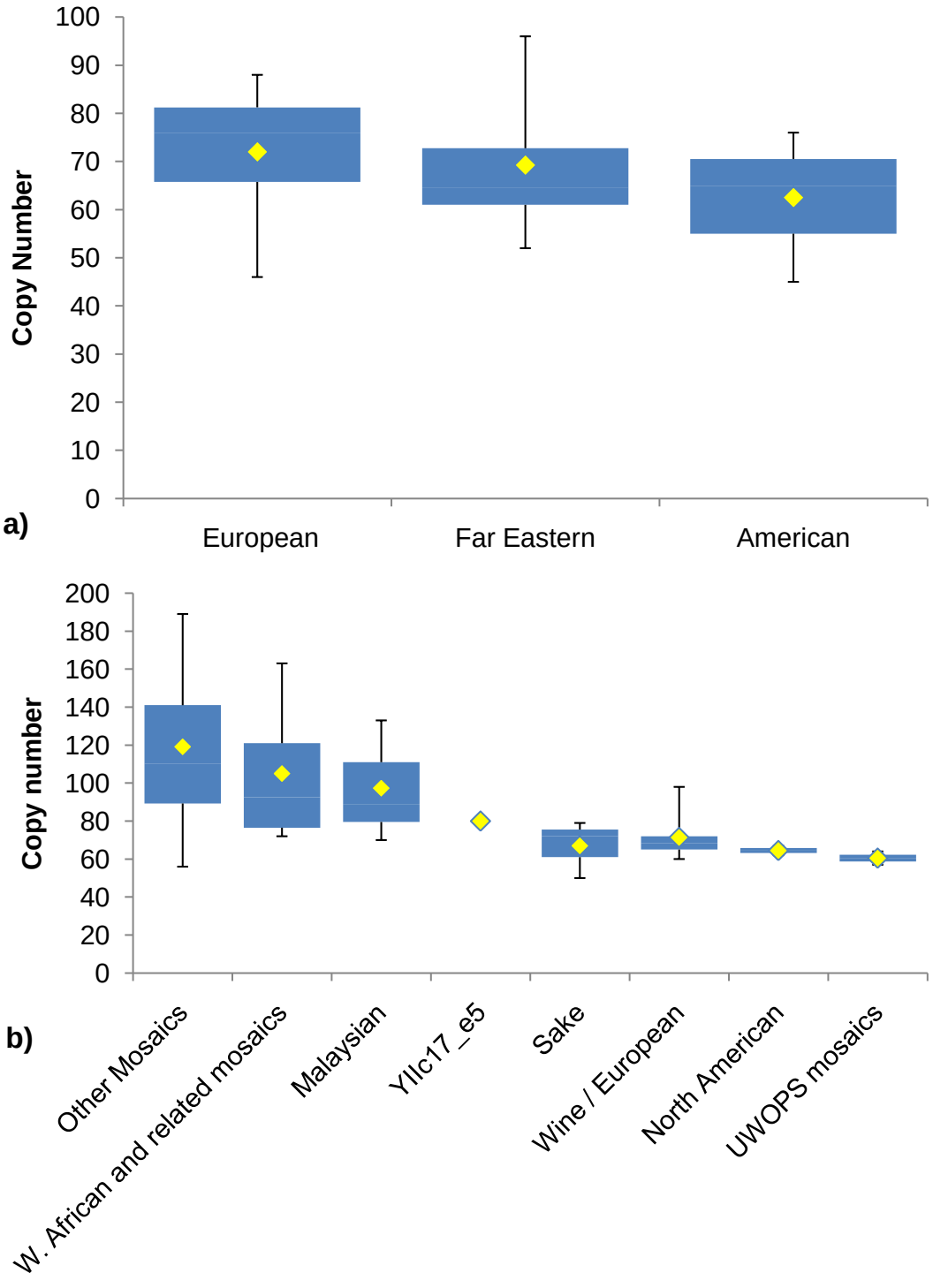

Figure 3.17.: a) Box plot of *S. paradoxus* geographical groups and their copy numbers b) Box plot of *S. cerevisiae* groups and their copy numbers, excluding outlying strains YJM981 and DBVPG1106

3.4. Putative hybrid origins of *S. paradoxus* strains N-17 and N-45

The majority of *S. paradoxus* strains show no strong evidence of mosaicism when looking at pSNP counts. In a previous study, Liti *et al.* (Liti et al., 2009) identified only one candidate *S. paradoxus* strain (UW0PS91-917.1) as having a potential mosaic-like genome. This strain, isolated from Hawaii (flux of *Myoporum sandwichense*), was not included in the current study as preliminary analysis of its rDNA sequence reads had revealed potential contamination with *S. cerevisiae* sequences. However, the European strain N-17 (from Russia) and the Far Eastern strain N-45 (also isolated in Russia, albeit in the eastern region of the country) are atypical of *S. paradoxus* strains in that they possess high numbers of pSNPs (Table 3.1), collectively totalling 72.6% of all pSNPs in this dataset.

Strain N-17 was earlier revealed to possess 17 pSNPs within its rDNA array, 16 at low occupancy, by far the most polymorphisms (29) of any of the European strains (Table 3.1). Despite this, in the pSNP- and SNP-based phylogenetic tree, N-17 is clearly shown to belong to the mainland European population which also includes the reference strain CBS 432 (Figure 3.10). Further examination of these pSNPs, and the strains that share them as pSNPs, SNPs or putative ancestral states (Figure 3.18), showed that 10 are shared with only Far Eastern (or Far Eastern and American) strains, albeit at low frequency. A further 6 are unique to N-17 alone and the remaining pSNP is shared with a single Far Eastern strain and a single European strain. The most likely hypothesis to reconcile this set of variation is that N-17 is the result of a hybridisation between a European and a Far Eastern strain. It is interesting that N-17 possesses many unique pSNPs, potentially indicating that at least one of N-17's parents is not found within the existing strain set.

The Far Eastern strain N-45 was found to possess 36 pSNPs, and like N-17 is slightly distinct from the rest of its group on the neighbor-joining tree (Figure 3.10). Again examining the co-occurrence of pSNPs across the strain set showed that 32 are shared only with Far Eastern (or Far Eastern + N-17) or Far Eastern and American (or Far Eastern + American + N-17), this time at high frequency. A further two pSNPs are unique to N-45, one is shared with an American strain and one is shared with two European strains. Again, this set of variation indicates that N-45 is a putative hybrid of a Far Eastern and a European strain (i.e. the low-frequency components of the majority of N-45's pSNPs indicate a European

origin).

Examination of the NeighborNet for *S. paradoxus* (Figure 3.13) shows a clear phylogenetic conflict implicating the American strain UFRJ50791, with a large box-like structure. Further examination of the source of this conflict shows that it derives from incompatible sharing of SNPs between different subsets of strains within the American group, with one explanation being a recent intra-group hybridisation. It is interesting to contemplate the clarity of this SNP-based conflict with our two putative pSNP-based mosaics, which are largely invisible on the NeighborNet. Further research could be carried out to determine whether pSNP-based conflicts can be easily identified using such tools or whether this is simply a consequence of potentially old events exhibiting low-frequency pSNPs in this particular case.

Three of the four Far Eastern strains (N-43, N-44 and N-45) were found on the same continental land mass as six of the European strains (CBS 432, CBS 5829, DBVP 4650, KPN 3828, KPN 3829 and N-17) (Figure 3.1). Furthermore, all of the *S. paradoxus* strains in these areas were isolated either from oak tree bark or exudate. The existence of a region in mainland Europe (perhaps Russia) where European and Far Eastern strains coexist is therefore a possibility, with such a region a potential source of hybrid strains. While further research would be needed to confirm the N-17 and N-45 hybridisations, the potential to identify hybridisation signals from features of rDNA polymorphisms, in organisms with population structures similar to *S. paradoxus*, is intriguing.

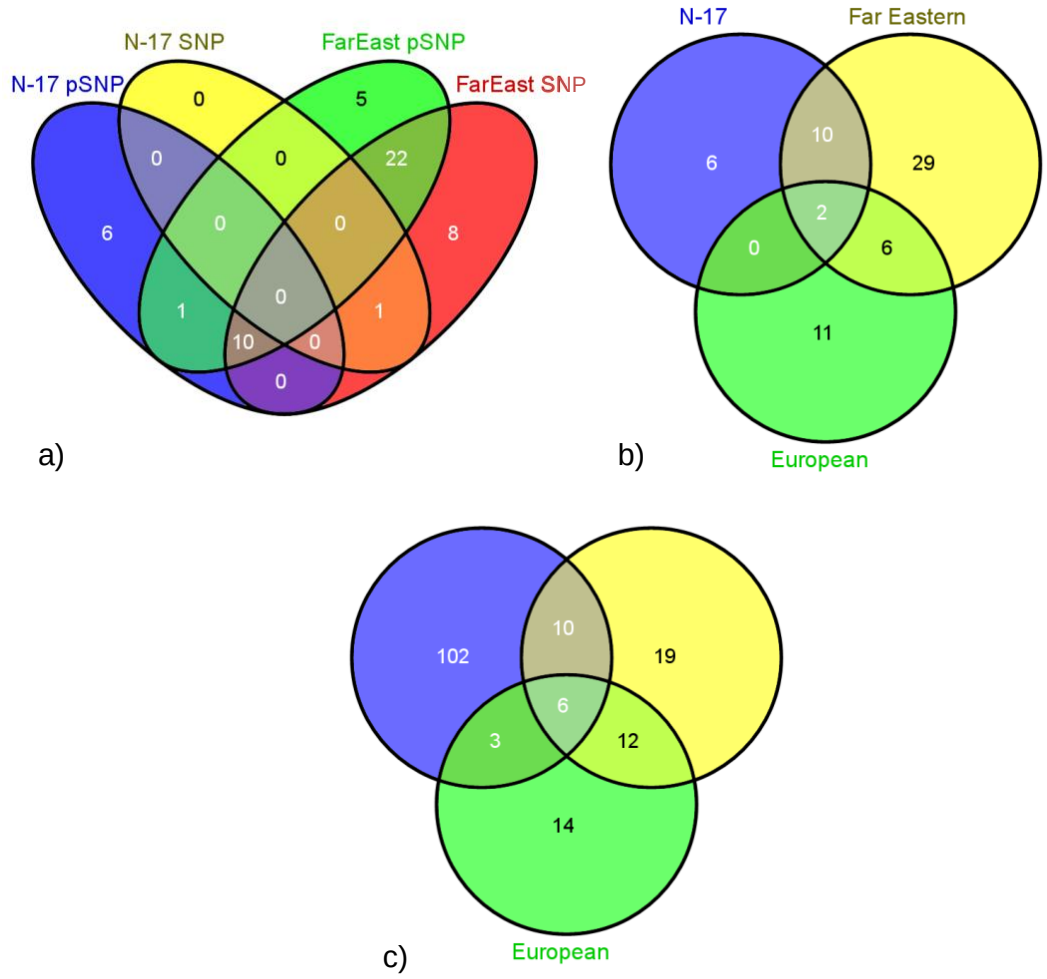

Figure 3.18.: a) Venn diagram of the different pSNP and SNP positions in strain N-17 in comparison to the Far Eastern strains. 11 of N-17's pSNPs are in the same position as pSNPs or SNPs in one or more Far Eastern strains. b)Venn diagram of the pSNP + SNP positions in N-17 compared to the Far Eastern and European strains. 10 sites of variation overlap with the Far Eastern strains alone, and 2 are present in all groups. c) Venn diagram of the overlap of pSNP + SNP positions between the three different geographical groups, and the number of pSNP or SNP positions that are unique for each group.

3.5. Conclusions and Chapter Summary

A thorough and detailed analysis of rDNA sequence variation in two contrasting yeast species was achieved. The data were cleansed and examined carefully before analysis, and the program tested to ensure results were as accurate as possible. Unlike many preceding studies, the analysis has encapsulated variation across the whole rDNA unit and has examined a broader range of polymorphism types across a larger strain set. The resulting datasets have therefore enabled deeper insights into inter-strain relationships within wild and domestic yeast, and to observe important differences in the manners in which these two species have evolved.

Within each species, the uncovered variation was shown to be substantial in size and rich in evolutionary information. Collectively each dataset follows a U-shaped distribution of allele frequencies predicted under mutation-drift evolutionary theory. The datasets have been used to successfully infer complex lineage relationships between strains, at a fine-scaled phylogenetic resolution, and these inferences have been shown to be consistent with existing knowledge. From this we further infer that large-scale sequencing of the rDNA locus can overcome at least some of the documented problems with phylogenetic inference deriving from its use, making its many advantages more prominent. While the rDNA coverage of the SGRP datasets was moderately high, deeper sequencing is now possible at a reasonable cost and it will be interesting to compare whether greater depth leads to better estimates of polymorphism counts and therefore more accurate phylogenies. In future, it would be interesting to test formally whether pSNPs within the rDNA array (or indeed from other repetitive genomic sequences known to be moulded by concerted evolutionary processes) have greater power to discriminate between organisms within species than SNPs.

Key differences have also been noted in polymorphism proportions, pSNP occupancies, rDNA copy numbers, and variation patterns across the rDNA unit between the species. Furthermore, some of these differences have been linked to the species' contrasting population structures. This is important, because it may be possible to extrapolate this understanding to studies of other species in the future. In the case of yeast, where a high frequency of genome mosaicism is inferred in one species, *S. cerevisiae*, but not the other, *S. paradoxus*, it is compelling to speculate that the mosaic strains in *S. cerevisiae* may be linked to human traffic and/or industrial processes whereas they are considerably less likely to arise in the wild *S. paradoxus*. It is also of note that the hybridisation patterns

inferred are not inconsistent with the geographical locations of these strains.

It has been hypothesised that some variation patterns may be used to infer key genomic events. In particular, the numbers of pSNPs across the rDNA array and the distribution of individual pSNPs across strains may be used to identify putative hybridisation signals. While this pattern has been previously observed in *S. cerevisiae*, it has been refined here to hypothesise the first *S. paradoxus* mosaic strains (N-17 and N-45) and, in conjunction with pSNP occupancy values, to pinpoint potentially older hybridisation events in *S. cerevisiae*. Furthermore, it has been shown that many of our inferences are consistent with Structure analyses of genome-wide SNP datasets gleaned from these strains, confirming the value of pSNPs as a predictor of genome structure.

This analysis has captured and characterised detailed snapshots of two yeast species undergoing both similar (concerted evolution, homogenising the sequences within the rDNA unit) and contrasting (levels of genome hybridisation) evolutionary processes. It would be interesting to learn more about these processes by modelling them mathematically. The datasets developed here are a major step in achieving successful models that can fully exploit the rich source of evolutionary knowledge held within the rDNA array. Given the ubiquity of this genomic region, the prospect of using such models to analyse the genomes of a wide range of species is an attractive one. In the next chapter, early work in formalising concerted evolutionary processes computationally will be described, and the resulting software will be used to understand more about the dynamics of concerted evolution in the rDNA unit.

4. Simulating rDNA Evolution using the SIMPLEX Software

Chapter Abstract

To investigate how a single point mutation might be spread throughout or be lost from an rDNA array, a program, SIMPLEX, was developed in the Java programming language to simulate the concerted evolutionary process. The development and testing of this program is discussed. The fate of individual pSNPs within an rDNA array is followed using SIMPLEX, and the effect of two concerted evolutionary processes upon the simulations is examined. Finally, insights into the process of concerted evolution and its constituent mechanisms gained from these preliminary analyses is discussed.

4.1. Background and Outline

The rDNA tandem array is believed to evolve through the process of concerted evolution which over time homogenises the sequences between array units, though as is now known, not perfectly. The term concerted evolution relates here to the observation that tandem rDNA units are uniform in sequence yet this sequence can change over time, and so repetitive units evolve "in concert" (Eickbush and Eickbush, 2007).

Two key mechanisms have been implicated in this process: Unequal Sister Chromatid Exchange (USCE) and Gene Conversion (GC). The relative contributions of these two events and their exact modes of action are not yet known.

Gene Conversion

Gene conversion is a non-reciprocal transfer of DNA and does not result in a change in the tandem array size, as events involve an overwriting or "copy paste" of one unit with another. Many possible mechanisms underly gene conversion in tandem arrays, such as Synthesis Dependant Strand Annealing (SDSA) or Double Strand Break Repair (DSBR). A simplified mechanism is shown in figure 4.1a. Unlike USCE, gene conversion can act between chromosomes, making it a potential mechanism for homogenisation between rDNA arrays on different chromosomes (Eickbush and Eickbush, 2007).

Unequal Sister Chromatid Exchange

USCE involves crossing over between sister chromatids that are not precisely aligned, and a non-reciprocal exchange of DNA, resulting in chromatids of unequal length. This change in chromatid size leads to USCE being experimentally visible. As a consequence, this process has been implicated in concerted evolution since the 1980s when a gene inserted into the rDNA array, *LEU2*, was shown to be unstable and was lost from the rDNA array due to USCE (Petes, 1980). USCE either involves an increase in chromatid size due to a duplication of one or more units, or a decrease due to a deletion, as illustrated in figure 4.1b.

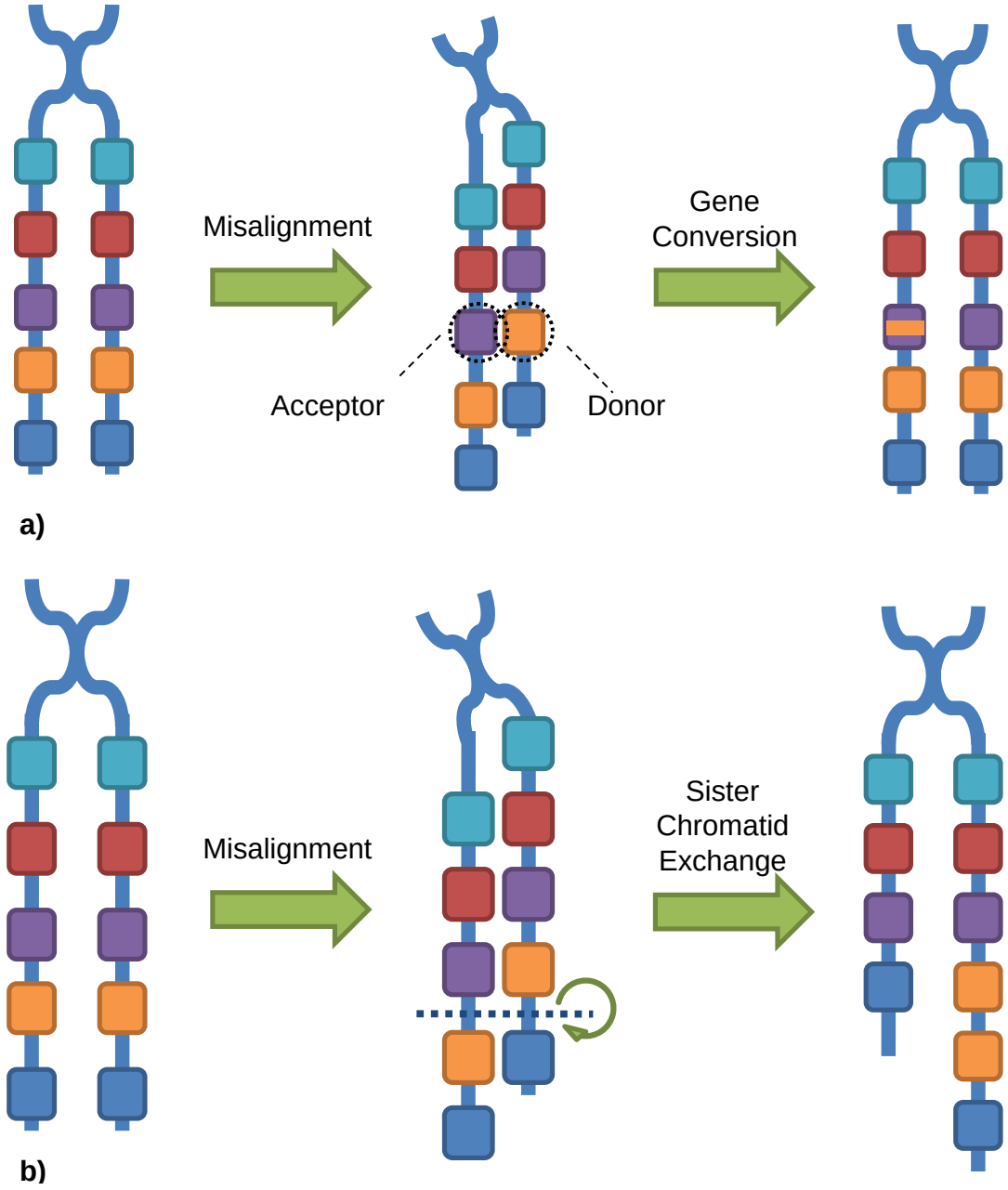

Figure 4.1.: Overview of the two main processes implicated in concerted evolution. a) Gene Conversion, b) Unequal Sister Chromatid Exchange

Mathematical Models of Concerted Evolution

As noted in Chapter 1, there have been many attempts to mathematically and computationally model the processes involved in concerted evolution, including:

- **Smith** – simulated random unequal crossover computationally, showing that a repetitive sequence would always be generated and maintained by this process when the sequence was not under selection (Smith, 1976).

- **Ohta** – produced mathematical models of fixation of mutations by random crossovers (Ohta, 1976).
- **Nagylaki and Petes** – modelled intrachromosomal gene conversion, showing that it would be possible to maintain sequence homogeneity with this mechanism alone. In this model, all units were deemed to be equally likely to be involved (Nagylaki and Petes, 1982).
- **O'Kelly** – modelled Unit Recombination Events (URE's), tried different models and showed a non-uniform recombination model best fitted observed data (O'Kelly, 2008).

These models were mainly concerned with inferences about the mechanisms from experimental evidence of the end products of concerted evolution, namely an already homogenized array. The identification of pSNPs provides a snapshot of concerted evolution in action, enabling new models of rDNA evolution to be developed, and improving understanding of the mechanisms involved. Furthermore, model-based analysis of pSNPs in large scale genomic and metagenomic datasets will facilitate fine-scale phylogenetics and provide a new approach to understanding strain and microbiome dynamics. In order to achieve this, a sensible first step is to develop a simulation program against which to compare results of computational analyses of pSNPs in large-scale datasets, such as those produced in Chapter 3.

4.2. The SIMPLEX Tool

A simple simulation tool to simulate the evolution of an rDNA tandem array was designed and developed using the Java programming language . The tool provided preliminary results on how concerted evolution moulds an array over time, which was then built upon by adding more complexity to the mutation events.

Initial simulation runs focussed upon mitotic USCE and GC events, tracking the spread of a single pSNP through the array to eventual fixation or loss. The number of pSNPs within the array are recorded after every event, and may be easily plotted. The mechanisms involved in concerted evolution are simplified to involve a single chromatid only. Furthermore, the program does not detail finer intricacies such as distinguishing between different types of gene conversion (i.e. SDSA or DSBR). In future, increasingly complex selection/mutation layers and array size variation could be added (Ide et al., 2010).

The Java tool, called SIMPLEX (**SIM**ulating **P**artial SNPs **L**oss or **EX**pansion), simulates simplified mechanisms of GC and USCE as described in the following sections. To start investigating these processes a clear definition of those mechanisms which are involved, and those which are excluded due to the need for simplicity, is required.

4.2.1. Assumptions and Parameters

Processes which are simulated by the program, and assumptions made, include:

- **Mitotic events only**
- **USCE** - intra locus crossing over. Ignore equal sister chromatid exchange which will not affect the sequence
- **Gene Conversion** - based on those used in the double-stranded break repair system

Processes not included in the program:

- **Meiotic events** - there is a 70- to 100-fold suppression of meiotic recombination between rDNA arrays on homologous chromosomes (Casper et al., 2008). As a result, meiotic recombination will not contribute as much to observed variation as mitotic events, and therefore it is sensible to exclude it from preliminary work.
- **Horizontal gene transfer, inter strain recombination or hybridisation** - inter strain recombination is not considered
- **Gene conversion - frequency of accompaniment with crossover** - gene conversion can be associated with a crossing over event, but crossing over is only condiered here as part of USCE events
- **Extrachromosomal rDNA circles, or ERCs** - ERC's are formed by homologous recombination (Gangloff et al., 1996; Johnson et al., 1999), but the program will not include their formation, or possible interaction with the array

In SIMPLEX, gene conversion is treated as a non-crossover outcome of a USCE event, therefore allowing the same parameters (and some of the underlying computer code) to be used for each process. This treatment assumes both mutations occur after a double-stranded break repair event, where the two sister

chromatids can misalign by a set amount. In USCE the chromatids crossover, resulting in a change to a number of units, whereas in gene conversion only a tract (of the size used as a template to repair) is changed and the chromatids do not crossover, as shown in figure 4.1. A gene conversion tract is the sequence which is copied across to the donor, sizes vary but are less than a unit in size.

Specific values chosen for initial parameters include:

- GC tract size initially fixed to be 4000bp, the lower estimate from a paper investigating this phenomenon (Judd and Petes, 1988).
- USCE tract to be between 1 and 10 units in length, from research on the *LEU2* locus (Szostak and Wu, 1980).
- GC donor and acceptor distance = USCE tract size – misalignment distance, as these are assumed to be different outcomes of the same process.
- Double-strand breaks are equally likely to occur anywhere within a unit.
- USCE is initially set to occur 20% of the time, as this is the observed ratio of the crossover product in a study looking at mitotic DSBR events in yeast (Nickoloff et al., 1999). This ratio can be varied.
- The limits of the array size are set to the approximate minimum and maximum calculated in different strains of yeast (∼70 and 210 units, see table 3.4 in Chapter 3 and (James et al., 2009)).
- The starting number of units is 140, the estimated number of units for *S.cerevisiae* (Eickbush and Eickbush, 2007). Similarly the number of base pairs in a unit is 9138, the size of an *S.cerevisiae* rDNA unit.

4.2.2. SIMPLEX Program Overview

An ArrayList of a specified size is created to represent an array of rDNA units. A typical size in yeast is ∼140 units. Only one ArrayList is needed to represent an rDNA array on one chromatid, as eventual homogenisation between sister chromatids is assumed. The simulation run begins with one polymorphic unit in the array (a single pSNP), the rest will be referred to as consensus sequence. The ArrayList is composed of Integer objects (a series of numbers), where each object represents a single rDNA unit. Each object will either take the value -1, to represent a consensus unit, or a number to represent the base position of a pSNP within the rDNA unit. The time until fixation or loss of a polymorphic rDNA unit will be tracked within a single simulation. The two different concerted evolutionary

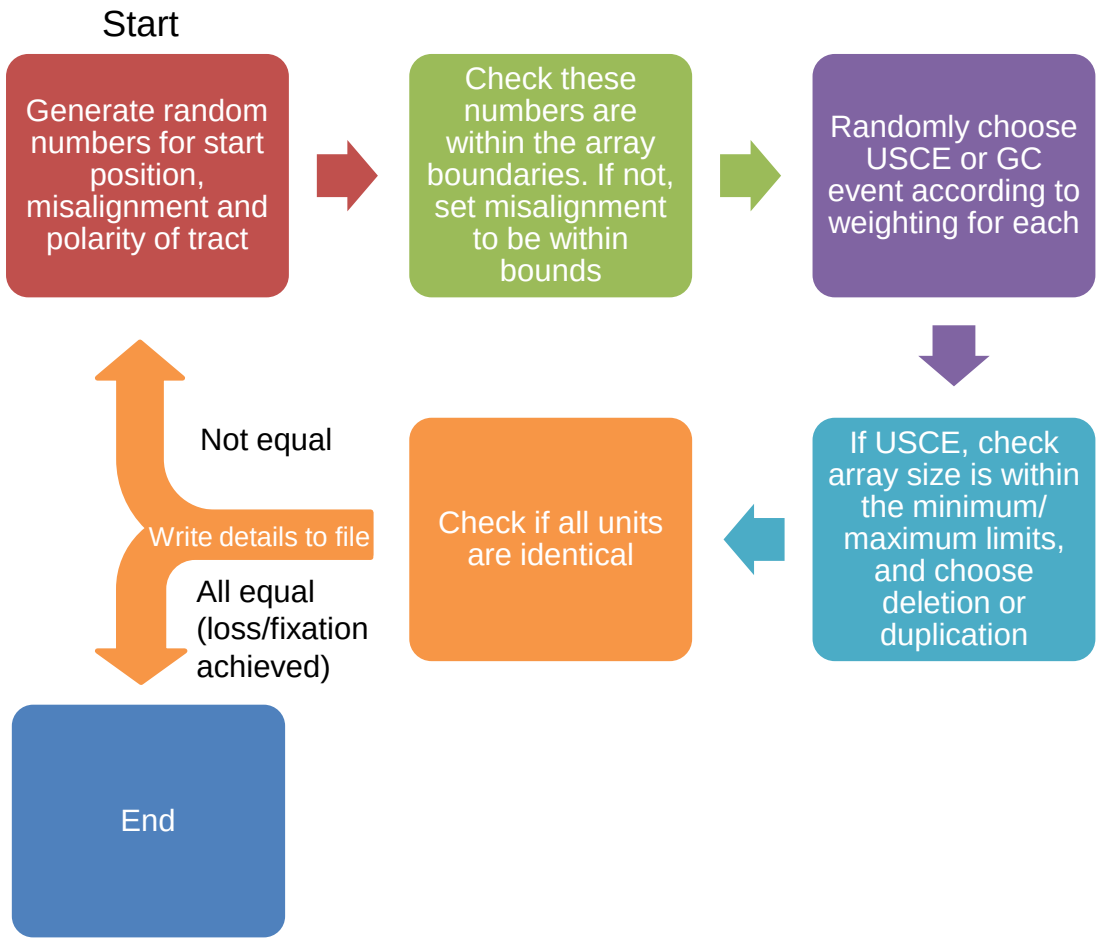

Figure 4.2.: Overview of each iteration of a simulation in the SIMPLEX program

mechanisms are run by calling their equivalent methods within SIMPLEX, in varying proportions which can be specified. Only USCE events can change the size of the chromatid array by duplicating or deleting units. The program initially assumes a basic, simplified model of each process, and each unit involved in a concerted evolutionary event is chosen entirely at random.

An overview of the loops used within SIMPLEX for each simulation, following an initial pSNP until loss or fixation, is shown in figure 4.2. At the start of each loop within a simulation, an rDNA unit is chosen at random from which to start the event, as well as a random number of units to be involved (equivalent to the misalignment of chromatids), shown in the red box in figure 4.2. The polarity of the tract is also chosen, in other words whether the units involved in an event are a certain number of units upstream or downstream the starting unit. A check is carried out to ensure all of the units involved are within the array boundaries, and the misalignment value is re-generated until they are not. For example, if unit 100 is randomly chosen as the starting unit, and the misalignment or tract

Variable	Description
numSimulations	Number of simulations to run
percentUSCE	Percentage of events that will be USCE
unitSize	Number of bases in one rDNA unit
startingArraySize	Number of elements in the array at the start of a simulation
maxArraySize	Maximum limit of array size
minArraySize	Minimum limit of array size
elementStartpSNP	Element in array possessing the initial pSNP
SNP	Base position within a unit which is a SNP
maxMisalign	Maximum number of units which can misalign
gcTract	Number of bases that are copied in a GC event
name	Name of the output files
maxIterations	Maximum number of iterations allowed in a single simulation, used if limiting run lengths

Table 4.1.: List of static variables in SIMPLEX. gcTract is static in this version of the software.

size is 6 units, but the array is only currently 102 units long, the misalignment value must be changed to be within the array size (in this case to 2 units). A method is then called to carry out the USCE or GC event on the array, using the above parameters. The likelihood of each type of method being called is set globally. If the method is USCE, a limit exists for the minimum and maximum number of units allowed in the rDNA array, and the units chosen at random need to maintain the array size within these bounds. Consequently, a method must check this and alter the relevant values if necessary. Finally a method is called to check if the most recent event has homogenized the array, so that the pSNP has either spread to all units or has been lost in all units. If this is the case, results are written to file, and a new simulation starts. If the units are not identical, the actions of that iteration of the simulation are written to a file and the loop starts again.

A number of parameters are set at the beginning of a simulation run, which will be identical for each simulation within it. These static or constant variables are shown in table 4.1. They include the percentage of USCE events, and the position in the unit at which the pSNP is located. In this program, to reduce complexity, the size of the sequence tract involved in GC events will be static. Lastly, a maximum number of iterations (one iterative cycle is shown in figure 4.2) can be set, to limit the run time of the program.

The methods that simulate USCE and GC events within an rDNA array will now be described in more detail.

Unequal Sister Chromatid Exchange Method

The USCE method can simulate deletions and duplications, both of which will effect the size of the rDNA array (see figure 4.3 and 4.4). The program follows the fate of a single chromatid, which can either grow or reduce in length with each event (figure 4.3b). Deletions and duplications are assumed to occur at the same frequency (Ganley and Kobayashi, 2011), so these are chosen randomly in the USCE method using a boolean (true or false) value.

The method requires the following information, with reference to figure 4.4 to how this relates to the array:

- An initial unit is chosen at random (the green unit)
- A point within this unit to start a break (the start of the dotted box)
- A misalignment value which will be the number of units involved in the exchange (the length of the dotted box, in this case 5 units)
- From these three values the last unit in the tract (the orange unit, 5 units from the green) and the point at which the tract ends (the end of the dotted box) are calculated
- The tract between the breaks will be duplicated or deleted (with all sequence within the dotted box copied or removed).
- The initial and last unit could change identity dependent upon the position of the break and the position of the pSNP (the green and orange split box)

Figure 4.4, where a pSNP is represented as a purple cross, illustrates how pSNPs are reduced or increased in number throughout the array by this process.

The ArrayList structure in Java has in-built methods to remove objects or to copy objects and insert them in specified element positions within an ArrayList, which is essentially the basis of USCE. This makes ArrayLists an ideal data structure to represent an rDNA array. However, a composite unit will be produced for each iteration of each simulation run (seen as the split orange and green unit in figure 4.4). The identity of this unit (whether it has a pSNP or it is consensus) is dependant on the identities of the two original units of which it consists, the break position used in the tract, and the position of any pSNPs present. The process

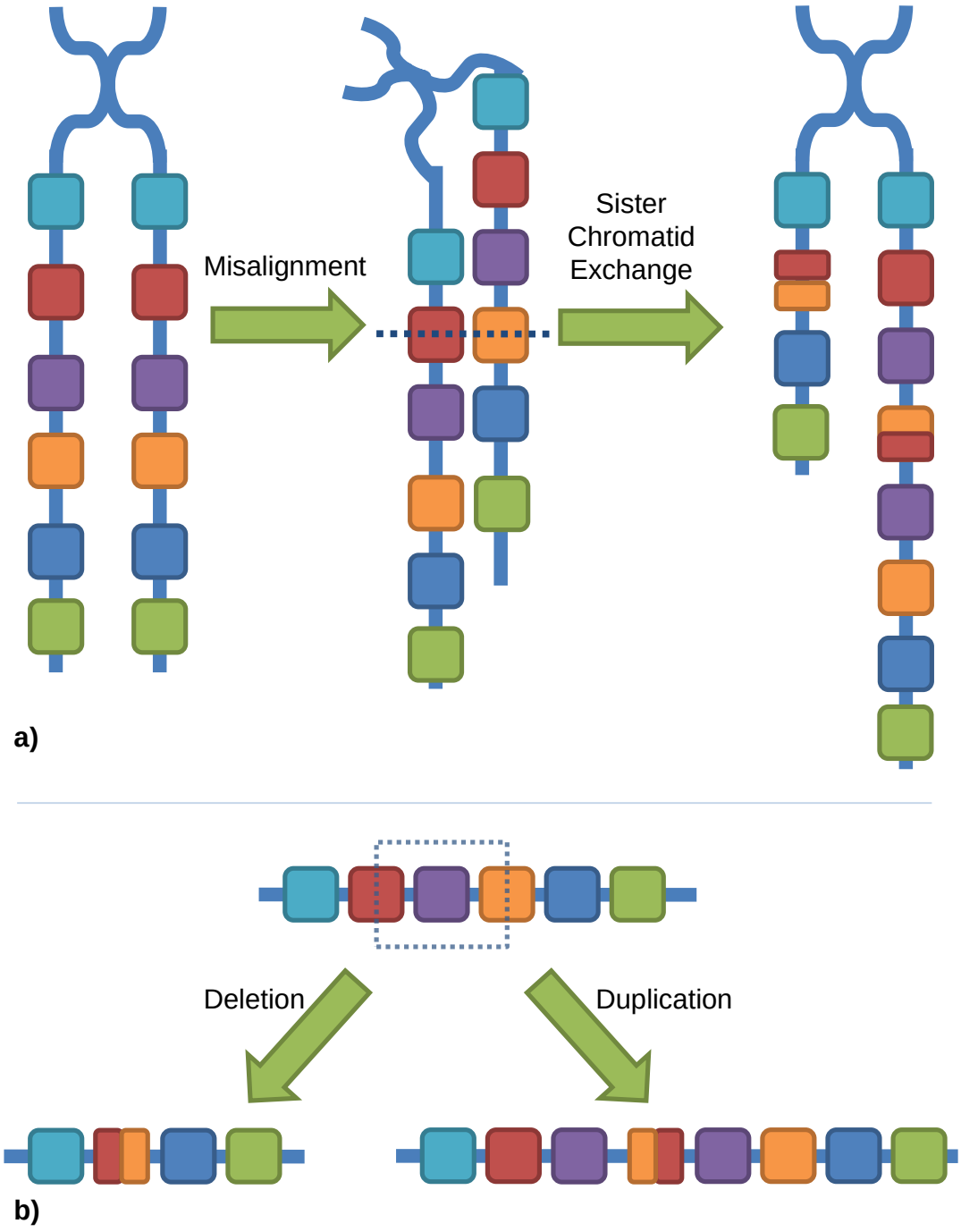

Figure 4.3.: Representation of USCE events in an rDNA array a) representation of a USCE event, involving a misalignment of 2 units with the two sister chromatids crossing over. b) representation of the same event as in a, except looking at the fate of one chromatid only. In this case one chromatid would show a duplication event, and the other a deletion. Note that in the deletion event the first unit in the tract changes (now red/orange), while in the duplication it is the last unit (orange/red).

to determine the identity of this unit, and what will happen to the ArrayList, is

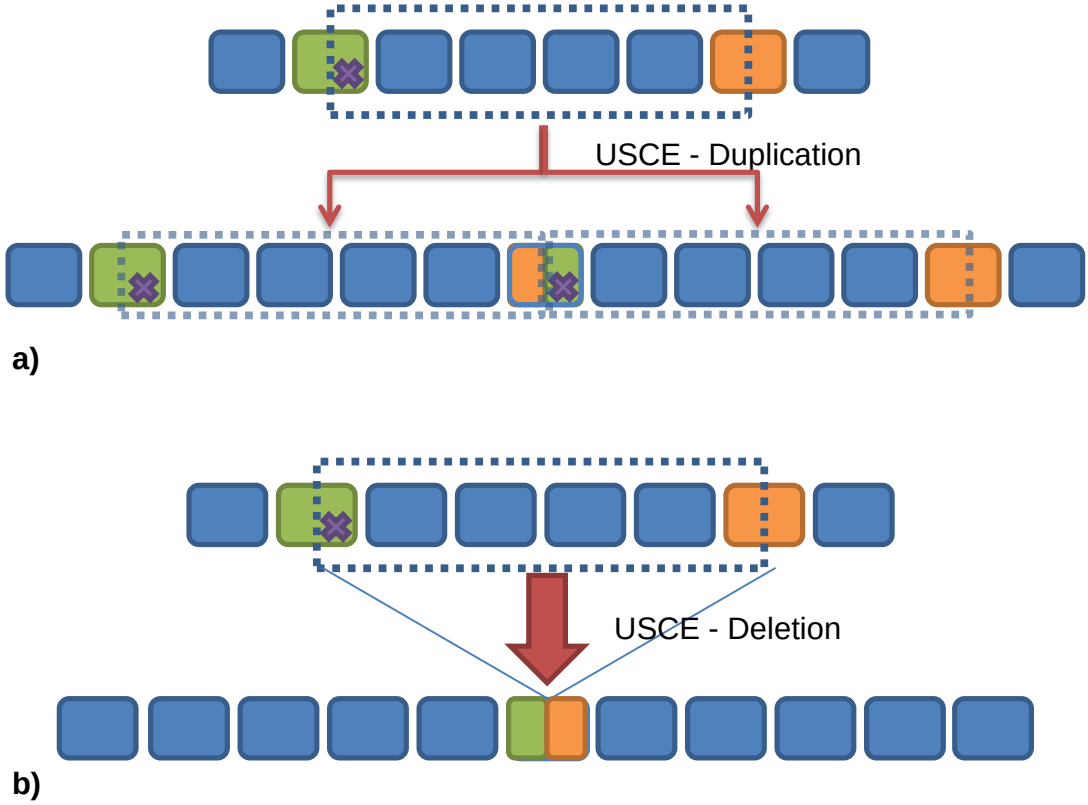

Figure 4.4.: Representation of USCE events in an rDNA array (representing fate of one chromatid), with a pSNP shown as a purple cross. a) a duplication event involving a tract of 5 units, resulting in the spread of a pSNP. b) deletion event involving a tract of 5 units, in this case resulting in the loss of the pSNP

represented as a flowchart in figure 4.5.

This flowchart forms the basis of a series of nested **IF** statements which will then duplicate or delete units which are part of the misalignment tract, and alter the identity of the composite unit depending upon the path through the flowchart. In many cases the identity of the composite unit will not need to be changed as the units involved are identical. It is only if they are not identical that the position of the break in relation to the pSNP is important. Also of note, depending on whether the event is a duplication or a deletion, the unit which is composite changes. In the case of a duplication, the last unit involved in a tract is the composite (the orange unit in figure 4.4a), because the copied units are inserted at a point within the unit. In the case of a deletion, the first unit involved in the break changes (the green unit in figure 4.4b), as a set number of units after this one are removed from the array. These composite units are dependant upon the break position within the unit, and the identities of the first and last units in the

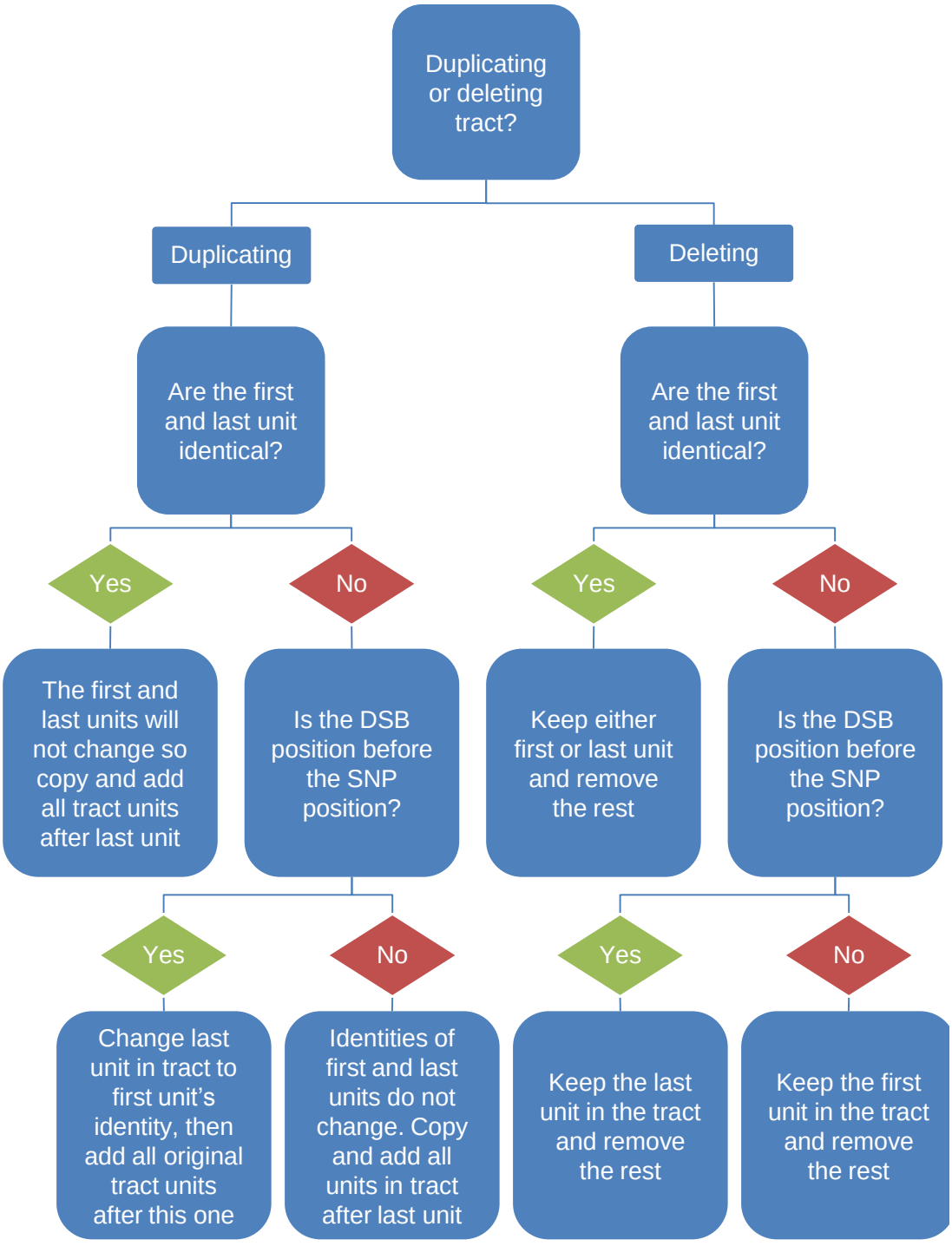

Figure 4.5.: Overview of the USCE method

tract (green and orange in figure 4.4). The different possible outcomes and their consequent unit changes are illustrated in figure 4.5, and have been implemented accordingly in the USCE code.

Another complication with the USCE method is alluded to in figure 4.2, where the number of units within the array needs to be maintained between reasonable (experimentally determined) values. This is important as the number of units in an array varies between species, but is maintained around a certain number within a species (Ide et al., 2010), as discussed in Chapter 1. Within SIMPLEX there are a series of **IF** statements that check whether the USCE event chosen will break the boundary conditions set for the size of the array, before the USCE method is called. These are illustrated in the flow chart in figure 4.6. If the event would cause the array to exceed these bounds, the USCE event is changed to the opposite, so to a duplication event if a deletion would result in the array being too small, and vice versa.

The considerations and layout of the methods used for Gene Conversion are different, as smaller tracts are used, and the array size will not change.

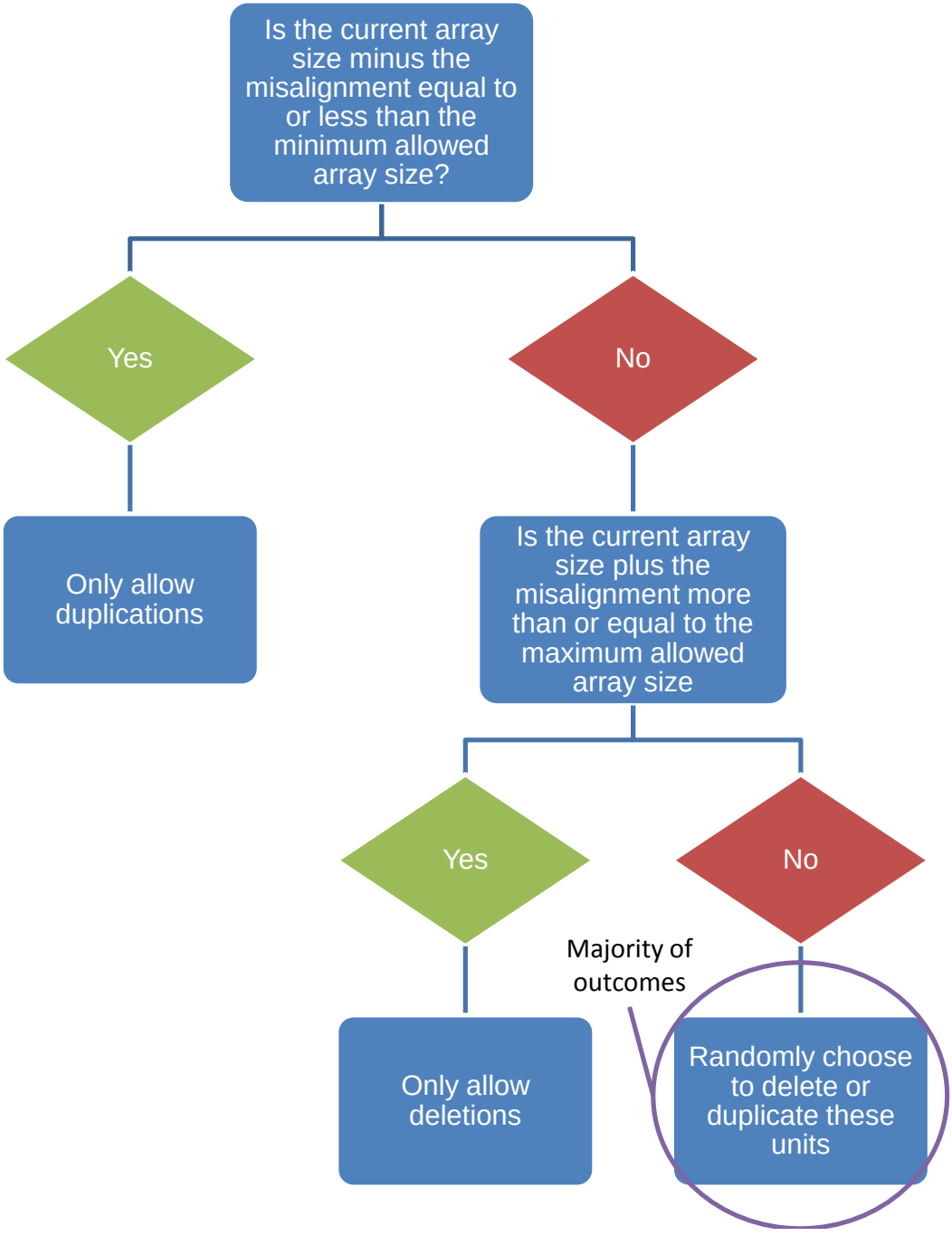

Figure 4.6.: How to deal with boundary conditions for the size of the rDNA array in USCE

Gene Conversion Method

Although the method to simulate a gene conversion event involves a smaller number of units, in some ways it is more complicated than a USCE event as a tract can span two units, both of which could both change identity. For a USCE event, where whole units were added or removed, only one unit could potentially change in composition.

As discussed at the beginning of this chapter, and in Chapter 1, gene conversion involves overwriting a section of an acceptor unit with a section from a donor unit, as illustrated in figure 4.7. This is sometimes described as a copy-paste event. As in figure 4.7, this sequence could span two units, or be contained within one unit.

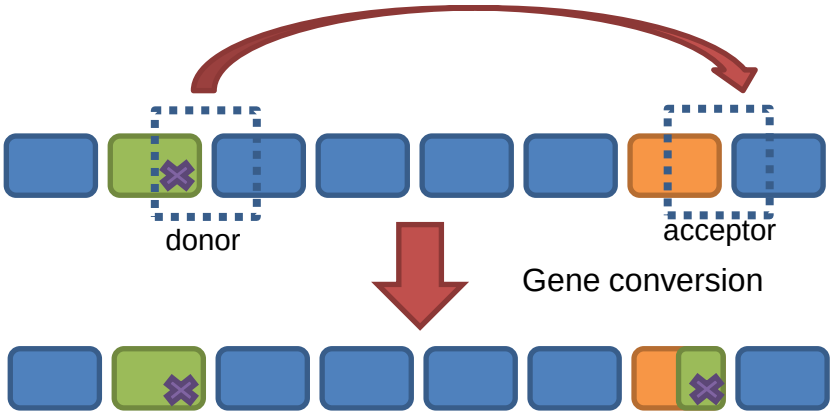

Figure 4.7.: Overview of a Gene Conversion event in an rDNA array (representing the fate of one chromatid). The X represents a pSNP within a unit. In this case, the pSNP frequency increases by one.

This method uses many of the same variables as USCE:

- The donor unit position, chosen at random
- The misalignment, how many units away the acceptor and donor are.
- The acceptor unit, which may be a misalignment distance away from the donor in either direction
- Unit size
- Double- stranded break position, this in conjunction with unit size and tract size will determine if the tract spans 2 units
- pSNP position

It also includes a number of other parameters:

- The size of the conversion tract (static in this version)
- Overflow tract, this is how far the conversion tract will go into the next unit (equal to the tract size plus the break position, minus the unit size)

If the overflow tract is >0, then there will also be acceptor+1 and donor+1 units to consider and compare (blue units within the dotted boxes in figure 4.7).

Similarly to the USCE method, a series of nested **IF** statements are used to compare donor and acceptor units, but in this case there will be more of them due to the possibility of tracts spanning units. There are too many options here to consider solely diagrammatically, so this is best visualised as a flowchart, as seen in figure 4.8.

Figure 4.8.: Overview of the twelve different outcomes for units in the rDNA array during the GC method. In the green boxes, D refers to the donor unit, A to the acceptor unit, D+1 refers to the donor + 1 unit, and A+1 refers to the unit after the acceptor unit.

Testing the Gene Conversion Method

As in the USCE method, there are a number of scenarios (in this case six) where a GC event results in no change to the rDNA array. But as there are more comparisons between unit identities in the gene conversion method, a number of different parameters to test each of the outcomes was formulated, to ensure the method was producing the correct results. Ten cases were devised, with all twelve possible outcomes from figure 4.8 implicitly tested as similar methods were used for different outcomes. In these tests, pSNP positions were generated before and after breaks, and in different units. Finally the results were compared to the expected outcomes. The following runs were undertaken:

1. no overflow, donor and acceptor different, donor with pSNP, differences not within tract.
2. no overflow, donor and acceptor different, donor with pSNP, differences within tract.
3. no overflow, donor and acceptor different, acceptor with pSNP, differences not within tract.
4. no overflow, donor and acceptor different, acceptor with pSNP, differences within tract.
5. overflow, both sets different, only donor + 1 and acceptor + 1 have pSNP differences within tract.
6. overflow, both sets different, both sets have differences within the tract.
7. overflow, both sets different, neither set has differences with the tract.
8. overflow, only donor + 1 and acceptor + 1 different, differences within tract.
9. overflow, only donor + 1 and acceptor + 1 different, not within tract.
10. overflow, none different.

Each test case gave the expected results, with the table of results for these runs shown in table 4.2

Tract size	**Start position**	**Over-flow**	**Unit i.d**				**I.d after GC**	
			Donor	D+1	Acceptor	A+1	Acceptor	A+1
4000	4000	0	3000	n/a	-1	n/a	-1	n/a
4000	2000	0	3000	n/a	-1	n/a	3000	n/a
4000	4000	0	-1	n/a	2000	n/a	2000	n/a
4000	2000	0	-1	n/a	3000	n/a	-1	n/a
4000	8000	3000	2000	2000	-1	-1	-1	2000
11000	2000	4000	3000	-1	-1	3000	3000	-1
3000	8000	2000	3000	3000	-1	-1	-1	-1
5000	8000	4000	-1	3000	-1	-1	n/a	3000
4000	6000	1000	-1	3000	-1	-1	n/a	-1
4000	8000	3000	-1	-1	-1	-1	-1	-1

Table 4.2.: Results from testing the GC method with known values of different units. Unit size was set to 9000 for simplicity. I.d.'s of -1 refer to consensus units possessing no pSNPs. All results returned were as expected.

Program Output

The final aspect of SIMPLEX to be discussed is the output of the program. The results of each simulation run are saved into a new directory named "simulation" + the date and time the run started. Each individual simulation is output into a tab delimited text file, named with its position in the run (i.e simulation 1), suffixed with "_lost" or "_fixed" depending on whether the pSNP was lost or fixed within the array during the simulation (for example, the filename "2010-10-12:15-55-27_simulation 6_lost", denotes that a pSNP was lost in run 6 (of 10,000), started at 15:55:27 on the 12th of October 2010). The output file includes:

- A header with the parameter details i.e. starting array size, percentage that should be USCE events, date and time, max and min array size permitted, the starting unit containing the pSNP, and the position of the pSNP within the unit.
- A line for each iteration (concerted evolutionary event), with the current array size, the number of units containing pSNPs, and the identity of the event (GC or USCE)
- A final line saying whether the pSNP was fixed or lost, and the number of iterations carried out in the run.

The output files can be input into spreadsheet software such as Microsoft Excel, allowing the way in which the number of pSNPs or array size varies over the

course of a simulation to be visualised easily.

Furthermore, a summary file is also produced for each run, which includes a header of all of the parameters set for that run, and the following information summarised for each simulation:

- The simulation number to identify the run
- The numbers of each type of event carried out in the run
- The maximum and minimum number of units reached
- The maximum number of pSNPs reached
- The maximum percentage occupancy of pSNPs reached
- Whether the pSNP was fixed or not
- The size of the array at the end of the simulation
- The total number of iterations reached

4.3. Preliminary SIMPLEX Experiments

A series of experiments was designed to evaluate the utility of the SIMPLEX program in shedding light on the concerted evolutionary process. Simulations of 10,000 runs were undertaken for each set of parameters, and the results compared. In Experiment 1, three different sets of event parameters were used. In the first set, runs only performed USCE events, in the second only GC events were undertaken, and in the third a ratio of 80% USCE and 20% GC events were run. The latter parameter set was chosen as this was the estimated balance between crossover and non-crossover events in previous research, although not on rDNA (Nickoloff et al., 1999). In the GC method, the tract size was constant at 4000 bases, the lower estimate of tract size (Judd and Petes, 1988). The minimum array size and maximum array size were set to be 70 and 200 respectively, the approximate range of array sizes estimated in the SGRP dataset (James et al., 2009). The size of the unit was set to be 9138bp, the size of a unit shown in the *S. cerevisiae* reference strain at the SGD. The pSNP position within the unit was set to be 4000, and the unit starting with a pSNP was set to be unit 70, both chosen as they are near the mid point of a unit, and the array respectively.

The proportion of simulations runs in which the initial pSNP was fixed or lost can be compared, as can the time (or number of iterations) that each run takes to fix or lose a pSNP and changes in the array size.

In Experiment 2 the effect of changing the pSNP occupancy at the start of a simulation was examined. In Experiment 3 the effect of the position of the initial pSNP in a unit and the unit within the array was examined. The parameters used in Experiment 1-3 are shown in table 4.3. In the following experiments the positions of pSNPs and units are zero-indexed, so the first base position within a unit is referred to as position 0, and the first unit within an array as unit 0.

Parameter	**Experiment 1**	**Experiment 2**	**Experiment 3**
Starting Array Size	140	140	140
Minimum Array Size	70 for most runs, 20 for end array size experiment	70	70
Maximum Array Size	200 for most runs, 270 for end array size experiment	200	200
USCE/GC Event ratio	100% USCE, 100% GC and 20%:80% USCE:GC	20% USCE:80% GC	20% USCE:80% GC
pSNP position within a unit	4000	4000	0, 1, 10, 50, 250, 1000 ,4000, 8000, 9080, 9120 ,9126, and 9127
Number of Units Starting with a pSNP	1	1, 14, 28, 42, 56, 70, 84, 98, 112, 126 and 139 units	1
Unit starting with a pSNP	Unit 70	Varies with number of units.	Unit 0, 1, 10, 80, 100, 138 and 139

Table 4.3.: Parameters used in SIMPLEX for the three sets of experiments. Unit 0 is the first unit in an rDNA array, position 0 is the first position.

4.3.1. Test Runs and Visualisation of Results

In test runs of 10,000 simulations at starting array size of 140, 20% USCE events, max array size of 200, minimum array size of 70, and misalignment of 10, a run took approximately 40 seconds, the output directory was 76 MB in size, and SNPs were fixed in ∼70 out of 10,000 runs.

A single simulation run can be visualised as a line chart illustrating how the percentage of units which contain a pSNP varies as the run progresses. An example is shown in figure 4.9. In this example, an initial pSNP is fixed within the array (as the percentage of units with a pSNP reaches 100% and the simulation

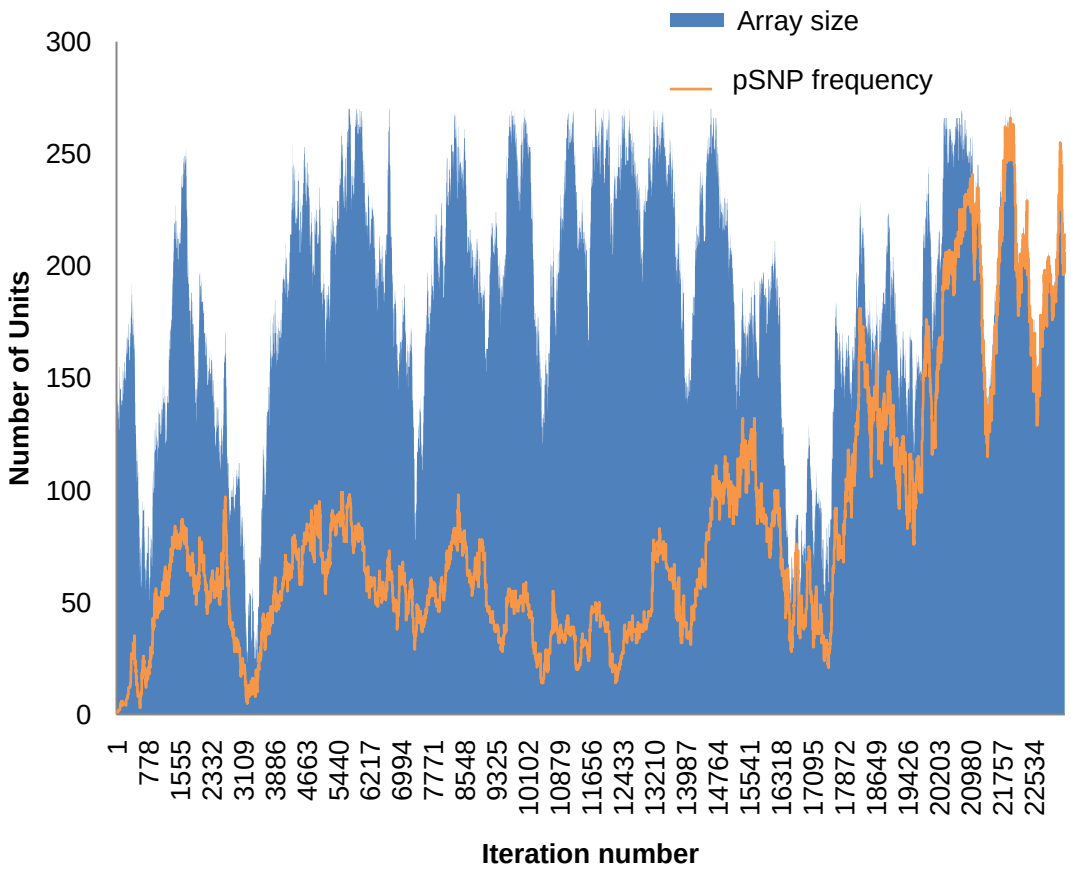

Figure 4.9.: Line chart example of pSNP frequency changing over the course of a single run. In this run the initial pSNP is fixed within the array after ∼23,000 events

run ends), after approximately 23,000 events. However, note that fixation was almost reached at various points along the run, particularly after ∼17,000 events. As expected with only one unit out of 140 containing a pSNP, the majority of runs result in the pSNP being lost from the array, with only 50-70 simulation runs out of 10,000 resulting in fixation for this parameter set.

Also, as expected, on average it takes considerably longer to fix a pSNP than to lose it, as illustrated in the cumulative frequency chart in figure 4.10. Taking the case where 20% of events are USCE, in those runs in which the pSNP was lost, approximately 90% had lost the pSNP within 1,500 concerted evolutionary events. However, in runs in which the pSNP was fixed, only 50% had completed after 6,500 events. This result is expected as only one unit out of 140 contains a pSNP, and therefore changing just this one unit will lead to loss of the pSNP, as opposed to fixation which will require all units to have been affected by one or more concerted evolutionary event.

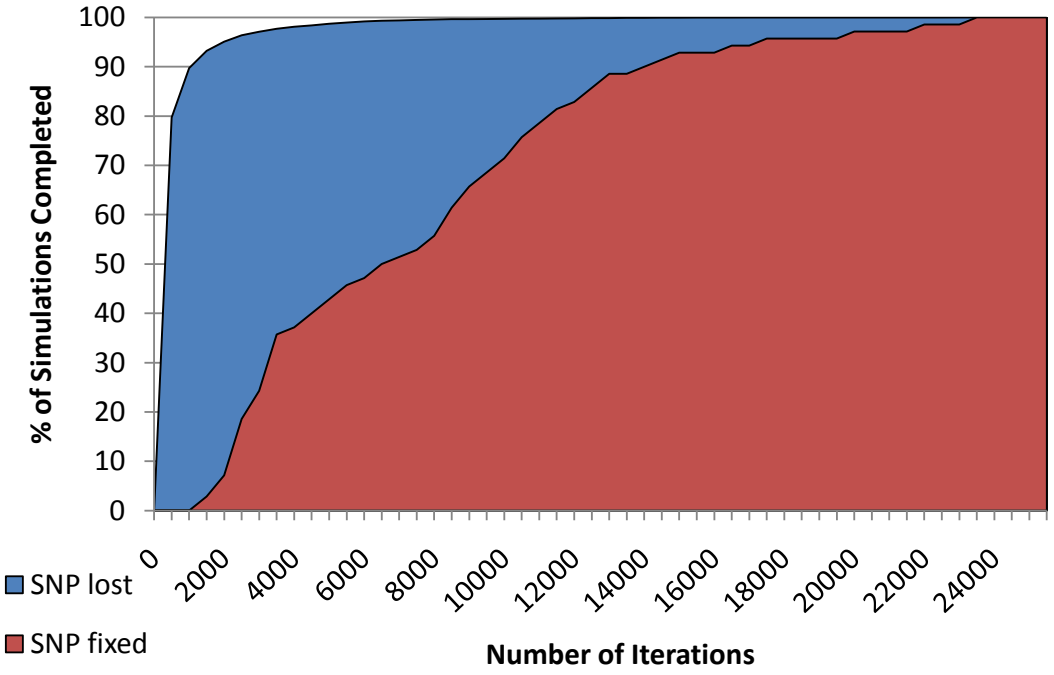

Figure 4.10.: Chart of cumulative frequency of percentage of total simulations completed within a number of iterations. This is for 20% of events being USCE, and starting with one unit containing a pSNP. Results are shown as a percentage of runs finished by number of iterations.

4.3.2. Experiment 1: Varying the ratios of USCE to GC events

Simulation of 10,000 runs were carried out for three different proportions of the two concerted evolutionary events, with the parameters shown in table 4.3. Comparing the effects that the different concerted evolutionary events have on the fixation and loss times, it can be seen that USCE events have a disproportionate effect on the rate of pSNP fixation, as shown in figure 4.11. Fixation takes more events to achieve than loss with all three event ratios examined: 100% GC, 100% USCE, and 20%USCE/80% GC. The number of events until fixation or loss is reached is smallest when only USCE events occur, and largest with 100% gene conversion. However, on average gene conversion takes 31 times more events to fix a pSNP, and 14 times more events to lose a pSNP, than USCE alone. This is to be expected to a certain degree, as gene conversion has a tract size of 4000 bases, whereas USCE can involve between 9,137 and 91,370 bases (2 to 22 times longer tract sizes than GC). Furthermore, USCE can alter the array size, influencing pSNP occupancy across the array. When only 20% of events are USCE, the number of events needed until fixation or loss is achieved is still drastically reduced in comparison to the 100% GC runs. Approximately 10 times fewer events are needed

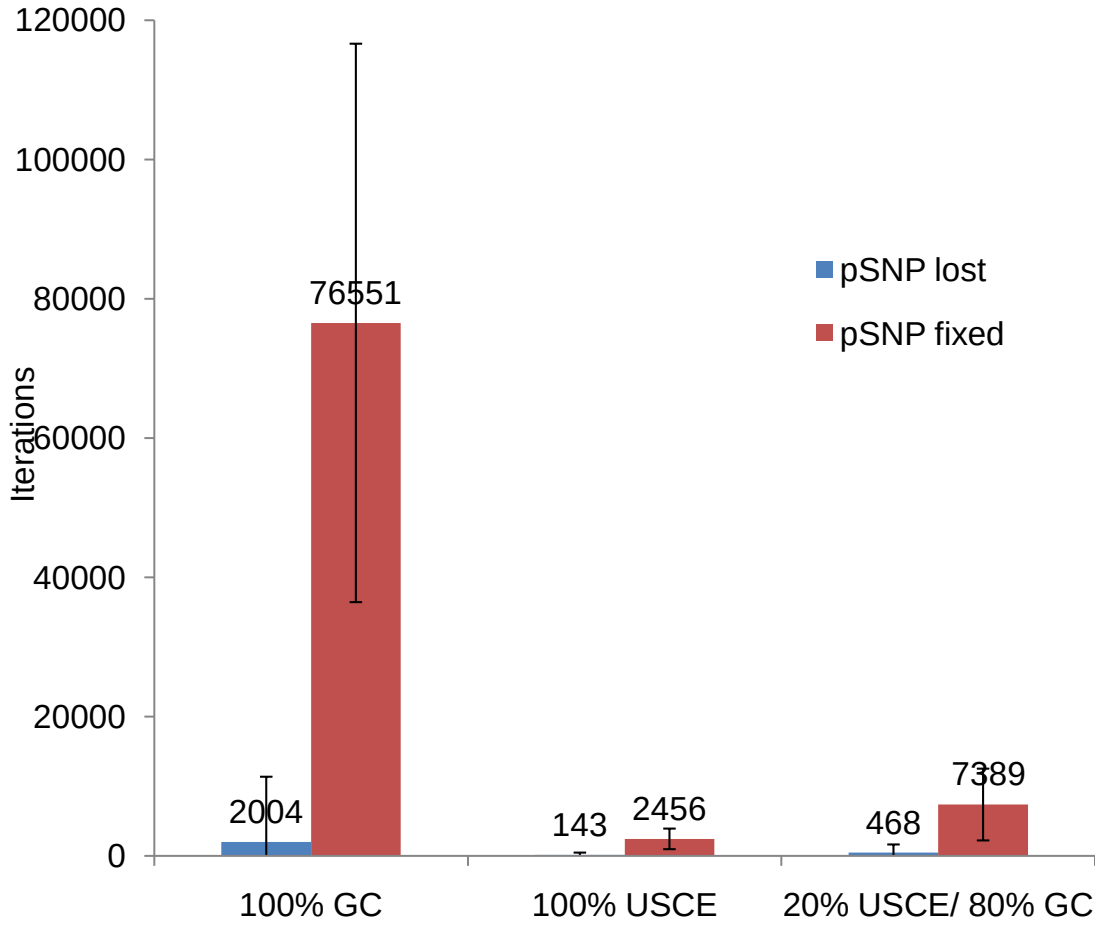

Figure 4.11.: Bar chart representing the mean number of iterations (from simulations of 10,000 runs) until a single initial pSNP is fixed or lost from an array, comparing three different ratios of the two event types, USCE or GC. USCE events greatly reduce the total number of events needed until fixation/loss compared to GC events. Error bars show standard deviation across the 10,000 runs.

for fixation, and 4 times fewer for loss. In all cases but 100% USCE there is a large variation in the number of events needed, illustrated in figure 4.11 by the large error bars. Event number variation is particularly large for the 100% gene conversion case.

The number of units in the rDNA array at the end of the simulation runs are very different between those in which pSNPs are lost or fixed, but show little difference between event ratios, as shown in figure 4.12. For this experiment, the minimum array size was set to be 20, and the maximum to be 270 units, to allow a larger variation. The similarity between event ratios is likely due to USCE being the only event which changes the array size, and because despite there being more total events in the 20% USCE simulations, the number of USCE events within the

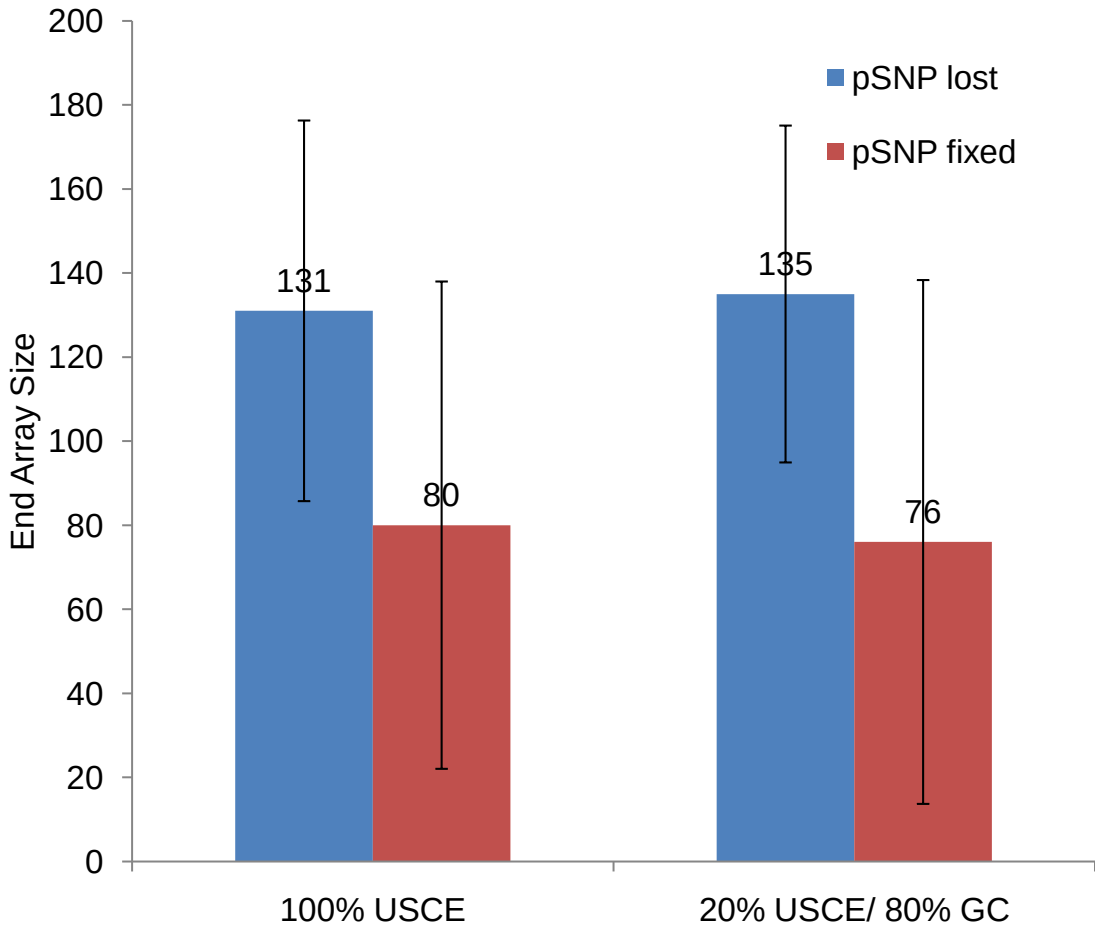

Figure 4.12.: Bar charts comparing the average end array size when simulation runs have completed. 100% GC not shown as this will not alter from the initial array size. Error bars show standard deviation.

100% USCE and 20% USCE/ 80% GC simulation runs are very similar. However, in both cases rDNA array sizes are on average much smaller when a pSNP is fixed than when it is lost. The average number of units when a pSNP is lost is very close to the starting array size of 140, whereas the array size is closer to the minimum when the pSNP is fixed. This could be accounted for by far fewer events being needed to lose a pSNP than to fix one, as far fewer units need to change state, shown previously in figure 4.10, where a large number of pSNPs are lost in a relatively small number of events. This would mean that many times there would be few events until pSNP loss, and the array size will be close to the starting size in many cases. Looking at the results for the spread of the array size data, shown in table 4.4, the maximum and minimum end array sizes are similar for each case. However, in those cases where a pSNP is fixed, it could be more likely to occur at a smaller array size as there are fewer units which need to gain the pSNP.

	100% USCE fixed	100% USCE lost	20%USCE/ 80%GC fixed	20%USCE/ 80%GC lost
minimum	22	21	21	21
maximum	246	265	253	270
median	55	131	46.5	135
average	80	131	76	135
standard deviation	58	45	62	40

Table 4.4.: End array size for 100% USCE and both event types (USCE and GC), comparing when pSNPs are fixed or lost from the array.

4.3.3. Experiment 2: Changing Proportions of Units Containing a pSNP at the Start of a Simulation

The effect of changing the number of units which start with a pSNP was investigated. pSNPs, at position 4000, were included in a series of different units, equating to approximately 1%, 10, 20, 30, 40, 50, 60, 70 ,80, 90 and 99% of the units, see table 4.3. Each case was simulated for 10,000 runs.

The number of simulation runs to either fix or lose pSNPs related linearly to the number of units which started with a pSNP, with a correlation coefficient of -0.99 and 0.99 for loss and fixation respectively, as shown in figure 4.13. The symmetrical pattern in figure 4.13 further indicates the program is working correctly, as containing or not containing a pSNP could be seen as two different alleles in a population, and at 50% pSNP occupancy an approximately equal number of simulation runs should result in fixation and loss.

As the starting percentage of array units possessing a pSNP varies across the simulation runs, the number of events until fixation or loss occurs also varies, as shown in figure 4.14. Although there is a large degree of variation in event number between individual simulation runs in each percentage bin (not shown in figure 4.14), the average number of events over the 10,000 simulations shows a distinctive pattern. The pattern is again symmetrical but in this case shows a polynomial relationship between pSNP occupancy and number of events (second order with an R-squared value of approximately 0.98 in both fixation and loss).

The distribution of the number of events observed for fixation and loss at different starting percentages is shown in figure 4.15, for starting occupancies of 50% and below. The data appear to follow Poisson distributions, a natural distribution for independent counts occurring at an identical rate. At 50% occupancy the

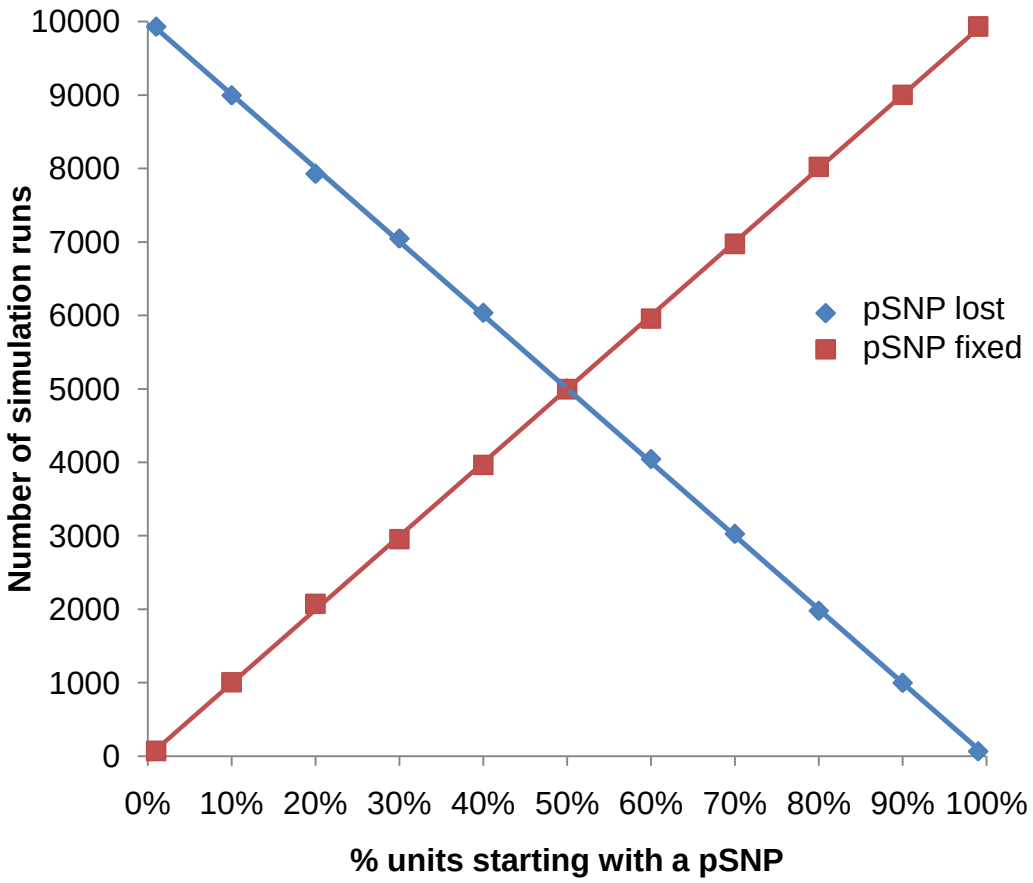

Figure 4.13.: Proportion of 10,000 simulation runs in which pSNPs were fixed or lost, when the percentage of units which start with a pSNP is varied

distribution is the same for those simulation runs in which the pSNPs are fixed and lost. At all starting occupancies, similar numbers of runs are fixed and lost after approximately 9000 events. Runs which complete after 9000 events contribute more towards the value of the average number of events until pSNP fixation when starting occupancies of pSNPs are low, (and similarly average number of events until loss when starting occupancies are high) as they account for a greater proportion of the total number of runs, as seen as the shallower distributions in figure 4.15.

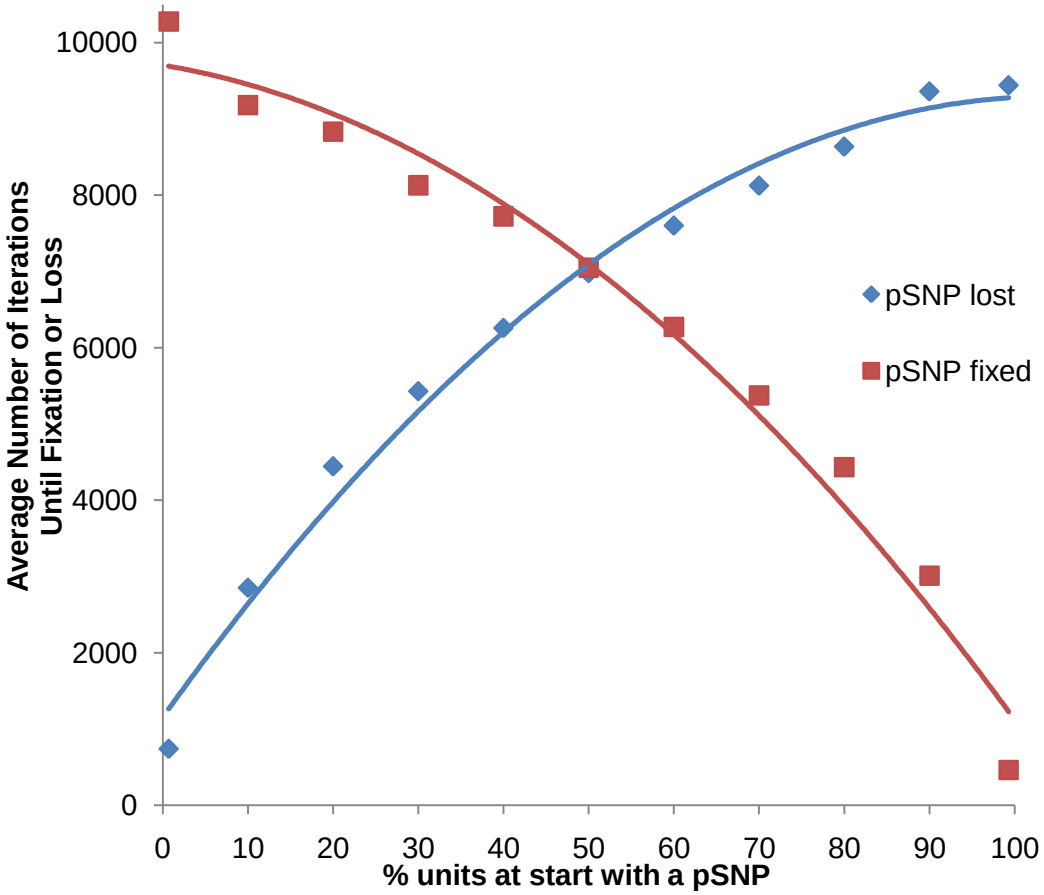

Figure 4.14.: The average number of events taken to fix or lose a pSNP, when the initial pSNP occupancy varies.

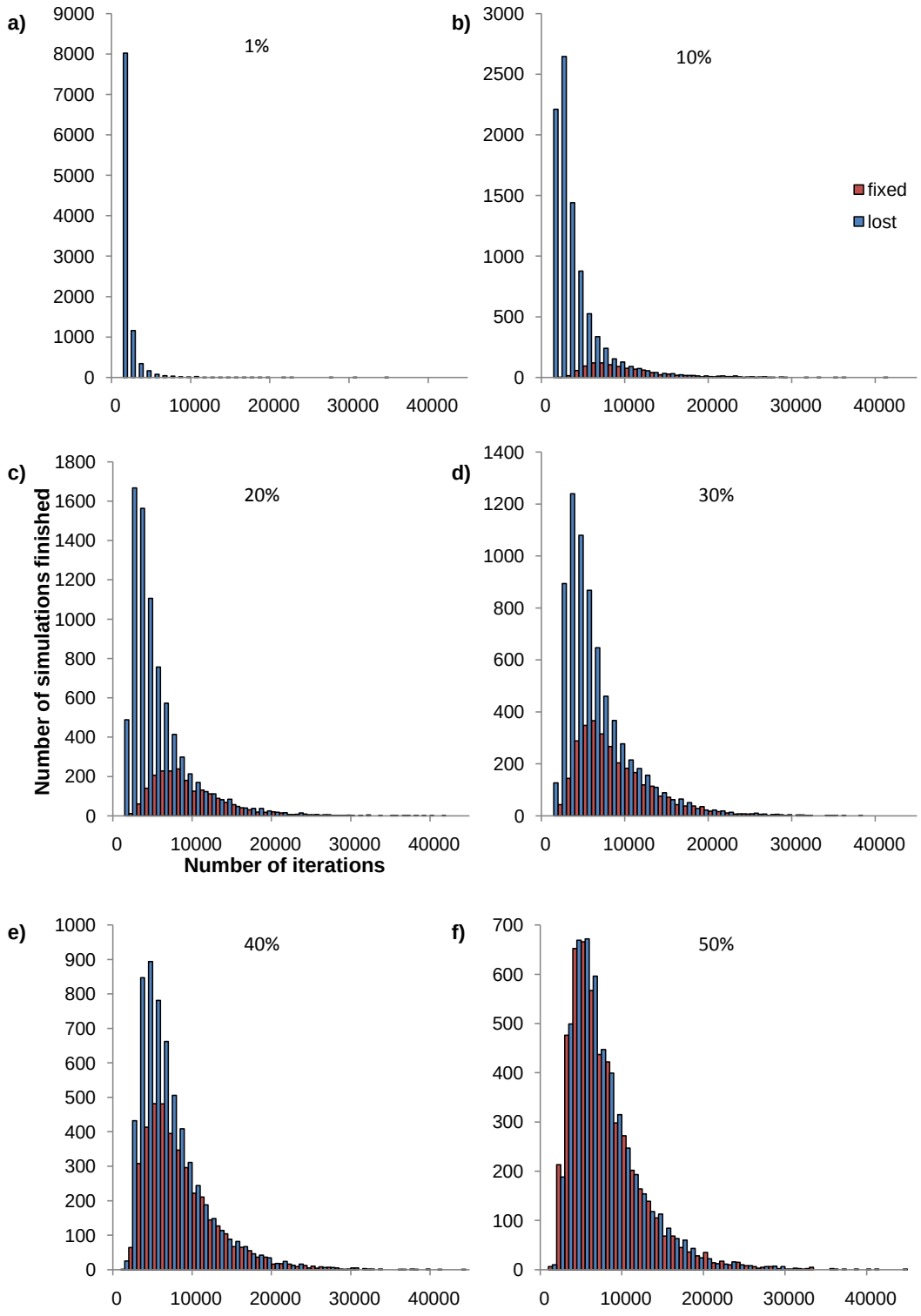

Figure 4.15.: Histograms of the number of events taken to fix or lose a pSNP, when the initial pSNP occupancy varies. Initial occupancies are shown at the top of each histogram, with each bin showing an interval of 1000 events

4.3.4. Experiment 3: pSNP Position Within the Array

This experiment aims to assess the effects of pSNP location within a unit, and unit position within an array, on the number of events to fixation or loss. A series of simulations were undertaken using 20% USCE and 80% GC with 10,000 runs for each simulation. The pSNP position within a unit, and the position of this unit within the rDNA array were varied. The positions are zero-indexed, so the first base position within a unit is referred to as position 0, and the first unit within an array as unit 0. Within the 140 units in the starting array, units 0, 1, 10, 80, 100, 138 and 139 were each tested. pSNPs were tested in each of those units, at base positions 0, 1, 10, 50, 250, 1000, 4000, 8000, 9080, 9120, 9126 and 9127. The results of the 84 simulations were split according to whether the pSNP was lost or fixed.

The results of runs in which pSNPs were lost are shown in table 4.5 and figure 4.16. In all units tested except unit 0, the position of the pSNP within the unit does not greatly effect the average number of concerted evolutionary events taken to lose a pSNP in the majority of positions. This is illustrated in figure 4.16 where the number of iterations is fairly flat for all but the first positions in these units, being under 1000 events in almost all cases. When pSNPs were at position 0 or 1 modest increases in the average number of events were seen, in particular in units 1 and 100. There was also a slight elevation in the number of events in the last unit in the array (139), for pSNP positions over 8000. However, when looking at unit 0 (the first unit within the array), pSNP positions under 1000 bases show considerably elevated average numbers of events, with values for pSNP positions under base 4000 not able to fit within the same axis of the bar chart in the figure (the full y-axis is shown in the top right of figure 4.16). It might be expected that the ends of the array are 'mirrored', with both ends proving more difficult to "access" via mutation, however the first unit seems to be more resistant to change.

However, these are very different results to those found when investigating the average number of events for different units and pSNP positions in those simulation runs where pSNPs are fixed, as shown in table 4.6 and figure 4.17. As in previous results, the number of events required to fix a pSNP is much greater than to lose it. Furthermore, for all pSNP positions above 1000 at most unit positions the number of events is relatively constant. The only exception to this is in the case of unit 139, where base position 9080 has a slightly higher average, and pSNPs

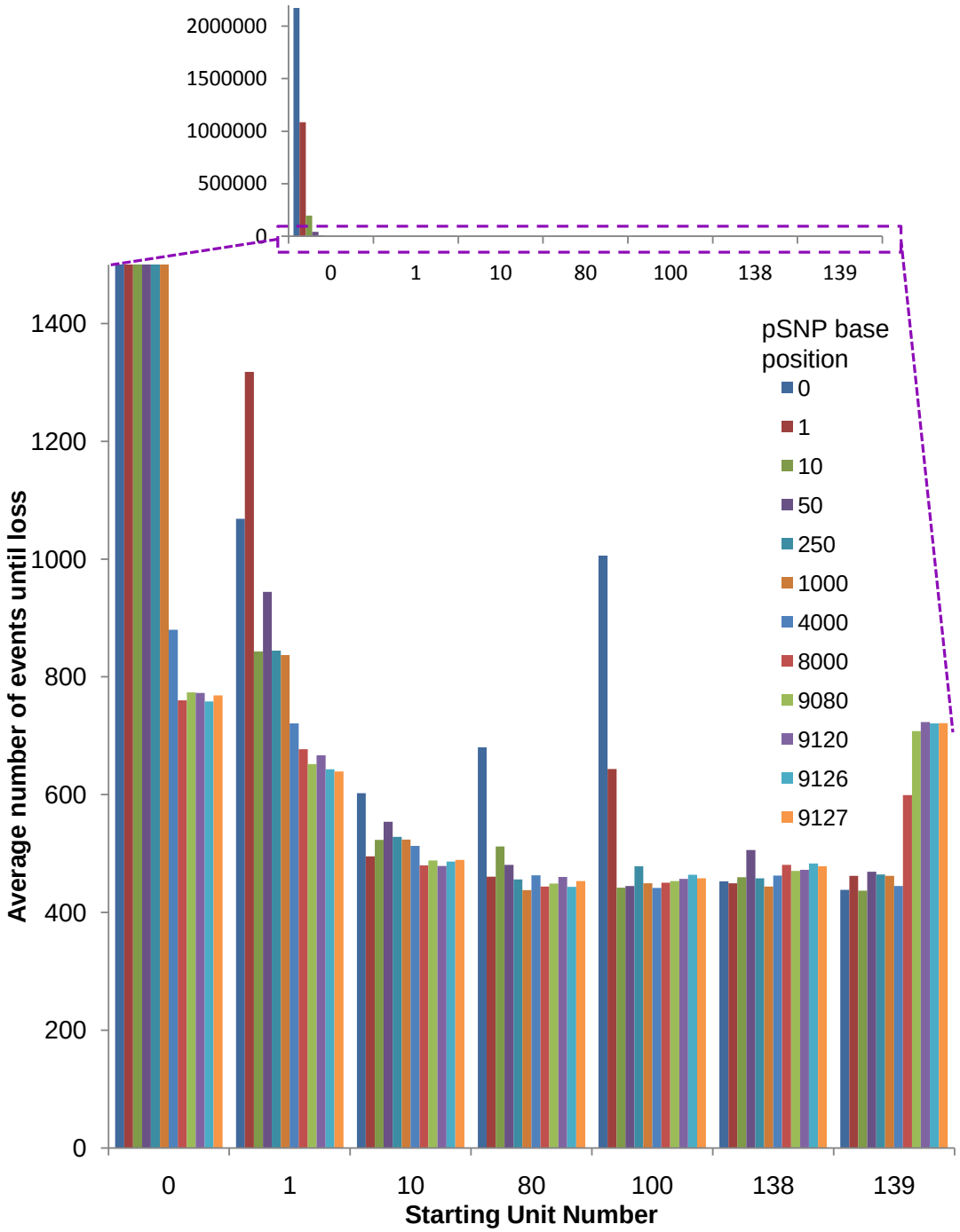

Figure 4.16.: Bar chart showing the average number of iterations in SIMPLEX until a pSNP is lost, varying the starting unit containing a pSNP, and the position of the pSNP within the unit. Top right shows the bar chart with a full y-axis, the main chart showing the same dataset but with a truncated y-axis

pSNP	unit						
position	0	1	10	80	100	138	139
0	2171821	1068	603	680	1006	453	439
1	1085915	1318	495	461	644	450	462
10	197451	843	523	512	442	460	437
50	42751	944	554	481	445	506	469
250	8728	844	528	456	478	458	464
1000	2437	837	524	438	449	444	462
4000	880	721	513	463	441	463	445
8000	760	677	480	444	450	481	599
9080	773	652	488	449	453	471	708
9120	773	667	479	460	457	472	723
9126	758	643	487	444	464	483	721
9127	768	639	489	453	458	479	721

Table 4.5.: Average number of iterations until the pSNP is lost for different starting units and pSNP positions

at base positions 9120 and above are never fixed. The absence of terminal base positions fixing in this unit could be linked to the increased number of iterations needed to lose a pSNP in the last unit, seen in figure 4.16. However, for all unit positions tested, as pSNP positions decrease from 1000 downwards increasing numbers of iterations are required to fix the pSNP, unlike the cases for loss where only position 0 was affected. This is illustrated in figure 4.17 where there are a very large number of iterations at the start of each unit, and also by looking across the rows of table 4.6, where the number of events at each pSNP position is similar across all units tested.

This difference in pattern, where a large number of events is required to lose a pSNP in the first unit only whereas large numbers of events are required in all units to fix a pSNP at the start of a unit, can be explained at different levels. Here rDNA units are set to be 9127 bases long, and there are 140 units. Consequently there is only one chance to choose the first base in the first unit by either method, as tracts go in the downstream direction. This translate to a chance of one in 1,277,780 events on average, the same magnitude as the average number of iterations to fixation and loss in the first base position (table 4.6). This also relates to the magnitude of the decrease in the average number of iterations for subsequent positions in the first unit. For example pSNP position 50 in unit 0 has approximately 50 times fewer iterations on average needed for fixation than

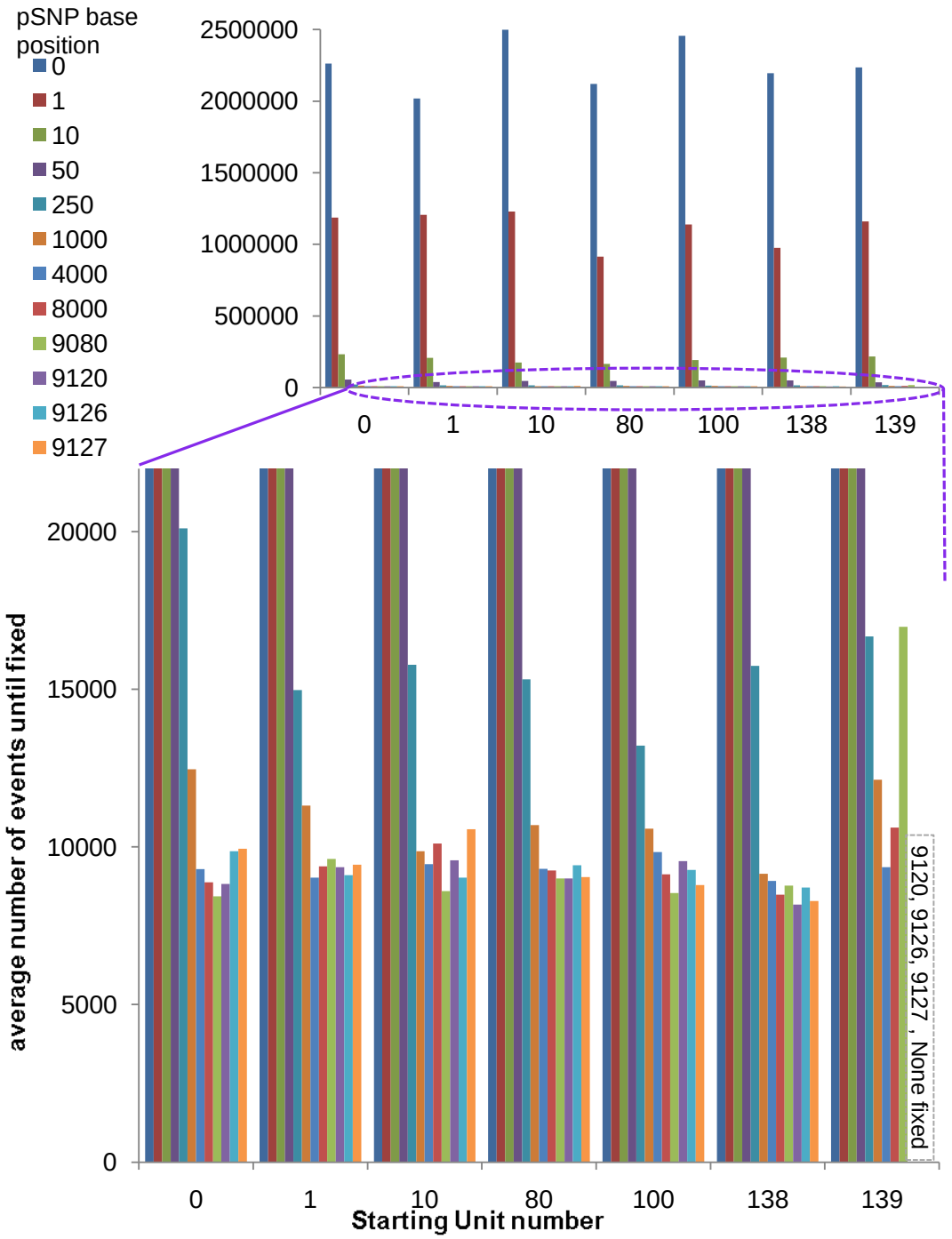

Figure 4.17.: Bar chart showing the average number of iterations of SIMPLEX until a pSNP is fixed, varying the starting unit containing a pSNP, and the position of the pSNP within the unit. Top right shows the bar chart with a full y-axis, the main chart showing the same dataset but with a truncated y-axis

pSNP	unit						
position	0	1	10	80	100	138	139
0	2262225	2019050	2498029	2121235	2455907	2195744	2236025
1	1187652	1206765	1228112	913174	1139146	976316	1159920
10	233028	206603	174881	164307	191106	208808	216708
50	55798	38325	46607	46321	49245	48668	36244
250	20100	14975	15777	15318	13215	15739	16676
1000	12459	11313	9859	10693	10575	9151	12130
4000	9296	9027	9455	9300	9840	8918	9354
8000	8878	9381	10105	9248	9133	8487	10614
9080	8436	9615	8595	9001	8533	8774	16983
9120	8821	9357	9579	9000	9547	8168	0
9126	9864	9105	9023	9416	9267	8709	0
9127	9943	9436	10558	9045	8790	8283	0

Table 4.6.: Average number of iterations until the pSNP is fixed for different starting units and pSNP positions

position 1.

SIMPLEX assumes that concerted evolutionary events do not go beyond the boundaries of the array so that the array ends are left tidy as whole rDNA units. But is this biologically realistic? Well, just as the array cannot go out of bounds in this simulation, an rDNA unit would presumably be unlikely to pair with a region outside the array. In the simple representation of misaligning chromatids in figure 4.1, the first unit (blue in the figure) will not be able to pair with anything "above" it. The other end of the array will be similar. When the ends of the array do undergo any concerted evolutionary processes, if the sequence which flanks the rDNA is changed, this would result in degradation of the rDNA sequence at the end of the array, as rDNA would become interspersed with sequence from the flanking regions. Although there are some indications of partial terminal rDNA units in nature, for example in *S. cerevisiae* strain S288c where the rightmost rDNA unit is believed to possess a variant 5S region which is truncated, along with a partial IGS region (McMahon et al., 1984; Hillier et al., 1997), recent genome sequencing projects have yet to confirm this finding.

In future the flanking sequence of rDNA should be examined to increase knowledge of concerted evolution in these regions. This could result in different approaches for modelling the boundaries between the ends of the array and the flanks. Such a change was not undertaken here, but with some modifications, SIMPLEX could

be used to investigate evolutionary processes at the ends of the rDNA array in more detail.

4.4. Chapter Summary

A computer program, SIMPLEX, was written to computationally model the evolution of a single pSNP in a single rDNA array. SIMPLEX was used to generate initial results on the behaviour of pSNP fixation and loss, according to various parameter sets.

USCE was found to be more influential than GC in homogenising a pSNP within an array. A linear relationship was found between the number of runs which fixed a pSNP, and the percentage of units which already contained one. However, a polynomial relationship was discovered for the average number of events until fixation or loss, and the percentage of units which already contained one. Furthermore, the data followed a Poisson distribution for the number of events taken until fixation and loss of a pSNP in simulation runs at different starting pSNP occupancies. Lastly, the effect of position of a pSNP within a unit, and the effect of the position of a unit containing a pSNP upon number of events until fixation and loss was investigated. It was found that large numbers of events were needed to fix a pSNP in the first 1000 bases of all units, but large numbers of events were needed to lose pSNP in first unit only.

Some departures from these results are to be expected with refinement of the simulation program, in particular when allowing different selection schemes, and altering how the ends and flanking regions of the array are dealt with. However, clear differences between the two simplified event types (USCE and GC) are still evident, and the balance between them is seen to have a strong effect on the evolutionary trajectory of the rDNA array.

It is currently difficult to place these fixation and loss times in a biological context without knowing the ratio between the two mechanisms, or the ratio of "visible" events to double-strand break repair events which do not affect the order or size of an array. However, a rough guideline could be inferred from a 2008 study on mitotic and meiotic instability in the rDNA array of *S.cerevisiae*, which estimated that 1.2×10^{-3} USCE recombination occurred events between sister chromatids in rDNA per cell generation (Casper et al., 2008).

5. Simulating rDNA Evolution Across Species using the CONCERTINA Software

Chapter Abstract

Although preliminary data obtained from SIMPLEX simulation runs yielded interesting results and potential insights into concerted evolutionary dynamics, the software was limited in terms of the questions it could be used to answer. Instead of following the fate of a single pSNP within an rDNA array, it would be beneficial to follow a continual process of pSNPs being introduced at a set mutation rate over many generations, examining the frequencies of pSNPs and SNPs which result. Furthermore, a tree-like process where rDNA arrays split after a certain number of generations could also give information into looking at "distances" between the respective species/strains in which they reside. Therefore a new simulation program was written to incorporate these new features, named CONCERTINA (CONCERTed evolution IN rDNA tandem Arrays). The development and testing of this program is discussed, as is its use in two experiments, tracking pSNP numbers and occupancies in a single rDNA array and in a set of ten diverging taxa respectively. Finally, the possibility of using and extending the software in future to learn more about concerted evolutionary processes in both real and simulated datasets is discussed.

5.1. The CONCERTINA Tool

CONCERTINA implements computational models that enable a progressive process of pSNP mutation and evolution. It also models a series of rDNA arrays diverging over time in a tree-like fashion. To implement these new features and their inherent additional complexities, CONCERTINA treats the rDNA array

as an object rather than just a data structure (an ArrayList in SIMPLEX). In object oriented programming, objects have attributes to describe themselves (for example, the size of an array could be an attribute), and methods which can change the values of these attributes. Treating the separate elements as different objects also increases the extensibility of the code, and allows further updates to be more simply achieved. Many copies of each object may exist, referred to as "instances" of the corresponding class. So, for example, there are many rDNAarray objects in a simulation, each of which is a different instance of the rDNAarray class. Each object has different values for the variables or attributes associated with it, for example different sizes or pSNPs, and there are methods within the rDNAarray class which can change these values. Like SIMPLEX, CONCERTINA is written in Java, an object-oriented programming language.

A number of different objects represent different levels of complexity within the concerted evolutionary process, as illustrated in figures 5.1 and 5.2. Each object will now be described, following these figures from the bottom up. Each rDNA unit is represented in CONCERTINA as an object, aUnit, each of which has a number of bases (i.e. sequence length), and a list of pSNP positions which are found in that unit. The class also contains a number of methods associated with the object, which include a number of constructors to create a new aUnit object, either with no pSNPs or with a pre-existing list of pSNP positions. A method also exists to add a pSNP to an aUnit object's ArrayList of polymorphisms.

The aUnit objects are contained within rDNAarray objects. rDNAarray objects possess an ArrayList containing many different aUnit objects. This is illustrated in figure 5.1, where each blue aUnit has a potentially unique set of pSNP positions represented as integers. rDNAarray objects also contain values for the current number of aUnit objects within theArray, and the minimum and maximum size that the array is allowed to reach. Methods include those to add or remove aUnit objects at given positions from an rDNAarray, and to access an aUnit in a given position within an rDNAarray to allow a pSNP to be added or removed. A method to print the details of the array to file, which includes the array size and the polymorphisms within the array and their occupancies is also included.

The next higher level object is a BinaryNode object. This object forms part of the uppermost data structure within CONCERTINA, which is a binary tree. Binary trees are themselves a hierarchical data structure, represented as the red and green nodes in a tree-like structure shown in figure 5.1. A binary tree contains nodes, each of which will itself contain between zero and two nodes, which can be

denoted as left and right child nodes. The binary tree has a root node, (shown as red in figure 5.1), which is the ancestor of all following nodes. Each node can be reached by following a path from this root node, with each step either going to a subsequent left or right child node. Binary trees are an appropriate choice of data structure for the rDNAarray objects due to their hierarchical nature, and ease in relating position within the tree to relationships between nodes. Traversal of a binary tree, in a number of different ways, is also a rapid process. In this case each BinaryNode contains a reference to its child BinaryNodes (left and right), but also contains an rDNAarray object. Furthermore, a BinaryNode contains a distance to its parent node, which is equivalent to the number of concerted evolutionary events separating the two. The final object is the BTree object, which only contains one BinaryNode object, the root of the tree. However, this class contains a number of methods to create the tree by inserting BinaryNodes, as well as a method to print the tree. To print the tree, as BinaryNode objects contain BinaryNode objects, a BinaryTree can be traversed in a recursive manner, where the method calls itself. In this case the tree is traversed in a pre-order manner, where the root is visited first, then the left subtree, and then the right subtree.

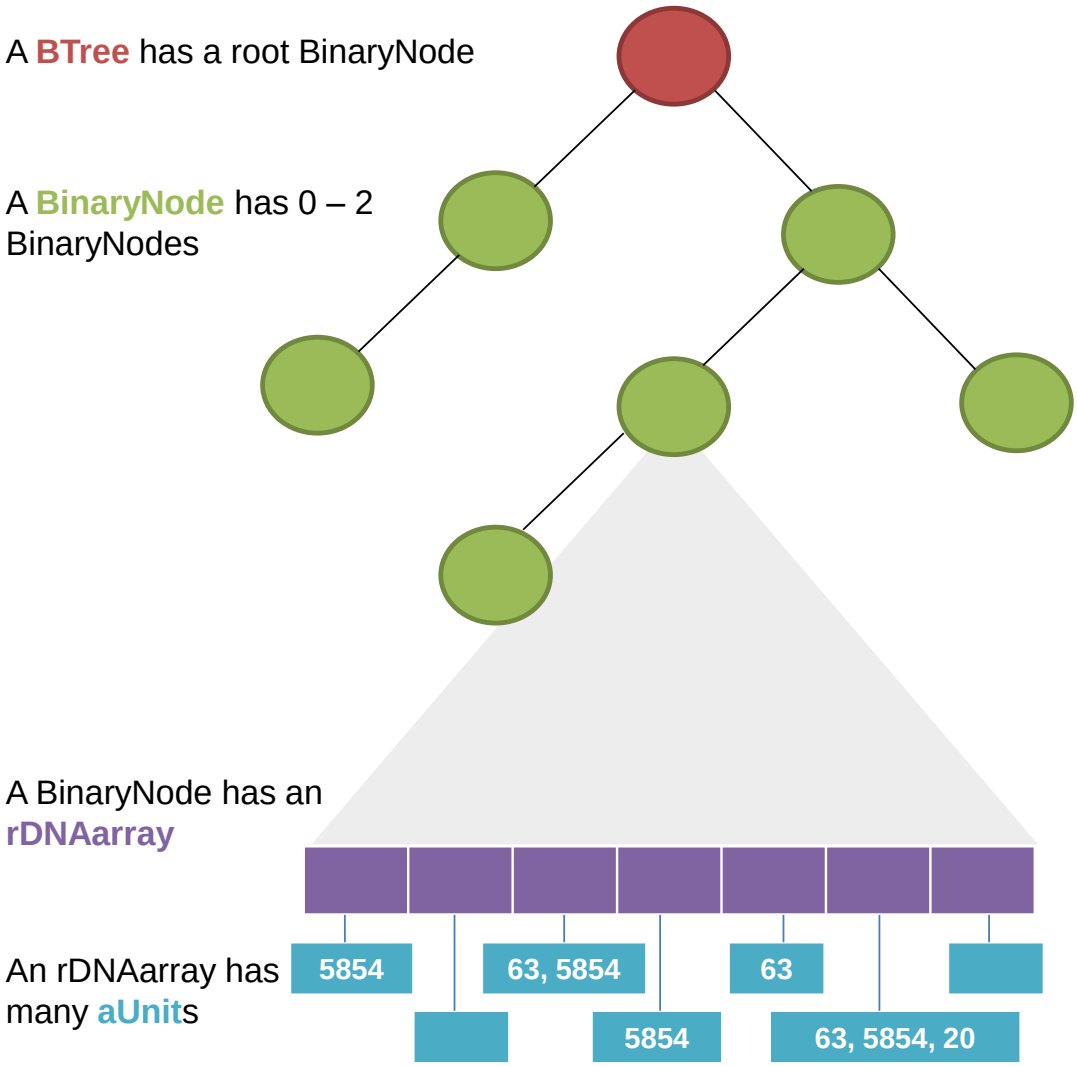

Figure 5.1.: Illustration of the hierarchical object structure in CONCERTINA. Blue **aUnit** objects contain different pSNPs, represented by different integers within each box.

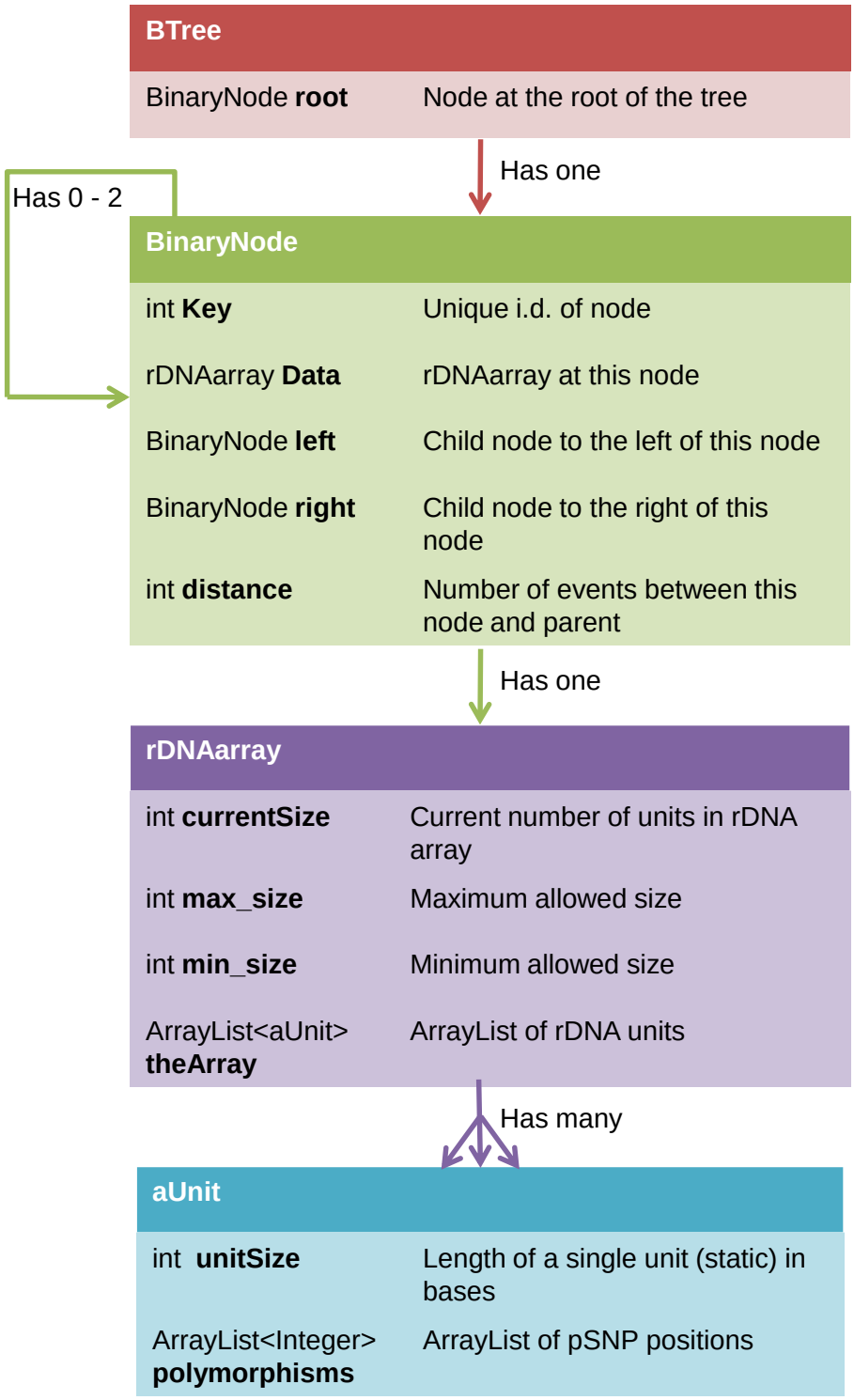

Figure 5.2.: Overview of the hierarchical object structure in CONCERTINA. Each box represents an object type, with the states of each object listed.

5.1.1. Changes to the Gene Conversion and USCE Methods within CONCERTINA

With the increase in complexity of code between SIMPLEX and CONCERTINA, moving from comparing rDNA array units in which there may be only one pSNP to those which could contain many polymorphisms necessitated changes to the gene conversion and USCE methods. Although the new methods follow the general flow of those used in SIMPLEX, as illustrated in figures 4.4 - 4.8, the details have changed to account for the possibility of multiple pSNPs within a unit.

The gene conversion class now contains two methods, gcOverwrite and compareUnits. The former accesses donor and acceptor units, determines whether the conversion tract will result in a subsequent unit being involved, and calls the compareUnits method. The compareUnits method first iterates through the polymorphisms in the donor unit, and checks if they are in the acceptor unit. If a pSNP is in the acceptor unit, but not in the donor, and is between the break position (start of the tract) and the end of the tract, this pSNP is removed from the acceptor unit. Each pSNP in the donor unit is then iterated through, and if the pSNP is in the donor, but not in the acceptor, and is within range of the tract, it is added to the acceptor.

The USCE methods are still split into deletions and duplications. Within the deletion method, pSNPs in the first unit's polymorphism list are removed if they are after the break, and pSNPs in the last unit's polymorphism list are removed if they are before the break. Then all of the remaining first unit's polymorphisms are added to the last unit, and all units from the first to the unit before the last are removed. Conversely, with the duplication method, all units from the second to the last are copied and added to the array of units, directly after the last unit. The original last unit is then altered, so that any pSNPs after the break in the first unit are copied to the end of the last unit. This is still essentially the same process as that shown in figure 4.4, except that as there are potentially a number of pSNPs to consider instead of just one, more loops are required to compare polymorphisms between the affected units.

5.1.2. Additional Classes in CONCERTINA

As well as the aforementioned objects and pre-existing, if altered, GCevents and USCEevents classes, a few extra classes have been added to CONCERTINA. As in SIMPLEX, the main class sets up the simulation and contains the variables for parameters such as the ratio of the different events, and the initial size of an rDNA array. In this case a simulation runs until a specified number of nodes have been added to the BinaryTree (rather than until a certain number of pSNPs have been lost or fixed). However, CONCERTINA also contains a few extra methods and classes which are now described.

Evolve Class

This class contains methods which determine the evolution of the array, both to choose whether a gene conversion or USCE event is undertaken for a particular step given earlier parameters (the evolve method), and to determine when a new mutation is added and where it is located (the mutate method). Although much of the code was present in SIMPLEX, it has been greatly reorganised in CONCERTINA, making future changes more easy to achieve.

The evolve method creates a loop to cause an array to undergo a certain number of concerted evolutionary events, and will call the GCevent and USCEevent methods according to the percentage of each event given as a parameter. It also implements the checks in the USCEevent method to ensure that the array size is maintained within the given limits.

The mutate method adds new functionality to the CONCERTINA program, by adding the ability to insert new pSNPs within the array at a chosen rate. Mutate is called from within the evolve method each time a concerted evolutionary event is undertaken. Parameters are set in the main CONCERTINA class for the mutation rate per base per generation, and for the number of concerted evolutionary events in the rDNA array per generation. The number of events until a point mutation is introduced is then calculated, where

- μ is the point mutation rate per base per generation
- n is the number of units of an rDNA array
- l is the length (number of bases) of an rDNA unit

- c is the number of concerted evolutionary events in an rDNA array per generation
- e is the number of events until a mutation within the rDNA array

The number of events is then simply:

$$e = \frac{c}{\mu \times n \times l} \tag{5.1}$$

The mutate method then checks if the current event should be accompanied by a point mutation (by dividing the current number of events by the calculated number of events until a mutation, and checking if the remainder is equal to zero). This results in point mutations occurring in a clock-like manner after a certain number of events. Alternatively, if the mutation rate is high compared to the number of concerted evolutionary events, more than one mutation could occur at each event. If a mutation is chosen to occur, a unit is chosen at random from the ArrayList. However, the position of the new pSNP within the selected unit is chosen by calling the rDNAregionWeight method.

rDNAregionWeight Class

The number of polymorphisms varies between distinct regions of an rDNA unit, as shown in earlier work (James et al., 2009), and in the rDNA analysis presented here of *S. cerevisiae* and *S. paradoxus* in Chapter 3, table 3.3. To emulate this variation in CONCERTINA a class, rDNAregionWeight, was written to weight the likelihood of a point mutation (pSNP) occurring in a region according to a given distribution, in this case that seen in the analysis of the SGRP data.

The size of each rDNA region (for example ETS1, 18S) in bases is input, as is the total number of pSNPs plus SNPs for each region, with the values as in table 3.3. These values can be changed in different runs of the program to allow for new knowledge or to compare different distributions. The percentage of pSNPs plus SNPs (from here on referred to as polymorphisms) found in each region is calculated (100 divided by the total number of polymorphisms across all regions multiplied by polymorphisms in the region in question) and added to an ArrayList. These percentages are then added together successively in a loop, and a random number between 1 and 100 is generated. If the resulting number falls within

Region	ETS1	18S	ITS1	5.8S	ITS2	26S	ETS2	IGS1	5S	IGS2
Polymorphisms	17	5	11	0	0	35	10	83	3	63
Weights	7.49	2.20	4.84	0.00	0.00	15.42	4.41	36.56	1.32	27.95
Range	7.49	9.69	14.54	14.54	14.54	29.96	34.36	70.93	72.25	100

Table 5.1.: Table illustrating an example of rDNA regional weighting for use in the rDNAregionWeight class. The top row shows the various rDNA regions, followed by the number of pSNPs + SNPs in each region. The number of polymorphisms in a given region is then represented as a percentage of the total number of polymorphisms. Finally the upper bound of the range that a number would fall within to generate a pSNP within that region is shown.

a certain range, a mutation will be generated within that region. This process is illustrated in table 5.1, with test data from earlier work (James et al., 2009). Using the data in table 5.1 as an example, if the random number 12 were to be generated, it would result in a pSNP within the ITS1 region, as it is greater than 9.69 (the upper limit of the range for 18S), but below 14.54, the upper limit for ITS1 in the table.

Once a region has been selected at random according to the weighting scheme, the location of a point mutation within the selected region then needs to be chosen. Another array is generated with the upper ranges of the position of each region. So for example, ETS1 is 699 bases long and is the first region in the unit. 18S is the second region and is 1799 bases long. Therefore its upper range position is (699+1799 =) 2498. The mutation position is then assigned by generating a random number between 1 and the size of the region chosen, and then subtracting it from the equivalent upper range in the array element containing that region. So, again using table 5.1 as an example, the region could be chosen by a random number generated as 12 (as in the previous paragraph), resulting in region ITS1 undergoing a point mutation. A random number is then chosen between 1 and 360 (the size of this region), for example 200. This number is then subtracted from the limit of the ITS1 region, 2858 (which is the ETS1 + 18S + ITS1 size), giving a final pSNP position of 2658.

This method is called for each new pSNP introduced to an rDNA array within CONCERTINA, to give a weighting to any pSNPs introduced.

5.2. CONCERTINA Experiments

Two sets of experimental simulation runs were undertaken using CONCERTINA. In all of these runs the mutation rate was varied, to investigate the dynamics of a pSNP's spread and loss from an rDNA array. The remaining variables were kept static:

- The ratio between USCE and GC remained at 20% and 80% respectively
- The gene conversion tract was static at 4000 bases in length
- The misalignment (the number of units between donor and acceptor, or number of units copied or deleted in USCE) was randomly selected between 1 and 10 units
- The number of events until the array is printed, set to be every 1000 events plus the first and last event.
- The unit in which a pSNP was introduced was chosen at random from the array. The base position possessing the pSNP was determined by the rDNAregionWeight class, using the pSNP and SNP weightings derived from the earlier *S. cerevisiae* results in table 3.3.
- The initial rDNA array size is 140, the maximum and minimum sizes are set to be 200 and 50 respectively. The unit length is 9137bp.
- The number of concerted evolutionary events per generation was kept static at 1 event per generation. This value is not experimentally known, but as it is the ratio between the point mutation and concerted evolutionary event rates that is under investigation, this value can be kept static and points mutation rates varied instead.

Although the point mutation rate is not uniform across the genome (Lang and Murray, 2008), one study estimated it to be 3.3 x 10^{-9} per base per cell division (Lynch et al., 2008). The mutation rate was set to vary between 6.6 x 10^{-5} and 3.3 x 10^{-10} mutations per base per generation, at rates shown in table 5.2. For each of the selected point mutation rates two sets of simulations were undertaken. In the first simulation, only one node was analysed, which would undergo 200,000 events. Any patterns of pSNP spread throughout the rDNA array could then be compared across the different rates. Each simulation was run three times, and the number of pSNPs (in bins of 0-<10% occupancy, 10-<20% and so on until 100% occupancy) was recorded every 1000 events. In the second set of experiments a random 10 node binary tree was generated for each point mutation rate, with a distance of 50,000 events between each node. In both experiments, a text file is

produced for each simulation run. The file contains a header with values for all of the variables, followed by a summary of the array at different points throughout the run. This summary consists of the current number of events undertaken, the array size, and the number of pSNPs in the occupancy bins. In the case of the 10 node binary tree runs, each node is numbered, with its placement within the tree noted (for example, "Node 1 added to the left of Node 0").

5.2.1. Experiment 1: Varying Mutation Rates Ratios for a Single rDNA Array

The results of the simulation runs involving a single node (rDNA array) undertaking 200,000 concerted evolutionary events are summarised in table 5.2. This table shows the numbers and occupancies of pSNPs in an rDNA array at the end of the run. Only point mutation rates of 10^{-7} or greater resulted in pSNPs which became fixed. The pattern of pSNP occupancies is similar for every mutation rate above 10^{-7}. The majority of pSNPs are found in the 0-20% and 90-100% bins, with very few pSNPs being found in the 30-80% bins, table 5.2. At rates less than 3.3 x 10^{-9}, no pSNP occupancies greater than 10% are seen. When the mutation rate was set to be 10^{-8} no pSNPs were fixed, but some higher occupancy pSNPs were observed.

The pattern of pSNPs spreading throughout the rDNA array to form the distribution shown in table 5.2 can be visualised in surface plots, by inputting the simulation run data into R and calling the persp function (R Development Core Team, 2011). Surface plots for those runs which resulted in a broad spread of pSNP occupancies, i.e. those with point mutation rates of 10^{-7} or greater, are shown in figure 5.3. The general pattern in all of these runs is that the 10 to 20% occupancy bin contains more pSNPs than those of the other, higher occupancies. Furthermore the occupancy bin frequencies are already established by 1000 events, being maintained throughout the rest of the simulation run, shown as a fairly steady value from 1000 to 50,000 events in all plots within figure 5.3. In those runs where the mutation rate is highest, a U-shaped distribution, skewed to the left, is established at 1000 events (the first recorded event), and is maintained thereafter. However, although this distribution is also established by 50,000 events for lower mutation rate runs, the number of events to establish it vary. This variation can be visualised when comparing the rightmost edges of these plots, which represent occupancies of 100%, such that the flat profile at this occupancy appears to change

Mutation Rate	Ratio PM:CE	<10	<20	<30	<40	<50	<60	<70	<80	<90	<100	100
6.6×10^{-5}	90:1	1552	700	670	639	503	326	180	72	37	33	289
3.3×10^{-5}	45:1	1526	892	507	280	230	150	86	32	49	83	262
6.6×10^{-6}	9:1	1265	212	125	47	19	35	38	22	10	50	145
3.3×10^{-6}	5:1	1058	134	40	12	13	16	4	6	2	23	50
6.6×10^{-7}	1:1	294	19	9	3	1	4	1	1	8	15	7
3.3×10^{-7}	1:2	110	12	5	3	1	1	0	0	1	11	7
6.6×10^{-8}	1:11	28	1	0	0	0	0	0	0	0	3	0
3.3×10^{-8}	1:22	11	1	1	0	0	0	0	0	0	1	0
6.6×10^{-9}	1:111	2	0	0	0	0	0	0	0	0	0	0
3.3×10^{-9}	1:221	0	0	0	0	0	0	0	0	0	0	0
6.6×10^{-10}	1:1106	0	0	0	0	0	0	0	0	0	0	0
3.3×10^{-10}	1:2211	0	0	0	0	0	0	0	0	0	0	0

Table 5.2.: Different point mutation rates (assumed to be genomic mutation rates per generation) for each run of 200,000 concerted evolutionary events, assuming 1 concerted evolutionary event per generation, with the equivalent ratio between mutations:concerted evolutionary events (PM:CE). The occupancies of the pSNPs present in the array after 200,000 concerted evolutionary events are given in bins of 10% intervals, with SNPs shown as 100% occupancy.

more gradually in plots representing lower mutation rates. Roughly triangular flat regions can also be seen in the bottom right hand corners, where the higher pSNP occupancies are slowly populated after increasing numbers of concerted evolutionary events.

In those runs where the spread of pSNPs across occupancies is more gradual, the U-shape continues to become more pronounced with increasing numbers of events. For example, this is illustrated in the surface plots for 50,000 and 200,000 events in figure 5.4 for a mutation rate of 3.3 x 10^{-6}, which is a ratio of 5 point mutations to every concerted evolutionary event. In the plot after 50,000 events, the number of pSNPs at 100% occupancy appears to have plateaued. However, completing the full run until 200,000 events have taken place shows the number of pSNPs at 100% has increased further.

The two previous figures (figures 5.3 and 5.4) only illustrate pSNP occupancies between 10% and 100%, to allow the patterns that form the end distributions to be visualised. The majority of pSNPs have occupancies of less than 10%, and are lost within 1000 events, as shown within the earlier SIMPLEX experimental simulations. When the point mutation rate is very high, many pSNPs are formed

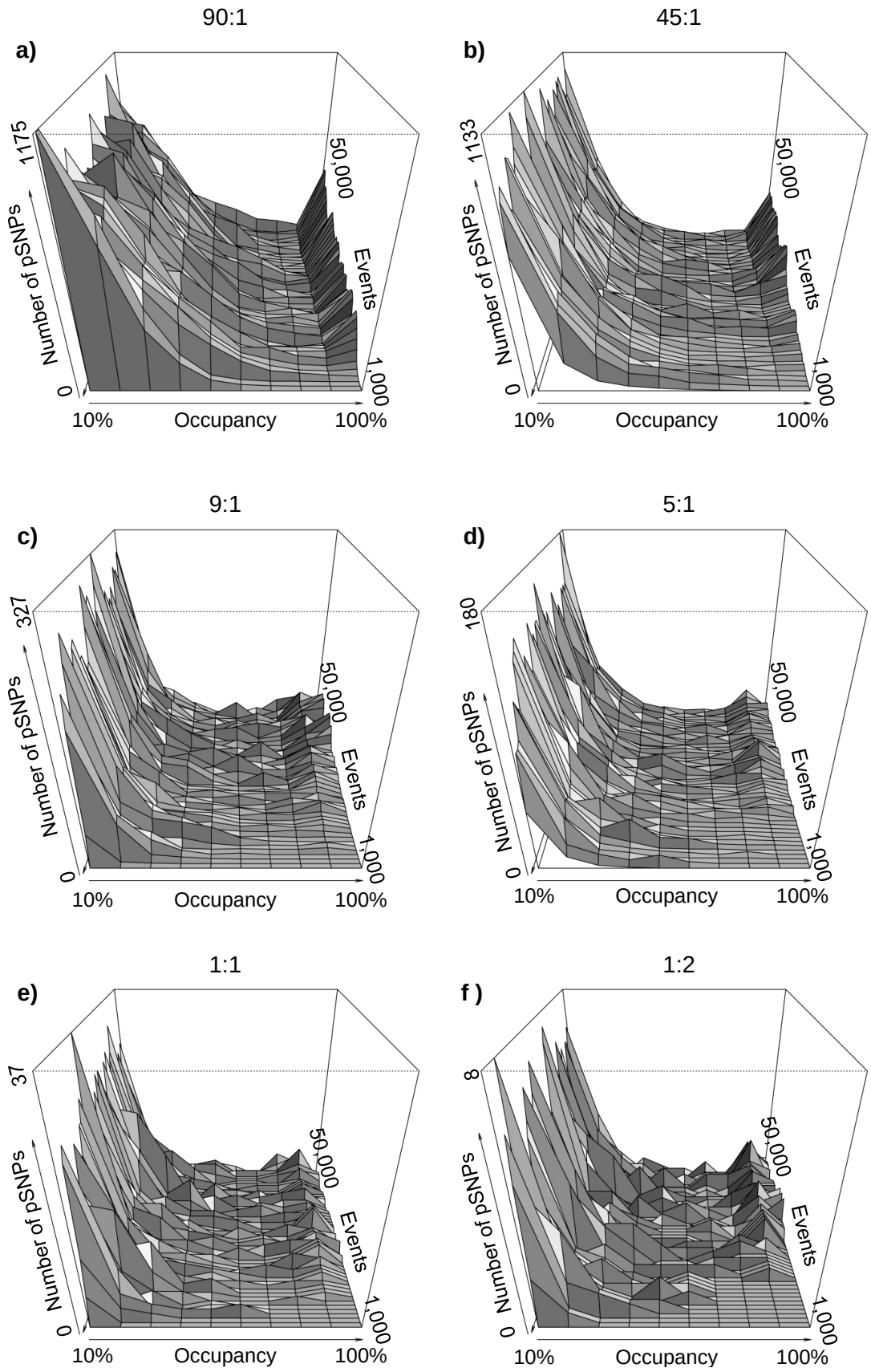

Figure 5.3.: Surface plots of pSNP occupancies of >10% to 100%, over 50,000 concerted evolutionary events, at different point mutation : concerted evolutionary event ratios, given at the top of each plot.

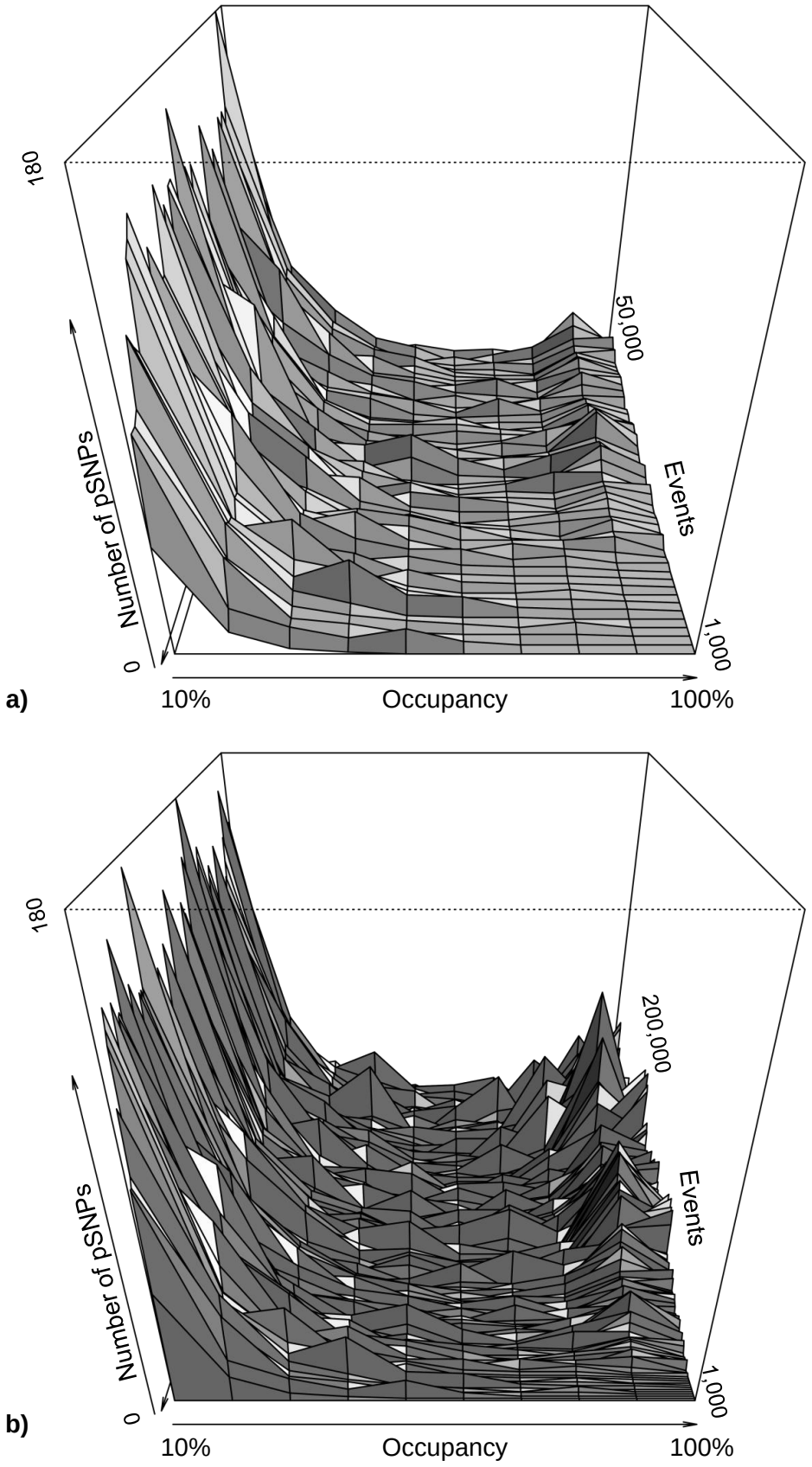

Figure 5.4.: Surface plots of pSNP occupancies of >10% to 100% for a point mutation rate of 3.3 x 10^{-6}, after a) 50,000 concerted evolutionary events and b) 200,000 concerted evolutionary events.

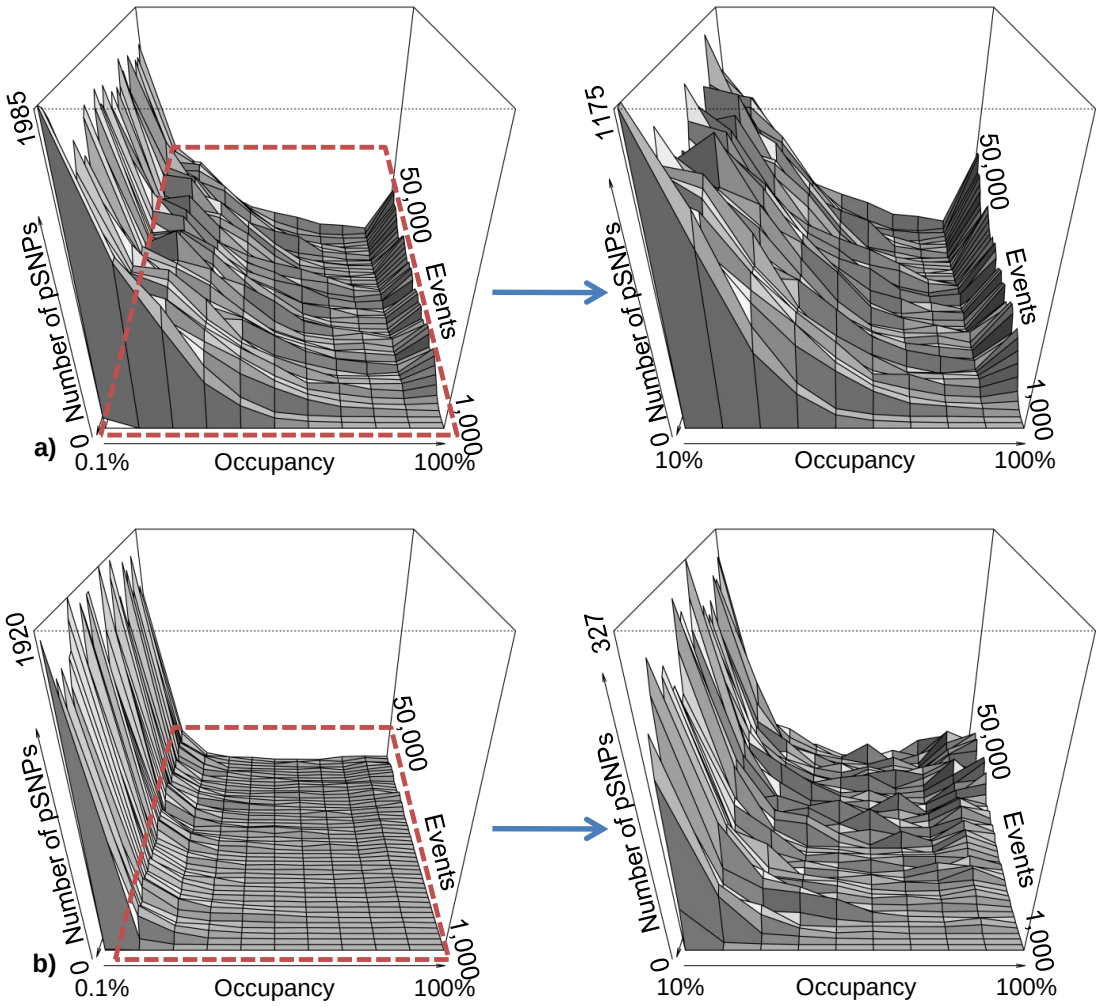

Figure 5.5.: Surface plots of a) 6.6 x 10^{-5} and b) 6.6 x 10^{-6}. Plots on the left are from >0% to 100% pSNP occupancy, with the red box highlighting results from >10% to 100%. The plots on the right are subsets of the plots on the left, restricted to occupancies of >10% to 100%.

within the rDNA array. Many of these polymorphisms will spread throughout the array as there are few concerted evolutionary events to remove them. However, as the ratio between point mutations and concerted evolutionary events decreases, it takes longer for the smaller numbers of pSNPs generated to increase their occupancies. This is illustrated in figure 5.5, where at the higher point mutation rate (top row, a)), the U-shaped distribution is visible when viewing all pSNP occupancies. However, for the lower mutation rate, the emerging, flatter, U-shaped distribution is not visible until the lower occupancy pSNPs (<10%) are removed (right plot of bottom row, b)).

5.2.2. Experiment 2: Varying Mutation Rate Ratios for Ten Diverging rDNA Arrays

The distribution of pSNP occupancies, as the ratio of point mutation to concerted evolutionary events was varied, was investigated in simulation runs with 10 node binary trees. Examples of three point mutation rates which resulted in a broad range of pSNPs occupancies are described below. As shown in table 5.2, any mutation rate less than 10^{-7}, results in a sparse pSNP occupancy distribution, and results for these mutation rates are not shown for the 10 node runs. The results for the highest mutation rate run (6.6×10^{-5}) is shown in figure 5.6. The first node shows the shape of the distribution after 50,000 concerted evolutionary events, which is maintained in all subsequent nodes, and is highly similar to that shown in figure 5.3. The distribution is highly skewed to the left, with a peak at 100% for all but the first node. For many nodes, the occupancies of the first two bins are more similar than those of other runs (see figures 5.7 and 5.8). 100% pSNP occupancies are also highly variable between nodes, ranging from just 50 at node 9, to 611 at node 0.

The trees in figures 5.7 and 5.8, for point mutation rates of 3.3×10^{-6} and 6.6×10^{-7} respectively, show very different pSNP distributions to those in figure 5.6. These two trees both exhibit a deeper U-shape than figure 5.6, with the least frequent bin (i.e. with fewest pSNPs) showing a slightly smaller occupancy value. Furthermore, the time taken to establish the U-shaped distribution (equilibrium distribution) varies between plots, being approximately 100,000 events in figure 5.7, and 150,000 events in figure 5.8. The plots also illustrate the stochastic nature of pSNP frequency. For example, in figure 5.7, nodes 2 and 6 are derived from the same parent node or rDNA array, but one has 70 pSNPs at 100%, whereas the other has only 12.

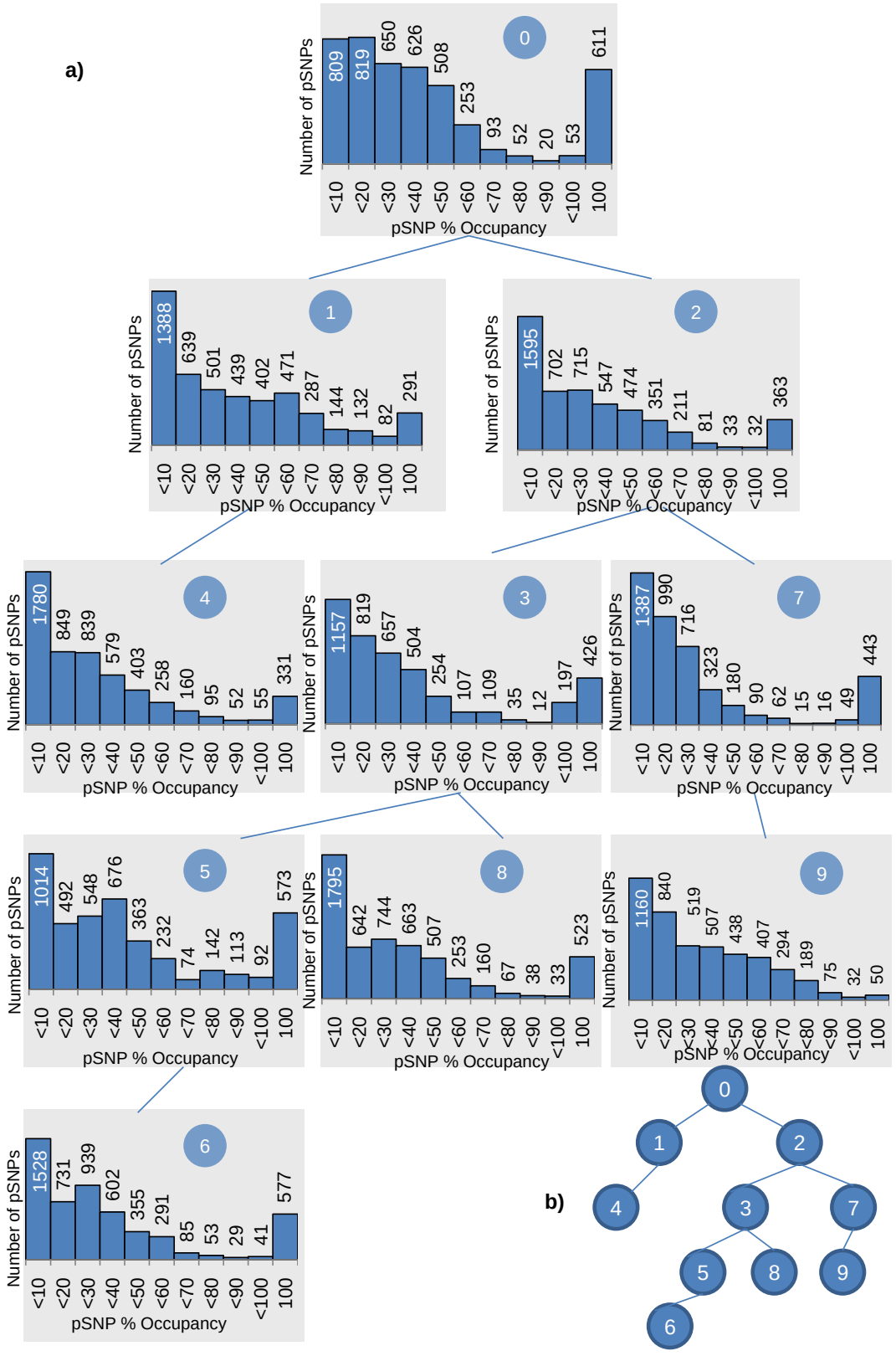

Figure 5.6.: a) Representation of a binary tree showing the results of a 10 node run, with a point mutation rate of 6.6 x 10^{-5}, and a 50,000 event distance between nodes. b) Overview of the shape of the tree, showing the order in which the nodes were added, in the bottom right. The number of pSNPs in each occupancy bin are shown in histograms.

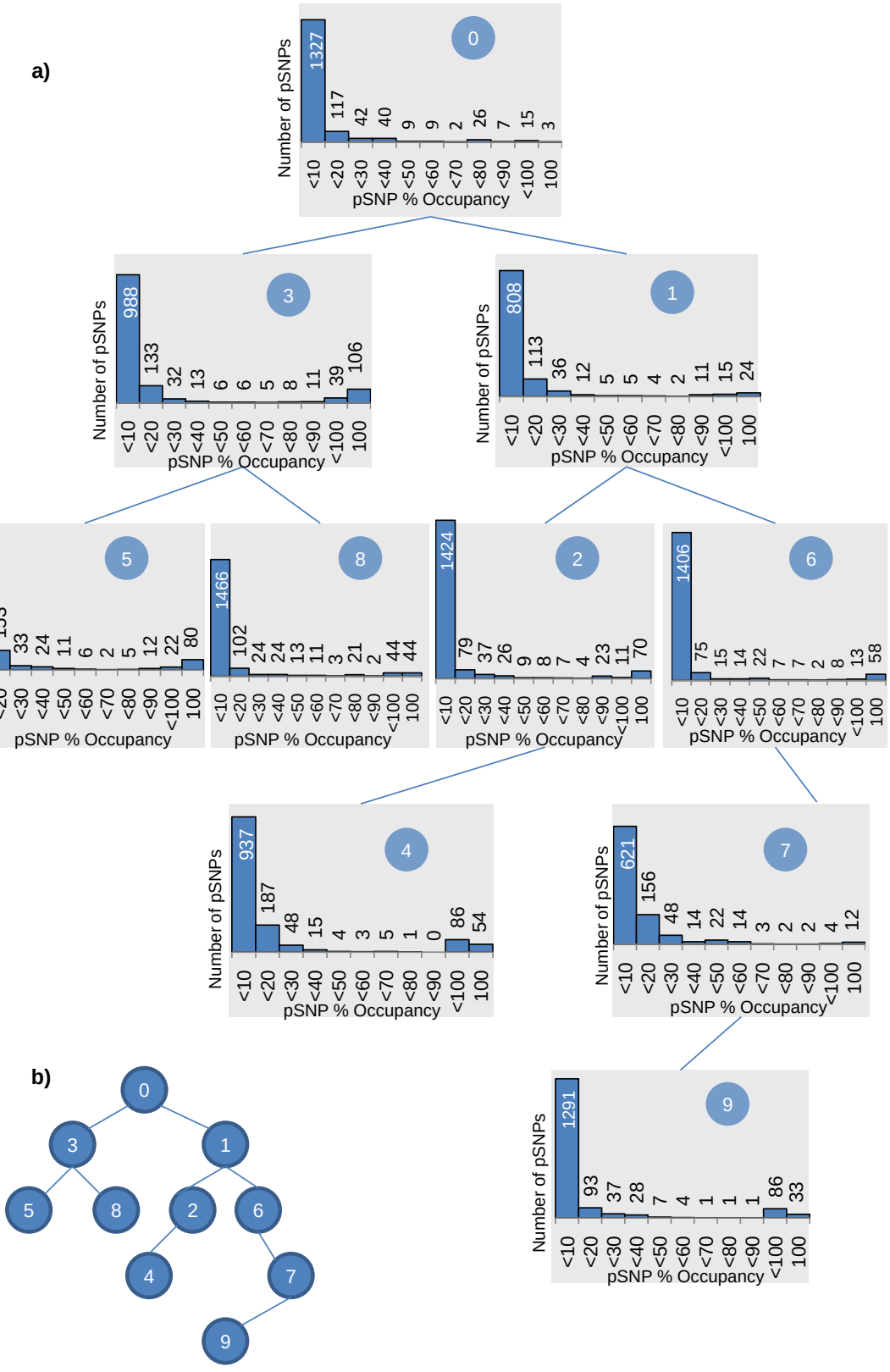

Figure 5.7.: a) Representation of a binary tree showing the results of a 10 node run, with a point mutation rate of 3.3 x 10^{-6}, and a 50,000 event distance between nodes. b) Overview of the shape of the tree, showing the order in which the nodes were added, in the bottom right. The number of pSNPs in each occupancy bin are shown in histograms.

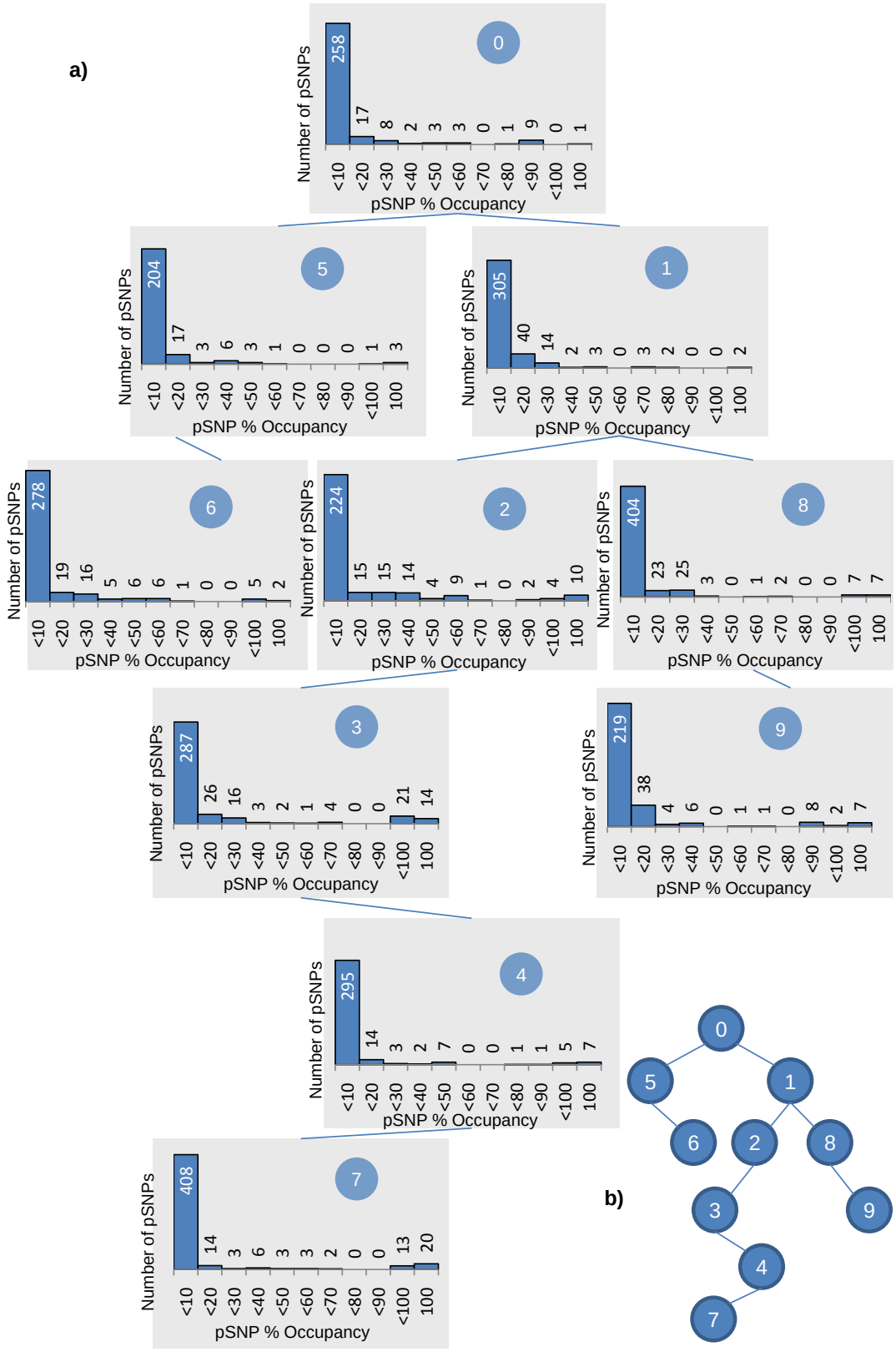

Figure 5.8.: a) Representation of a binary tree showing the results of a 10 node run, with a point mutation rate of 6.6 x 10^{-7}, and a 50,000 event distance between nodes. b) Overview of the shape of the tree, showing the order in which the nodes were added, in the bottom right. The number of pSNPs in each occupancy bin are shown in histograms.

5.3. Conclusions and Chapter Summary

The CONCERTINA software builds on the software framework of SIMPLEX to enable a continual process of point mutation balanced against the GC and USCE concerted evolutionary processes. Furthermore, CONCERTINA can simulate the evolution of both a single rDNA array, or sets of arrays, the latter related by a tree-like structure.

CONCERTINA was used to investigate the nature of the balance between divergent and concerted evolutionary rates. This balance of rates was found to strongly affect the shape of the pSNP occupancy distribution and the time taken to reach an equilibrium distribution. In particular, when concerted evolution was frequent compared to point mutation, few pSNPs were found. When the situation was reversed, a shallow U-shaped distribution resulted. For similar rates, a deep U-shaped distribution could be seen, with several pSNPs at very low or very high occupancy.

Based on these preliminary results and those presented in Chapter 3 for *S. paradoxus*, which exhibits tree-like evolution, it seems likely that the two rates are of a similar order, resulting in a deep U-shaped pSNP occupancy distribution with few pSNPs at intermediate frequencies.

6. rDNA Flanking Regions

Chapter Abstract

The concerted evolutionary process USCE, which can shorten or lengthen an rDNA array, is believed to result in an array consisting of complete rDNA units. However, the *S. cerevisiae* type strain S288c is thought to possess partial terminating rDNA units on each side of its array (SGD, 2013). The DNA sequence at the flanks of the rDNA array has been investigated in four yeast strains, using Pacific Biosciences SMRT sequencing. The left flank is conserved between the four strains, whereas the right flank varies between the industrial yeast strain S288c and the other three strains, two from *S. cerevisiae* and one from *S. paradoxus*. Furthermore, all eight flanking regions of the four rDNA arrays terminate in partial rDNA arrays.

6.1. Background

The sequences flanking the rDNA array, also referred to as the junctions, have been of interest for a number of years. In 1982 a study found single copy genes flanked the yeast rDNA, but that different strains had one of two alternative genes on the right flank (closest to the telomere) (Zamb and Petes, 1982). When the genome of *S. cerevisiae* type strain S288c was sequenced in 1997, the right junction proved difficult to sequence and was not present in the cosmid closest to the right end of the rDNA array, nor in phage lamda clones mapped to this region (Hillier et al., 1997). Ultimately, the right flank was inferred from PCR products close to the rightmost 5S rDNA subunit. The right flanking sequence was found to be similar to that discovered in earlier work (McMahon et al., 1984). The order of rDNA units and genes flanking them are shown in figure 6.1.

In *S. cerevisiae* strain S288c, the *ACS2* gene is approximately 4kb from the left junction, on the centromeric side of the rDNA array (Hillier et al., 1997). The Acs2p protein is located mainly in the nucleus, and mediates synthesis of

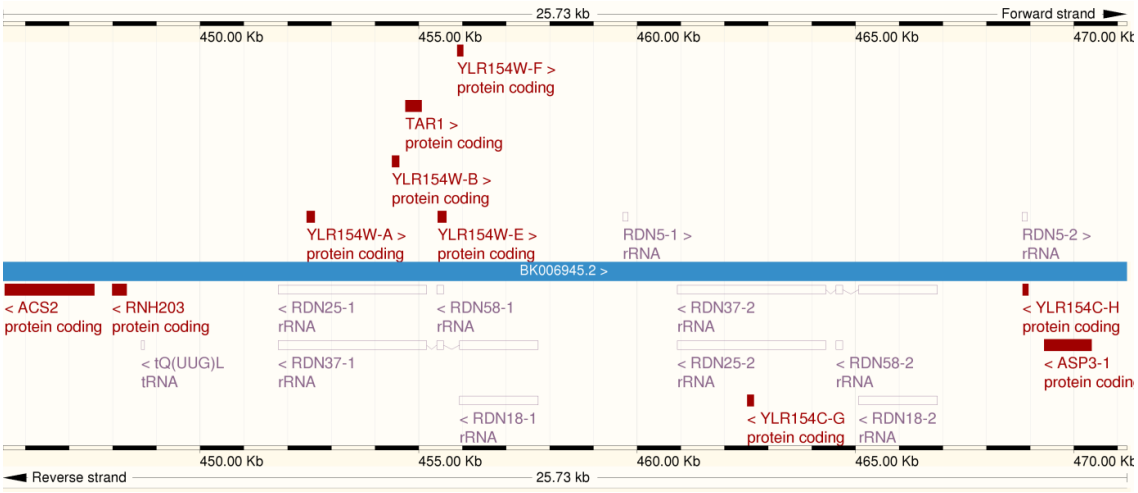

Figure 6.1.: Layout of rDNA in ENSEMBL Fungi (http://fungi.ensembl.org), S288c rDNA and flanking genes, Chromosome XII co-ordinates 445482-471206 shown. rRNA regions are shown in purple, and coding genes are shown in red.

acetyl-CoA. *ACS2* deletion strains contain more ERCs within their nucleus, and so there are indications that this protein is involved in promoting rDNA silencing, reducing ERCs and so increasing replicative lifespan in yeast (Falcón et al., 2010). The rDNA array at this junction ends in a partial rDNA unit, terminating in a partial IGS1 sequence.

In some *S. cerevisiae* strains, including S288c, the *ASP3* locus is closest to the right junction. In S288c the locus comprises a cluster of four identical *ASP3* genes interspersed with variant forms of the 5S sequence. *ASP3* encodes cell-wall associated L-asparaginase II, and is switched on during limited nitrogen conditions. A recent study investigated the origins of *ASP3*, and found it present in differing copy number in industrial or laboratory strains only, hypothesizing it was passed from the wine yeast *Wickerhamomyces anomalus* via horizontal gene transfer, conferring an advantage to harbouring strains in artificial environments (League et al., 2012). The rDNA array at this junction again ends in a partial rDNA unit, here terminating in a partial 26S sequence.

Upstream of the *ASP3* gene cluster, the next gene of known function is *MAS1*, approximately 1.5kb from the final variant 5S sequence in the *ASP3* locus. Mas1p is part of the mitochondrial processing protease, and cleaves targeting sequences from proteins which have been imported into mitochondria (Witte et al., 1988).

Earlier work investigating rDNA flanking sequences within the SGRP dataset

was carried out by MSc student Prashanth Kumar (Kumar, 2011). He sought to identify flanking reads via sequence matching to the S288c flanking sequence downloaded from SGD (SGD, 2013). Using this approach he successfully identified putative reads covering the left flank for 12 out of the 38 strains investigated. However, he was unable to identify the left flanking sequences of the remaining strains or the right flanking sequences of any of the strains. This suggested that variability of the flanking sequence between strains necessitated a different approach to their discovery.

We decided to investigate the rDNA flanking sequences of four strains. Any uncovered sequence variability at the rDNA junctions might then inform future modelling of the dynamics of rDNA arrays. Some recently developed Next-Generation Sequencing technologies are capable of producing long read lengths (several kb). Applying such a technology to these yeast strains should result in a few reads covering both a large proportion of the rDNA terminal units and the start of the flanking sequence, enabling a comparison between these sequences across the four strains to be made.

6.2. Methods

6.2.1. Data

Three *S. cerevisiae* strains were selected for analysis: the *S. cerevisiae* reference strain S288c; YIIc17_E5, a mosaic wine strain; and Y12, a structured mosaic wine strain. YIIc17_E5 and Y12 were chosen as examples of mosaic and structured strains, as earlier analysis had revealed both to contain a moderate number of SNPs and pSNPs. One *S. paradoxus* strain, CBS432, the European reference strain, was also chosen.

These four strains were sequenced by a single molecule real time sequencing method (also known as SMRT), using a Pacific Biosciences (PacBio) RS sequencer, with an 8-12 kb insert library, and 6 SMRT cell runs per strain. Sequencing was undertaken by GATC Biotech. The PacBio SMRT technology provides long sequence reads, which overcome some of the limitations of other NGS technologies (Roberts et al., 2013). Although error rates are high, errors are randomly distributed and unbiased to particular sequence motifs. These frequent but random errors might prove

Strain	Number of Reads	N50 (bp)	GC%
S288c	36,013	6,392	38.58
YIIc17_E5	36,357	6,641	38.15
Y12	33,902	6,642	38.43
CBS432	37,802	6,371	38.86

Table 6.1.: Details of PacBio corrected reads for each strain.

difficult for detecting pSNPs, but for examining variation at the end of the rDNA array, rather than individual polymorphisms within it, PacBio sequencing is a good approach.

Details of the sequence reads produced from the SMRT cell runs are shown in table 6.1. In the subsequent analysis, unassembled, corrected reads were used, which had been filtered using a Hierarchical Genome-Assembly Process (or HGAP). This process used subreads (filtered for quality), that were below a length threshold, to correct filtered reads above the length threshold. This correction process is believed to improve read accuracy, reducing some of the inherent unbiased sequencing errors outlined above.

6.2.2. Analysis

Corrected reads from each strain were filtered using an adapted version of the Perl script filter_reads_v3.pl, introduced in Chapter 2. The read names were first systematically altered to remove any forward slashes, which were parsed incorrectly by the script.

Genomic sequences of a single rDNA unit, and of the *ASP3*, *ACS2* and *MAS1* genes in yeast strain S288c were downloaded from the SGD database (SGD, 2013), in FASTA format. For processing of the *S. paradoxus* strain CBS432, an additional FASTA sequence of the *MAS1* gene for that strain was downloaded from the SGRP website (SGRP, 2013). A blast database was constructed from each of these FASTA files using the makeblastdb command (BLAST version 2.2.27+). Each strain was first filtered with filter_reads_v3.pl by blast-ing against the rDNA database, such that reads passed the filter if they had more than 90% identity (sequence similarity) with the rDNA sequence in greater than 25% of the read length. The subset of reads that passed this filter was then re-run through the

Strain	**vs rDNA**	**+ *ACS2***	**+ *ASP3***	**+ *MAS1***
S288c	2485	3	1	N/A
YIIc17_E5	1535	6	N/A	8
Y12	2226	3	N/A	10
CBS432	1378	1	N/A	13

Table 6.2.: Details of the number of reads which passed each stage of the filter. N/A refers to strains where this gene is not the closest to the flank.

script, this time blast-ing against the *ACS2* (left flank) and *ASP3* or *MAS1* (right flank) genes. The number of reads to pass each filtering step are shown in table 6.2. No length requirements were made in the second filtering step, but a threshold of 90% sequence similarity was still enforced.

Reads passing this double filter were then blast-ed against individual regions of the rDNA unit to identify the composition of the terminal rDNA units. This process comprised of creating a blast database of all ten of the individual regions of an rDNA unit. Finally, each filtered read was blast-ed against the *ACS2* (13 left flank sequences) or *ASP3*/MAS1 (32 right flank sequences) databases, resulting in a characterisation of the flanking sequences with regard to both the rDNA sequence and the flanking genes.

6.3. rDNA Left Flank: *ACS2* Gene

All thirteen of the left flank sequences derived from the four analysed strains matched to the *ACS2* gene at the left flank of the rDNA. In all cases, the rDNA terminates in a partial rDNA unit, ending with a partial IGS1 region (between 144 and 161 base pairs in length), approximately 16% of the usual IGS1 size (see table 6.3).

An illustration of the placement of the nine reads from strains S288c and YIIc17_E5 matching this flanking region is shown in figure 6.2. All reads span a region from the *ACS2* gene to the 26S region of the leftmost (partial) rDNA unit. The distance between *ACS2* and the partial IGS1 region varied between 2,862 and 3,916 bp in length. The placement of the four filtered reads from strains Y12 and *S. paradoxus* strain CBS432 are shown in figure 6.3. These reads extend further into the leftmost rDNA unit, with the CBS432 read matching to part of the 18S

Strain	Read	*ACS2* match (bp)	*ACS2* to IGS1 Distance (bp)	IGS1 Match (bp (%))
S288c	1	153*	3862	156 (17%)
	2	1528*	3914	160 (17%)
	3	2070	3888	156 (17%)
YIIc17_E5	1	2110	3901	150 (16%)
	2	2070	3879	154 (17%)
	3	2078	3900	158 (17%)
	4	87*	3886	146 (16%)
	5	729*	3890	144 (16%)
	6	2078	3916	148 (16%)
Y12	1	1280*	3975	151 (16%)
	2	83*	3902	152 (16%)
	3	1065*	3925	158 (17%)
CBS432	1	332*	4768	161 (17%)

Table 6.3.: Length of matches to the *ACS2* gene closest to the left flank, the terminal partial IGS1 region and the intervening sequence for each read. Numbers in brackets are the percentage of the IGS1 region found (as it is a partial region). * denotes a partial match, as the read ends within this region.

region. Y12 possesses a longer sequence between *ACS2* and the rDNA array than the other two *S. cerevisiae* strains, of approximately 3,902-3,975 bp. However, *S. paradoxus* appears to have the longest intervening sequence of the four strains, of approximately 4,768 bp in length (table 6.3). This left flanking sequence structure is the same as that previously described in earlier studies (Hillier et al., 1997), as illustrated in figure 6.1, and is conserved between the wild yeast *S. paradoxus*, and the mosaic and structured strains of *S. cerevisiae*.

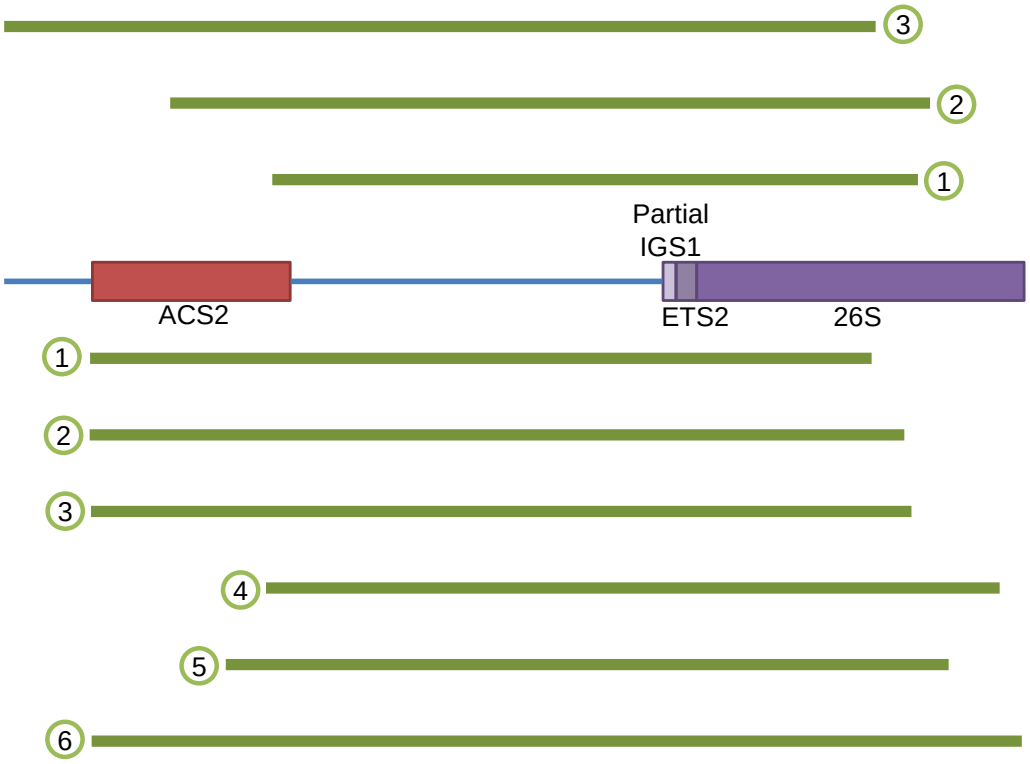

Figure 6.2.: Schematic diagram representing the position of reads which matched to both the *ACS2* sequence, and the rDNA array. Reads from strain S288c are shown above the left flank, and those from strain YIIc17_E5 below.

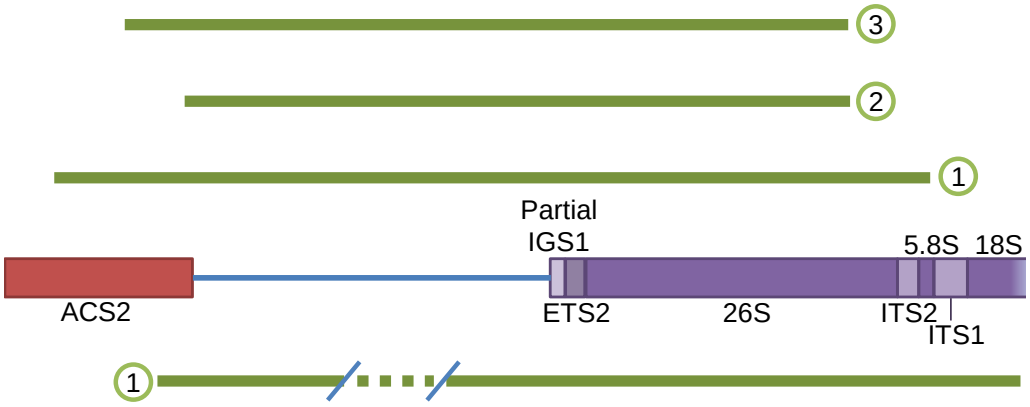

Figure 6.3.: Schematic diagram representing the position of reads which matched to both the *ACS2* sequence and the rDNA array. Reads from strain Y12 are shown above the left flank, and the single read from strain CBS432 below. The CBS432 read exhibits a longer distance (over 4,700 bp compared to approximately 3,900 in the *S. cerevisiae* strains) between the *ACS2* gene and the rDNA array, represented as a dotted line.

6.4. Right Flank: *ASP3* and *MAS1*

The right flank was found to be more variable in sequence between the four strains. *S. cerevisiae* S288c was the only strain to contain the *ASP3* gene cluster, with only one read passing the dual rDNA/*ASP3* filter. This read spanned all of the (leftmost) *ASP3* gene, and extended to cover most of the rightmost rDNA unit, from the partial 5S region at the terminal end of the unit to the 5.8S region, as illustrated in figure 6.4. The distance between the partial 5S region and the *ASP3* gene (shown in table 6.4) is slightly longer than that shown in the SGD (SGD, 2013) (413 bp compared to 386), but whether this is the result of sequencing error or sequence variation is currently uncertain.

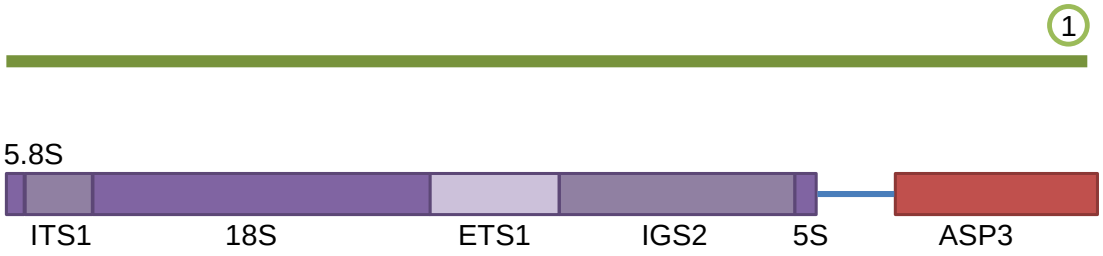

Figure 6.4.: Schematic diagram representing the position of reads which matched to both the *ASP3* sequence and the rDNA array. A single read from strain S288c is shown above the right flank.

The other two *S. cerevisiae* strains analysed (Y12 and YIIc17_E5) did not contain the *ASP3* cluster, instead possessing the *MAS1* gene closest to the rightmost rDNA unit. The location of the reads covering the right flank are shown in figure 6.5, some extending to span most of the terminal rDNA unit (only missing the ITS2 region), and others extending past the *MAS1* gene. In both strains the rightmost end of the rDNA array terminates in a partial rDNA unit, ending with a small fragment of the 26S region (approximately 24% of a full 26S sequence, see table 6.5).

Strain	Read	*ASP3* match (bp)	*ASP3* to 5S Distance (bp)	5S match (bp(%))
S288c	1	1111	413	115 (95%)

Table 6.4.: Length of matches to the *ASP3* gene closest to the right flank, the terminal partial 5S region and the intervening sequence for each read. Numbers in brackets represent the percentage of the 5S region found (as it is a partial region).

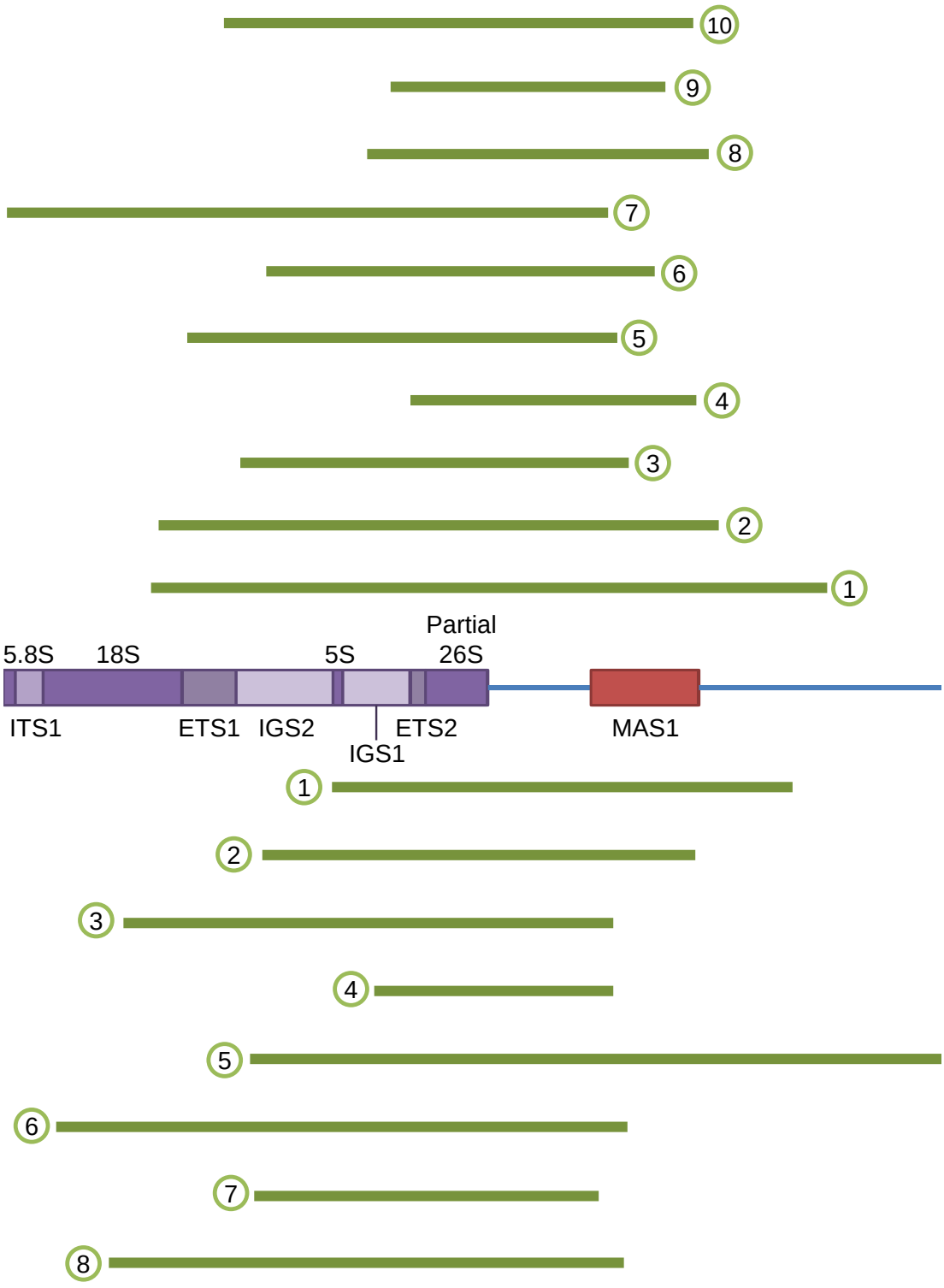

Figure 6.5.: Schematic diagram representing the position of reads which matched to both the *MAS1* sequence and the rDNA array. Reads from strain Y12 are shown above the right flank, and those from strain YIIc17_E5 below.

This structure of the right flanking sequence is also seen in *S. paradoxus*, where the rightmost unit is again a partial rDNA unit terminating in a small fragment of the 26S region (figure 6.6). The CBS432 reads do not extend as far into the last unit as those in Y12 or YIIc17_E5, but still reach the tip of the 18S region, and beyond the *MAS1* gene. The distance between the rDNA array and the *MAS1* gene appears to be longer in this strain than in the two *S. cerevisiae* strains, at approximately 1,500 bp in length, see table 6.5.

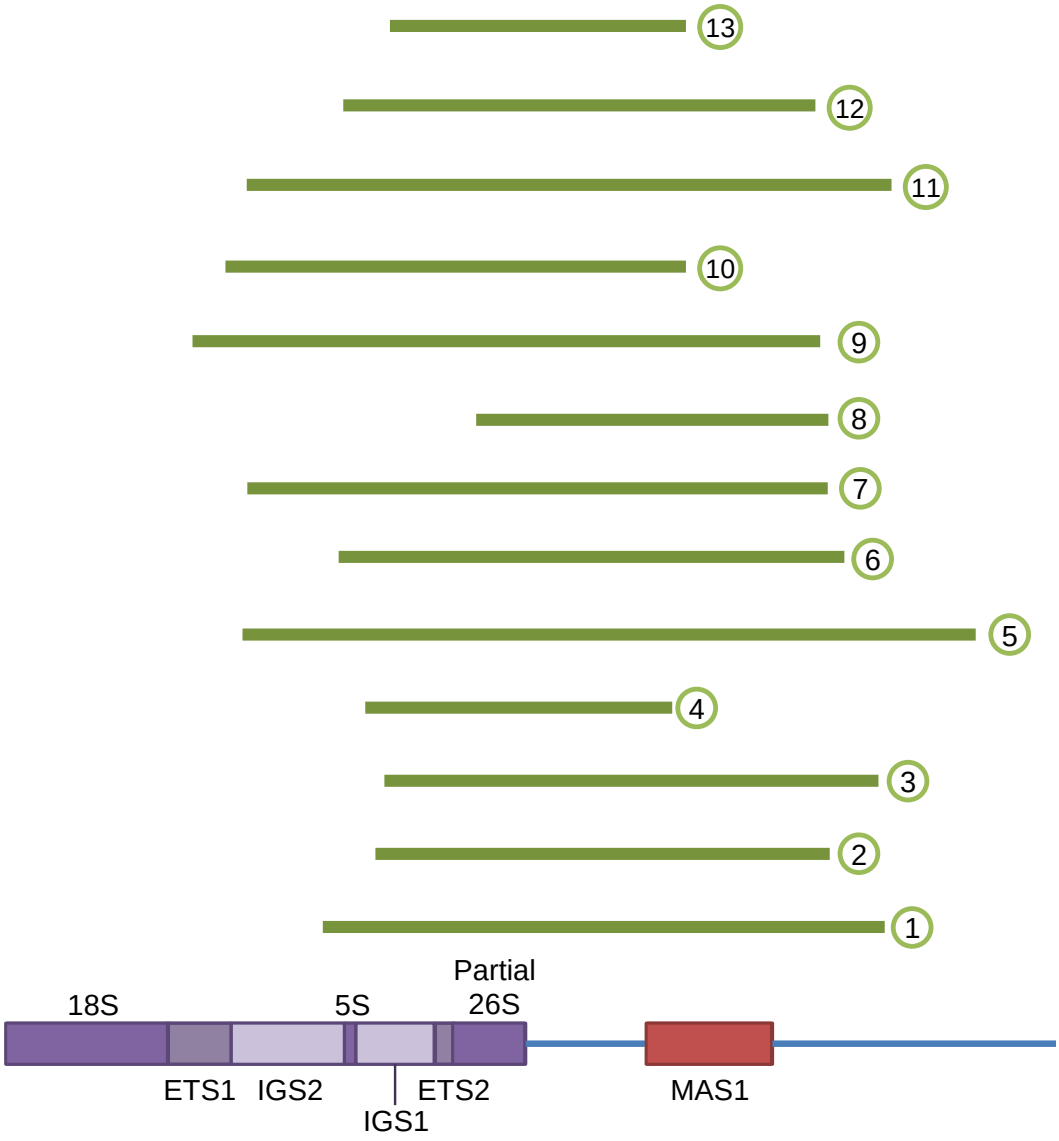

Figure 6.6.: Schematic diagram representing the position of reads which matched to both the *MAS1* sequence and the rDNA array. Reads from strain CBS432 are shown above the right flank.

Strain	Read	*MAS1* match (bp)	*MAS1* to 26S distance (bp)	26S match (bp(%))
YIIc17_E5	1	1407	1319	797 (23%)
	2	1157	1322	796 (23%)
	3	534*	1319	801 (24%)
	4	411*	1350	817 (24%)
	5	1417	1338	811 (24%)
	6	343*	1317	797 (23%)
	7	74*	1326	804 (24%)
	8	334*	1283	866 (26%)
Y12	1	1403	1344	801 (24%)
	2	1398	1342	800 (24%)
	3	332*	1346	801 (24%)
	4	1413	1360	819 (24%)
	5	231*	1328	798 (23%)
	6	1071	1336	803 (24%)
	7	221*	1343	789 (23%)
	8	1408	1350	801 (24%)
	9	1062	1366	824 (24%)
	10	1402	1336	802 (24%)
CBS432	1	1424	1507	809 (24%)
	2	1412	1499	816 (24%)
	3	1414	1499	811 (24%)
	4	369*	1530	824 (24%)
	5	1422	1506	820 (24%)
	6	1412	1492	811 (24%)
	7	1410	1490	810 (24%)
	8	1472	1513	245* (7%)
	9	1429	1497	810 (24%)
	10	366*	1494	809 (24%)
	11	1413	1518	818 (24%)
	12	1426	1493	817 (24%)
	13	507*	1509	814 (24%)

Table 6.5.: Length of matches to the *MAS1* gene closest to the right flank, the terminal partial 26S region and the intervening sequence for each read. Numbers in brackets are the percentage of the 26S region found (as it is a partial region). * denotes a partial match, as the read ends within this region.

6.5. Analysis of the rDNA Boundaries

In all of the four strains analysed, the left flank of the rDNA terminated in a partial rDNA unit, ending in a partial IGS1 region. This structure was consistent with that found previously, although the distance between the rDNA array and the *ACS2* gene appears to be slightly longer in strain Y12, and longer still in *S. paradoxus* strain CBS432.

However, the right flank varies between the four strains. *S. cerevisiae* reference strain S288c possesses the *ASP3* gene cluster upstream of a partial rDNA unit terminating in a variant 5S region. In contrast, strains Y12, YIIc17_E5, and CBS432 each possess a partial rDNA unit ending in a fragment of the 26S region, with the closest gene being *MAS1*. This right flank is likely to be the ancestral structure for this region as it is shared by Y12, basal to S288c in the *S. cerevisiae* phylogenetic tree, and *S. paradoxus* strain CBS432. The ends of the rDNA array also appear to be maintained across these strains, with the 26S fragment being of a similar size in all three. As discussed in a previous study (League et al., 2012), the *ASP3* gene cluster is present in varying copy number within industrial and laboratory strains of *S. cerevisiae*, likely resulting from a horizontal gene transfer event from the yeast *Wickerhamomyces anomalus*. The insertion of this gene cluster (or part of it if it subsequently expanded *in situ*) may have removed part of the rDNA array, potentially explaining the presence of an alternative partial rDNA unit in the right flank of S288c. A comparison of the flanking sequences of other industrial strains would be interesting in this regard, to investigate whether the right flank of the rDNA array in *ASP3*-containing strains always terminates in a variant 5S region. Although this variant 5S region is thought to be transciptionally active (McMahon et al., 1984), it would be unlikely that the partial 26S region in the other strains could be actively transcribed or used, particularly as only ∼50% of rDNA units are thought to be transcribed (McStay and Grummt, 2008). If that is the case, the conservation of the 26S regions across the strains is intriguing.

A multiple sequence alignment of reads from a single strain was attempted using the MAFFT (Katoh and Standley, 2013) software. Surprisingly, the alignment suggested a high number of sequencing errors in the PacBio reads, many of which appear to be slippage type errors, resulting in the duplication of a single base one or more times in a single read only. In future, multiple sequence alignments of the 13 left flanking sequences and the 32 right flanking sequences will characterise sequence variation between strains within these regions at a finer level of detail.

However, it would be helpful to resolve the slippage errors before carrying this out. To do this a consensus sequence of these reads could be attempted, or further correction of the reads could be undertaken, for example by correcting the PacBio reads using Illumina sequencing reads.

Lastly, the differences between the right flanks of industrial strains containing the *ASP3* cluster, and the conserved nature of those without it, could present an opportunity to create a rapid screening assay for *ASP3*, by designing primers to this *MAS1*/26S flank. However, a detailed assessment of sequence variability across a broader number of strains would be needed first, and the current prevalence of slippage errors in the PacBio sequence reads would need to be resolved.

6.6. Conclusions and Chapter Summary

The terminal sequences of the rDNA array across four selected yeast strains are confirmed to be partial rDNA units, some of which are conserved between strains. The left flank terminates in a partial IGS1 sequence, with the *ACS2* gene falling approximately 3,900 bp away. The right flank varies between S288c, in which the rDNA terminates in a variant 5S region followed by the *ASP3* cluster, and the other three strains, which terminate in a partial 26S sequence and the *MAS1* gene. This latter organisation of the right flank is likely to be the ancestral state.

This analysis has created a broad framework for the structure of the rDNA array flanking regions in *S. cerevisiae* and *S. paradoxus*. Further work on flanking region sequence variation between the four selected strains, and new investigations on the structure of the flanking regions in other related strains, will provide valuable new knowledge on the dynamics of this important genomic region.

7. Discussion

Ribosomal DNA is a highly dynamic area of the genome, upon which the mechanism of concerted evolution acts quickly to homogenize introduced variation. Micro-heterogeneity between different repeats of an rDNA array, such as partial Single Nucleotide Polymorphisms (pSNPs), provides a snapshot of these homogenising processes in action. Patterns of variation were identified both within and between the rDNA of two closely related yeast species. This variation then provided a focus for a series of simulation experiments that investigated the dynamics of concerted evolutionary processes.

Variation in rDNA was discovered within 34 strains of *S. cerevisiae* and 26 strains of *S. paradoxus* using the TURNIP software. Subsequent analysis revealed varying levels of sequence heterogeneity both within and between the rDNA arrays of individual yeast strains, including the recently discovered pSNP variation type. Phylogenetic relationships inferred from the identified rDNA polymorphisms have been shown to mirror those of previous whole-genome wide analyses, and distinct distributions of pSNPs have been discovered within the two species datasets. The results from this analysis of the yeast datasets informed preliminary work on the development of software tools to simulate concerted evolutionary processes. Subsequent simulation experiments suggest similar rates of point mutations to concerted evolutionary events may have led to the pSNP patterns observed in the yeast datasets.

7.1. Variation Discovery

Initial work on identifying and removing software bugs from the TURNIP variation discovery software, and on discovering and eliminating potential contamination from the yeast datasets emphasized the need for methods to assess the different steps of the variation discovery pipeline. At present, the first step in the pipeline is a script that filters yeast sequencing reads to those which contain rDNA sequence (known as read clipping). It relies upon choosing appropriate BLAST parameters that remove non-rDNA reads, whilst retaining potentially divergent

rDNA sequences. This filtering step necessarily includes manually checking the rDNA status of a subset of reads that are retained or lost after filtering, a process which is hugely time consuming. An alternative filtering procedure, or an automated checking process, would be a priority for future TURNIP development.

A new validation script was developed to assess TURNIP's performance and sensitivity. The script produces a simulated rDNA sequence read dataset containing known variation. This enables the modification of TURNIP's parameters for rDNA variation discovery, increasing confidence that TURNIP analyses are capable of capturing true rDNA variation while removing false positive polymorphisms. Testing suites have also been developed for other software, including reference-free SNP detection tools (Dou et al., 2012). However, these testing suites are unsuitable for rDNA sequence, as indeed are the tools themselves. Hence, a slightly different approach and script was needed to generate appropriate datasets for the case of rDNA.

Using these scripts in tandem with the TURNIP software, 1,168 and 978 polymorphisms were identified within the *S. cerevisiae* and *S. paradoxus* datasets respectively. In *S. cerevisiae*, fewer single point mutations in strains were identified than in one previous study of the same dataset (James et al., 2009). The discrepancy in number of identified pSNPs between these results and the former study were attributed to different software tools (that used in the previous study is not publicly available to our knowledge) and to the stricter filtering parameters in this study. However, in *S. paradoxus* similar levels of variation were identified to that of another analysis of a single, distinct strain (Ganley and Kobayashi, 2007).

Although our methods successfully identified rDNA variation within the SGRP datasets, very few studies now use Sanger sequencing. In future, the methodology and software would need to be updated for use with the more common Next-Generation Sequencing (NGS) datasets. This would enable many more datasets to be analysed, including publicly available data such as mutation accumulation lines (e.g. Nishant et al., 2010) to discover changes in rDNA variation over time. To achieve NGS analysis using TURNIP, a number of technical issues would need to be overcome.

Firstly, the manual checking step that currently follows read clipping presents a barrier to TURNIP analysis of short-read NGS datasets, due to the large number

of reads. The choice of appropriate BLAST parameters for short reads would also be a problem. In addition, TURNIP currently possesses a sequence read length requirement, added to increase specificity to the rDNA region. This would need to be removed or greatly reduced for short read lengths, potentially lowering the quality of results.

The TURNIP validation script would need to be altered to produce short length reads, a change which should be simple to enact. However, validation should also include assessing the effect of the different sequencing errors that can be introduced using different NGS technologies. For example, Illumina sequencing reads contain few biased errors, whereas PacBio sequencing reads contain many uniformly distributed errors. Adding an error profile to the validation script would enable a user to test whether the variation have uncovered would be detectable by the extended TURNIP software.

Currently, any variation detected by TURNIP is treated as a sequencing error if it is only found in a single read. This may mean some genuine variation is discarded, presumably particularly low occupancy unresolved polymorphisms. This is especially important given the results of simulation experiments, which show the majority of any variation present is likely to be at a low or high occupancy. Introducing simulated sequencing errors into the validation script, altered for the known error profiles of each technology type (Ross et al., 2013), would allow an indication of the types and quantities of variation that could be missed by each sequencing technology. This knowledge could then be used to tailor the technology used to the type of outcome required.

To extend TURNIP for the analysis of NGS datasets further considerations would need to be made. Firstly, memory requirements would be considerably more demanding for NGS short read length data as the datasets are much larger, with more reads at a higher depth of sequencing. This could present difficulties if the data are to be held in memory and processed. Input FASTQ files should be converted to an NGS format such as the BAM format (Li et al., 2009). Furthermore, a read mapping rather than a multiple alignment approach, using programs such as BWA or Stampy (Li and Durbin, 2009; Lunter and Goodson, 2011) might be more appropriate for short read datasets. Finally, consideration should be given to implementing a reference-free variation calling approach. As noted in the introduction, a number of software tools that use such an approach are already available, though none of them applicable to repetitive sequence data (Ratan et al., 2010; Iqbal et al., 2012; Dou et al., 2012). Of course, the efficacy of any

reference-free approach could be tested with an updated version of the validation script.

7.2. Analysing Variation

Analysis of the variation uncovered by TURNIP inferred evolutionary relationships between yeast strains consistent with previous research. It also gave insights into interesting or unexpected relationships which could be followed up in future.

S. cerevisiae mosaic genome types were found to have, on average, 4.4 times more pSNPs than structured genome types, agreeing with a previous study (James et al., 2009). Within both datasets pSNP occupancies were found to follow a U-shaped distribution. This U-shaped distribution is predicted by mutation-drift theory, and is seen in datasets of allele frequency (Chakraborty et al., 1980) and gene frequency (Haegeman and Weitz, 2012) within populations. However, and perhaps unexpectedly, clear differences were observed between the U-shaped distributions of the *S. paradoxus* and structured genome type *S. cerevisiae* strains. Within *S. cerevisiae* over half of the observed pSNPs were found to have occupancies between 10 and 90%, whereas in *S. paradoxus* less that 10% of pSNPs had occupancies in that range. These differences in shape were hypothesised to be due to *S. cerevisiae* strains having higher copy numbers and having undergone frequent hybridisation. Furthermore, a number of *S. cerevisiae* strains which were previously assigned to structured groups were re-classified into additional subdivisions (structured-clean and structured-mosaic) to explain the evolutionary histories of these strains.

Phylogenetic trees derived from combined SNP+pSNP rDNA datasets were found to be highly similar to previous whole-genome SNP-based trees for the two yeast species (Liti et al., 2009), with *S. paradoxus* strains splitting clearly into geographical groups. Comparison of NeighborNets to these phylogenetic trees illustrated the existence of conflict within the phylogenetic signal of the *S. cerevisiae* dataset. This is likely to be a consequence of genomic mosaicism that arose from the hybrid origins of the *S. cerevisiae* strains. In contrast, the phylogenetic structure of the *S. paradoxus* strains appeared to be more tree-like.

This analysis also suggests that pSNPs could potentially be used to identify hybridisation signals within genomes. As already noted, mosaic genomes possess more pSNPs than structured ones. The *S. paradoxus* analysis led to the hypothesis

that the strains N-17 and N-45 resulted from hybridation between presently unknown European and Far Eastern strains. N-17 and N-45 were shown to possess low occupancy pSNPs normally associated with the other geographical group. Further analysis on these and closely related strains would be needed to confirm this hypothesis, including checking if the variation could be the result of low level sequence contamination from another strain.

7.3. Simulating rDNA Dynamics

Two Java programs, SIMPLEX and CONCERTINA, were developed to simulate "idealised" versions of mutational processes thought to be involved in concerted evolution. A series of preliminary simulation experiments were devised using a core set of parameters taken from previous studies, enabling some of the patterns observed in earlier rDNA variation analysis to be investigated.

The SIMPLEX program followed the fate of a single pSNP whilst it was spread across or lost from an rDNA array. Varying parameters for simplified USCE and GC events revealed preliminary insights into the dynamics of concerted evolution. USCE was found to more rapidly homogenize an rDNA array than GC, and on average the size of the array was smaller when a pSNP was fixed than when it was lost. A polynomial relationship was identified between the pSNP occupancy at the beginning of a simulation run and the average number of concerted evolutionary events until fixation or loss was achieved. Delving deeper into this latter case, for each pSNP occupancy bin a Poisson distribution for rapidity of fixation and loss was found, a natural distribution for data of this type. The position of the pSNP-possessing unit within the rDNA array was not found to effect the spread of the pSNP, except if it was within the first unit. Furthermore, a positional effect existed whereby greater numbers of events were needed to fix a pSNP if it was located within the first 1,000 bases of an rDNA unit.

The second of the two programs, CONCERTINA, expanded on the processes introduced in SIMPLEX. It allowed a continual process of pSNP birth (point mutation) within an rDNA array balanced against the previous USCE and GC processes of concerted evolution. CONCERTINA also modelled the divergence of strains (rDNA arrays) over a phylogenetic tree. These two enhancements enabled differences in pSNP dynamics to be investigated by varying the balance between point mutation and concerted evolution, both in a single rDNA array

and across sets of rDNA arrays related by a tree-like structure. The shape of pSNP occupancy distributions was found to vary with the underlying parameters. In particular, a deep U-shaped distribution resulted for similar rates of point mutation to concerted evolutionary events.

This preliminary research in the computational simulation of concerted evolutionary processes could be built upon in a number of ways. For example, at present values drawn from (discrete) uniform distributions are used for many of the parameter values (e.g. USCE tract lengths are currently distributed as $U\{1, ..., 7\}$), when a (discretised) Gaussian distribution might provide a better fit to the biological processes involved. Furthermore, in the current model, all units in the rDNA array are equally likely to be chosen to start a USCE or GC event. However, previous research has suggested that the innermost, central units within an rDNA array are more likely to be involved in a concerted evolutionary mutation event (O'Kelly, 2008). Again a Gaussian, rather than a uniform, distribution could be used to choose the units for each event. Other parameters could also be updated, such as the GC tract length. This parameter is currently static but it could also become a variable in future.

In addition to incorporating more natural variation within the SIMPLEX or CONCERTINA parameters, recently discovered features of the concerted evolutionary process could also lead to model changes. For example, large deletion events, where an rDNA array rapidly decreases in copy number, have been discovered experimentally (Ganley and Kobayashi, 2011). Adding such an event to SIMPLEX would likely result in significant changes to the results of the simulation runs, as low rDNA copy number and hence fixation of a pSNP could be achieved much more rapidly. Mathematically, this new event is reminiscent of a particular type of random walk known as a Lévy flight, whereas the current model treats rDNA copy number more similarly to a standard random walk. Lévy flights are often seen in larger scale biological processes, for example in foraging strategies of animals such as albatross and marine predators (Humphries et al., 2010, 2012).

A priority for future CONCERTINA development is the addition of a hybridisation process. This could be incorporated as part of the current tree structure, with different nodes having sections of their rDNA array cross over at a certain rate. Hybridisation would be expected to have a large effect on the results of simulations, as such an event would immediately inject a number of pSNPs at greater than 1% occupancy into an rDNA array. Furthermore, hybridisation has been linked by variation analysis to a shallow U-shaped pSNP occupancy

distribution in *S. cerevisiae* strains. By carrying out simulation runs varying the rates of hybridisation, it could be formally tested whether such variation could distinguish between pSNP occupancy distributions similar to those of *S. cerevisiae* and *S. paradoxus*. To implement hybridisation, a change to the main CONCERTINA data structure to a more biologically representative structure, such as a balanced bifurcating tree, could also be included at this point. This change would allow inferences to be made between different time points on a tree, and to track hybridisation events more clearly.

To more rigorously make inferences from simulations about parameter values acting on real datasets, the software would need to be extended to include methods to measure the goodness-of-fit (for example, sum-of-squares or Chi-squared statistics) between experimentally observed and simulated data points. Indeed, updating the TURNIP software and other scripts for the analysis of NGS data would immediately generate a new raft of experimental datasets that could be used to explore a greater portion of parameter space than is currently possible. This might in turn lead to updating of the core model parameters. The preliminary simulation runs made many assumptions about parameter values based on the current literature. In future, some of these values are likely to change in the light of further research. Ultimately, only by the analysis of real experimental datasets can any meaningful conclusions regarding concerted evolutionary mechanisms be drawn.

Other extensions to CONCERTINA should include simulating concerted evolution in multi-locus systems. Such systems are present in many organisms, and here processes such as gene conversion are thought to have a greater importance than in single-locus systems, as they are required to homogenise the rDNA sub-arrays scattered across the genome. Furthermore, concerted evolutionary processes such as intra-chromatid recombination, resulting in ERCs, and the effect of meiotic recombination could also be simulated. Including other polymorphism types, such as indels, might also refine knowledge of rDNA array evolution. Based upon this variation analysis, such mutations are more likely to occur in specific regions of an rDNA unit, such as homopolymeric tracts found in IGS regions, and this would need to be reflected in a computational model.

7.4. Analysis of rDNA flanking regions

The rDNA flanking regions of one *S. paradoxus* and three *S. cerevisiae* strains were analysed. The broad structure of the left flank was found to be conserved across all four strains, with a partial rDNA unit beginning the array, confirming the previously defined structure for S288c (SGD, 2013). However, the right flank was identical only in the *S. paradoxus* strain and two of the *S. cerevisiae* strains. Given the phylogenetic relationships of these strains, this structure is likely to be the ancestral arrangement of *S. cerevisiae*, and potentially further across the *sensu stricto* group. In the *S. cerevisiae* type strain S288c, a different partial rDNA unit is found, along with tandemly arranged groups of *ASP3* and 5S sequences. It is likely that the insertion of *ASP3* into chromosome XII of S288c (or its ancestor) via horizontal gene transfer from *Wickerhamomyces anomalus* (League et al., 2012) has deleted a section of the rDNA array, giving rise to the different (partial) terminating units. It is known that other yeast strains from laboratory or industrial environments also contain *ASP3*. This member of the asparagine degradation pathway is induced in response to nitrogen starvation, and may have enabled these strains to adapt to artificial environments. It would be interesting in future to examine the structures of right flanking rDNA units in other Asp3p-containing strains, to see whether they are arranged similarly to S288c.

The presence of partial rDNA units at the end of rDNA arrays poses an interesting question. How have such units arisen? Given our current understanding of the way in which the rDNA array evolves, it would seem that partial rDNA units could be acquired in three ways. Firstly, an intact terminal unit could degrade, for example, via deletion. This is certainly a likely event in the case of the right terminal unit of S288c (although it was likely to have been a different partial unit even prior to this event). Secondly, the USCE process may tolerate a certain level of inexact pairing, giving rise to partial units only at the end of rDNA arrays. Thirdly, currently unknown mechanisms of concerted evolution may allow partial rDNA units to be added to the ends of an array. Further research in this area, including sequence analysis of rDNA arrays, will enable evidence to be gathered in support of one or more of these scenarios. Ultimately, this could lead to an update of the CONCERTINA and SIMPLEX models for dealing with terminal rDNA units, which are currently treated no differently from all other rDNA units.

7.5. Conclusion

Significant new knowledge on rDNA variation, structure and evolution has been presented. A range of new software tools for variation discovery, validation, analysis and modelling has been introduced. Together this knowledge and toolkit form a framework for further investigation of this key genomic region and of the concerted evolutionary processes that mould it. Finally, many aspects of further experimentation have been identified, both laboratory- and computer-based, which would be highly interesting to explore further.

Appendices

A. rDNA Variation

Strain	Position	Read	NCBI	SGRP	Other info
A12	3617	A12-1f02.p1k	*Saccharomyces cerevisiae* YJM789 mitochondrion 851/863 (99%)	N_45 chr 13 360/635 (56%)	likely contamination
A4	3622	A4-13m20.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XI 566/638 (89%)	A12 . chr11 671/698 (96%)	wrong chromosome
A4	5929	A4-13n11.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 229/239 (96%)	A12. chr15 863/872 (98%)	wrong chromosome, lookes like small subsection (a couple hundred nt) match rDNA
CBS432	3072	CBS432-171a16.p1k	*Saccharomyces paradoxus* Ty3-like retrotransposon, partial sequence 97/112 (87%)	KPN3829. chr07 182/252 (72%)	only a couple of reads, doesn't match well
CBS432	7547	CBS432-25b09.q1k	*Plasmodium falciparum* 3D7 chromosome 11, complete sequence 1147/1175 (98%)	REF. chr12 264/320 (82%)	Plasmodium contamination, matches poorly to subsection of rDNA only a couple of reads
CBS5829	3601	CBS5829-32m04.p1k	*Saccharomyces cerevisiae* S288c chromosome XVI, complete Sequence, Features in this part of subject sequence: Vma13p 764/845 (90%)	CBS5829 chr16 743/823 (90%)	wrong chromosome

Strain	Position	Read	NCBI	SGRP	Other info
DBVPG 4650	3607	DBVPG4650-27g05.q1k	*Saccharomyces cerevisiae* S288c chromosome II, complete Sequence, hypothetical protein 1044/1196 (87%)	Q95_3. chr02 1123/1227 (91%)	wrong chromosome
DBVPG 4650	4029	DBVPG4650-27a11.p1k	*Saccharomyces cerevisiae* S288c chromosome XV, complete sequence Hypothetical protein 188/261 (72%)	REF. chr13 967/993 (97%)	wrong chromosome, very poor hit in NCBI
DBVPG 6304	3606	DBVPG6304-22m16.p1k	*Saccharomyces cerevisiae* S288c chromosome XVI, complete Sequence, Vma13p, 605/679 (89%)	DBVPG6304. chr16 636/668 (95%)	wrong chromosome
DBVPG 6304	5945	DBVPG6304-41m13.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII, EC1118_1L7 genomic scaffold, whole genome shotgun sequence Sec10p, 532/628 (85%)	A4. chr12 716/718 (99%)	wrong part of chromosome, Sec10 protein
IFO1804	3070	IFO1804-13a18.p1k	*Saccharomyces paradoxus* Ty3-like retrotransposon long terminal repeat, partial sequence 292/372 (78%)	IFO1804. chr07 285/362 (78%)	no good hits. SGRP Gbrowse aligned to Chr 7, no protein coding area
IFO1804	3639	IFO1804-5o03.p1k	*Saccharomyces cerevisiae* S288c chromosome XIV, complete Sequence, hypothetical protein, 696/826 (84%)	N_45. chr16 419/419 (100%)	wrong chromosome

Strain	Position	Read	NCBI	SGRP	Other info
IFO1804	4022	IFO1804-14n20.q1k	*S.cerevisiae* proline-specific permease (PUT4) gene, complete Cds, 745/866 (86%)	IFO1804. chr15 777/862 (90%	wrong chromosome, matches part of PUT4 gene
KPN3828	5951	KPN3828-14o01.q1k	*Saccharomyces cerevisiae* S288c chromosome XII, complete Sequence, Sec10p, 528/628 (84%)	REF. chr12 908/912 (99%)	wrong part of chromosome, Sec10 protein
KPN3828	4470	KPN3828-4j09.q1k	*Saccharomyces cerevisiae* S288c chromosome II, complete, Ubc4p Sequence 776/862 (90%	REF. chr02 769/806 (95%)	wrong chromosome, Ubc4 protein
KPN3828	3605	KPN3828-3e13.q1k	*Saccharomyces cerevisiae* S288c chromosome XVI, complete sequence , Vma13, 803/892 (90%)	KPN3828. chr16 849/890 (95%)	wrong chromosome, Vma13 protein
KPN3829	3629	KPN3829-17n22.q1k	*Saccharomyces douglasii* mitochondrial cytochrome c oxidase subunit I (COXI) gene, complete cds 863/872 (99%)	DBVPG6304. chr15 137/221 (61%)	Possible contamination? Or very poor hit to SGRP, cannot find read in SGRP gbrowse
KPN3829	4139	KPN3829-14d18.q1k	*Saccharomyces cerevisiae* S288c chromosome XVI, complete Isr1 Yth1 genes Sequence 790/898 (88%)	REF. chr16 895/897 (99%)	wrong chromosome, matches ISR1 and YTH1 region
N_17	3067	N_17-11g12.p1k	*Saccharomyces cerevisiae* S288c chromosome VII, complete sequence Tim13476/686 (69%)	CBS432. chr07 874/879 (99%)	wrong chromosome, match to just before Tim13 gene (looking at SGRP gbrowse

Strain	Position	Read	NCBI	SGRP	Other info
N_17	3601	N_17-10b07.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 897/924 (97%)	N_45. chr12 897/905 (99%)	TRUE positive, filtered out because other read was false positive, see below
N_17	3601	N_17-61g16.q1k	*Saccharomyces cerevisiae* EC1118 chromosome VII, 712/902 (79%)	REF. chr07 830/884 (93%)	wrong chromosome
N_43	3086	N_43-15b14.p1k	*Saccharomyces cerevisiae* EC1118 chromosome VII, 418/608 (69%)	N_43. chr07 110/130 (84%)	wrong chromosome, poor match to SGRP, near Tim13
N_43	3616	N_43-21n22.p1k	*Saccharomyces cerevisiae* S288c chromosome VII729/772 (94%)	N_45. chr07 650/671 (96%)	wrong chromosome, Pfk1 gene
N_43	5967	N_43-34f22.q1k	*Saccharomyces cerevisiae* S288c chromosome XI, complete Sequence 468/595 (79%	N_45. chr11 845/854 (98%)	wrong chromosome
N_43	4072	N_43-25n19.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XV Irc23p Tom6p 768/949 (81%)	N_45. chr15 876/908 (96%)	wrong chromosome, near IRC23 and TOM6 genes
N_44	3096	N_44-32h13.q1k	*Saccharomyces cerevisiae* EC1118 chromosome I 307/404 (76%)	IFO1804. chr07 284/362 (78%)	poor matches generally, wrong chromosome, SGRP matches it to chr7
N_44	3616	N_44-12f23.p1k	gb—DQ115391.1— 777/954 (81%)	N_45. chr07 868/919 (94%)	wrong chromosome, possibly some putative non essential genes

Strain	Position	Read	NCBI	SGRP	Other info
N_44	5846	N_44-12c15.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 907/922 (98%)	N_44. chr12 904/922 (98%)	TRUE positive, was filtered into a SNP (as consensus reads were actually from Sec10p)
N_45	3067	N_45-10g02.p1k	*Saccharomyces paradoxus* Ty3-like retrotransposon, partial sequence 230/275 (84%)	N_45. chr07 235/267 (88%)	doesn't match well to anything, but SGRP matches it to chr 7
N_45	3621	N_45-10k20.p1k	*Saccharomyces cerevisiae* EC1118 chromosome VII, 707/860 (82%)	N_45. chr07 836/883 (94%)	wrong chromosome, see N_44 3616 description
N_45	4056	N_45-10d11.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 695/712 (98%)	N_45. chr12 866/975 (88%)	TRUE positive, filtered out because consensus read matched to another chr, but not put as SNP, and read did pass filtering. A change in MUSCLE alignment perhaps?
N_45	8671	N_45-46n03.q1k	*Saccharomyces cerevisiae* S288c chromosome XVI, complete Sequence,Hat1p 525/613 (86%)	N_45. chr16 825/925 (89%)	wrong chromosome, part matches to Hat1
Q32	3601	Q32_3-3b24.p1k	TPA_inf: *Saccharomyces cerevisiae* S288c chromosome IX Syg1p hypothetical protein 782/911 (86%)	KPN3828. chr09 820/883 (92%)	wrong chromosome

Strain	Position	Read	NCBI	SGRP	Other info
Q32	4038	Q32_3-20a12.p1k	*Saccharomyces cerevisiae* S288c chromosome XV, complete sequence Hypothetical protein 188/261 (72%)	REF. chr13 799/819 (97%)	wrong chromosome
Q59	3601	Q59_1-1o22.q1k	*Saccharomyces cerevisiae* S288c chromosome XVI, complete sequence ,Gln1p Vma13p 735/816 (90%	N_17. chr16 701/799 (87%)	wrong chromsome, Vma13 gene again
Q59	4469	Q59_1-9d13.p1k	*Saccharomyces cerevisiae* BIO6 gene for biotin biosynthesis enzyme, partial cds, strain:Sake yeast kyokai No.7, 38 Kb cosmid 450/668 (67%)	Q74_4. chr02 896/914 (98%)	wrong chromosome
Q62	3610	Q62_5-17e20.q1k	*Saccharomyces cerevisiae* S288c chromosome VII, complete sequence Npp2p Edc3p 739/840 (88%	Z1_1. chr05 820/835 (98%	wrong chromosome
Q62	4037	Q62_5-11d05.p1k	*Saccharomyces cerevisiae* S288c chromosome XV hypothetical protein 134/188 (71%	REF. chr13 967/993 (97%) 881/900 (97%)	wrong chromosome, poor blast match
Q89	4040	Q89_8-10h01.p1k	*Saccharomyces cerevisiae* S288c chromosome XII, 120/150 (80%)	Ref. chr12 845/884 (95%)	right chromosome, wrong part
Q89	4479	Q89_8-8n23.q1k	*Saccharomyces cerevisiae* S288c chromosome II, complete Sequence, Ubc4p 743/826 (90%)	REF. chr02 730/761 (95%)	wrong chromosome, Ubc4 protein

Strain	Position	Read	NCBI	SGRP	Other info
Q95	3604	Q95_3-38k03.q1k	*Saccharomyces cerevisiae* EC1118 chromosome VII 586/701 (84%)	Q95_3. chr07 762/808 (94%)	wrong chromosome
Q95	4046	Q95_3-48g10.q1k	*Saccharomyces cerevisiae* S288c chromosome XV 467/571 (82%)	REF. chr15 476/492 (96%)	wrong chromosome and poor match
S36	3608	S36_7-10f16.p1k	*Saccharomyces cerevisiae* EC1118 chromosome I 829/938 (88%)	REF. chr01 905/923 (98%)	wrong chromosome
T21	3607	T21_4-14p12.q1k	*Saccharomyces cerevisiae* EC1118 chromosome VII 739/877 (84%)	Z1_1. chr07 835/882 (94%)	wrong chromosome
T21	5920	T21_4-21k03.p1k	*Saccharomyces cerevisiae* S288c chromosome XII, Sec10p 331/402 (82%)	Ref. chr12 647/649 (99%)	right chromosome, wrong part
UFRJ50791	3608	UFRJ50791-14d07.q1k	*Saccharomyces paradoxus* BUD3p (BUD3) gene, partial cds; YCL012Cp (YCL012C) gene, complete cds; and GBP2p (GBP2) gene, partial 798/831 (96%)	UFRJ50791. chr03 821/836 (98%)	wrong chromosome
UFRJ50791	5225	Non consensus read UFRJ50791-10g03.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, 924/967 (96%)	UFRJ50816. chr12 950/973 (97%)	right chromosome for non consensus reads, but wrong one for consensus, therefore actually a SNP
UFRJ50791	5225	Consensus read UFRJ50791-14h21.q1k	*Saccharomyces cerevisiae* S288c chromosome III Csm1p 813/1023 (79%	A12. chr03 709/774 (91%)	as above

Strain	Position	Read	NCBI	SGRP	Other info
UFRJ50816	3617	UFRJ50816-18e22.p1k	*Saccharomyces cerevisiae* S288c chromosome XIV, complete Sequence, Bni1p 629/761 (83%)	UFRJ50816. chr14 719/765 (93%)	wrong chromosome
UWOPS91_917_3	3601	UWOPS91_917_1-10c10.p1k	*Saccharomyces cerevisiae* YJM789 mitochondrion, complete genome 1020/1040 (98%)	IFO1804. chr11 316/566 (55%)	contamination
UWOPS91_917_3	5948	UWOPS91_917_1-13c09.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence Length=9103 229/239 (96%)	UWOPS91_917_1. chr15 986/995 (99%)	wrong chromosome, but poor match to rDNA in NCBI was top hit
Y6_5	3086	Y6_5-23d22.p1k	*Saccharomyces cerevisiae* S288c chromosome V 186/230 (81%)	Q69_8. chr07 897/900 (99%)	wrong chromosome
Y6_5	4050	Non consensusY6_5-19h05.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 822/826 (99%)	REF. chr12 753/813 (92%)	the only consensus read actually matched wrong chromosome therefore is actually a SNP
Y6_5	4050	consensus Y6_5-8e12.q1k	*Saccharomyces cerevisiae* S288c chromosome XV 596/750 (79%)	REF. chr15 710/764 (92%)	as above
Y7	3602	Y7-1p03.p1k	*Saccharomyces cerevisiae* EC1118 chromosome II 776/854 (91%)	REF. chr02 617/651 (94%)	wrong chromosome

Strain	Position	Read	NCBI	SGRP	Other info
YPS138	2617	YPS138-3o07.p1k	*Saccharomyces cerevisiae* S288c chromosome XII 268/269 (99%)	YPS138. chr12 807/811	poor blast match, but does to rDNA. SGRP Gbrowse matches it to YLR162W-A
YPS138	4072	YPS138-32b24.p1k	*Saccharomyces cerevisiae* RF1095, RF435, and inner membrane protease 1 (PET2858) genes, complete cds 782/886 (88%)	DBVPG6304. chr13 857/899 (95%)	wrong chromosome, matched to IMP1 protein on SGRP?
Z1_1	3600	Z1_1-11f09.p1k	*Saccharomyces cerevisiae* S288c chromosome XVI Vma13p 866/965 (90%	Z1_1. chr16 901/950 (94%)	wrong chromosome, Vma13?
Z1_1	4123	Z1_1-26f23.q1k	*Saccharomyces cerevisiae* S288c chromosome XVI Isr1p Yth1p 761/843 (90%)	REF. chr16 842/844 (99%)	wrong chromosome

Table A.1.: *S. paradoxus* pSNPs lost after filtering, and what they were identified as

Strain	Position	Read	NCBI	SGRP	Other info
A12	3612	A12-29c18.p1k	891/899 (99%)*Saccharomyces cerevisiae* EC1118 chromosome XII	YPS138. chr12 655/911 (71%)	YJM975. chr12 856/896 (95%) in SGRP Sc blast though
A12	5029	A12-10g12.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 1049/1125 (93%)	A4. chr12 1050/1106 (94%)	Likely a genuine pSNP

Strain	Position	Read	NCBI	SGRP	Other info
A4	4068	A4-25e07.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 781/811 (96%)	A4. chr12 764/785 (97%)	only one kept, seems genuine
CBS432	5655	CBS432-10d18.q1k	780/786 (99%)*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence	CBS5829. chr12 773/786 (98%)	possibly SNP as high occupancy? Also the one consensus read matches better to Sc than Spd strains?
CBS5829	4052	CBS5829-10b10.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 878/889 (99%)	CBS5829. chr12 809/895 (90%)	genuine
DBVPG 4650	4050	DBVPG4650-10i17.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 866/874 (99%)	KPN3828. chr12 816/881 (92%)	genuine
DBVPG 6304	4067	DBVPG6304-40j24.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 730/759 (96%)	DBVPG6304. chr12 707/727 (97%)	genuine

Strain	Position	Read	NCBI	SGRP	Other info
DBVPG 6304	3645	DBVPG6304-13c17.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 768/834 (92%)	A4. chr12 408/409 (99%)	probably genuine
IFO1804	N/A	N/A	N/A	N/A	3 changed to SNP, 4050, 4052, 4054
KPN3828	5011	KPN3828-16o24.p1k	*Saccharomyces paradoxus* strain BY20111 35S ribosomal cistron external transcribed spacer, partial sequence; ribosomal DNA intergenic spacer 2, complete sequence; and 5S ribosomal RNA gene, partial sequence951/984 (97%)	CBS5829. chr12 895/910 (98%)	genuine, Also note position 4050 and 4067 changed to SNP from pSNP
KPN3829	6436	KPN3829-7i01.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 911/926 (98%)	KPN3829. chr12 909/924 (98%)	genuine
N_17	3456	N_17-10b07.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 897/924 (97%)	N_45. chr12 897/905 (99%)	genuine

Strain	Position	Read	NCBI	SGRP	Other info
N_17	3818	N_17-10d24.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 621/679 (91%)	N_45. chr12 830/961 (86%)	genuine
N_17	8951	N_17-23n09.p1k	*Saccharomyces cerevisiae* strain CHY1011 18S ribosomal RNA gene, partial sequence; internal transcribed spacer 1, 5.8S ribosomal RNA gene, and internal transcribed spacer 2, complete sequence; and 26S ribosomal RNA gene, partial sequence 690/695 (99%)	REF. chr12 691/694 (99%)	genuine
N_43	4041	N_43-28a05.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 1079/1188 (91%)	N43. chr12 1059/1156 (91%)	genuine
N_44	6510	N_44-10f05.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 923/936 (99%)	N_44. chr12 917/926 (99%)	genuine

Strain	Position	Read	NCBI	SGRP	Other info
N_44	8377	N_44-13k07.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 761/763 (99%)	N_45. chr12 762/763 (99%)	genuine
N_45	3456	N_45-10b02.q1k	*Saccharomyces paradoxus* strain BY20111 5S ribosomal RNA gene, partial sequence; ribosomal DNA intergenic spacer 1, complete sequence; and 25S ribosomal RNA gene, partial sequence 972/1117 (87%)	N_45. chr12 953/1087 (87%)	genuine
N_45	4296	N_45-10f11.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 761/763 (99%)	N_45. chr12 740/765 (96%)	genuine
N_45	5817	N_45-42c24.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 951/970 (98%)	N_45. chr12 949/968 (98%)	genuine
Q32	N/A	N/A	N/A	N/A	N/A

Strain	Position	Read	NCBI	SGRP	Other info
Q59	3558	Q59_1-30h04.q1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, 808/812 (99%)	REF. chr12 818/821 (99%)	genuine
Q59	6104	Q59_1-10a13.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 945/960 (98%)	943/957 (98%)943/957 (98%)	genuine
Q62	3558	Q62_5-18l17.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 789/791 (99%)	REF. chr12 801/801 (100%)	genuine
Q89	N/A	N/A	N/A	N/A	N/A
Q95	N/A	N/A	N/A	N/A	N/A
S36	N/A	N/A	N/A	N/A	N/A
T21	4050	T21_4-1b07.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA 888/898 (99%)	T21_4. chr12 821/898 (91%)	genuine
UFRJ50791	N/A	N/A	N/A	N/A	N/A
UFRJ50816	543	UFRJ50816-15i03.p1k	*Saccharomyces cerevisiae* S288c chromosome XII 723/723 (100%)	N_43. chr12 719/723 (99%)	genuine
UFRJ50816	4837	UFRJ50816-15e09.q1k	S. carlsbergensis rDNA not transcribed spacer (NTS) sequence 832/839 (99%)	N_44. chr12 702/844 (83%)	genuine

Strain	Position	Read	NCBI	SGRP	Other info
UFRJ50816	6076	UFRJ50816-15i03.q1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-ETS1-1 958/960 (99%)	N_44. chr12 842/969 (86%)	genuine
UFRJ50816	9098	UFRJ50816-10e07.q1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-ETS1-1 870/876 (99%)	UFRJ50816. chr12 876/876 (100%)	genuine
UWOPS91_917_3	3493	UWOPS91_917_1-13b03.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 rRNA-RDN37-2 558/568 (98%)	N_45. chr12 481/582 (82%)	genuine
UWOPS91_917_3	9040	UWOPS91_917_1-12n15.p1k	*Saccharomyces cerevisiae* strain CHY1011 18S ribosomal RNA gene, partial sequence; internal transcribed spacer 1, 5.8S ribosomal RNA gene, and internal transcribed spacer 2, complete sequence; and 26S ribosomal RNA gene, partial sequence 872/885 (99%)	A4. chr12 861/886 (97%)	genuine
Y6_5	N/A	N/A	N/A	N/A	N/A
Y7	N/A	N/A	N/A	N/A	N/A
YPS138	N/A	N/A	N/A	N/A	most pSNPs changed into SNPs
Z1_1	N/A	N/A	N/A	N/A	N/A

Table A.2.: *S. paradoxus* pSNPs kept after filtering, and what they were identified as

Strain	Position	Read	NCBI	SGRP	Other info
273614N	1897	273614N-10k10.q1k	*Saccharomyces cerevisiae* EC1118 chromosome X Vps55p 991/1024 (97%)	273614N.chr10 952/994 (95%)	wrong chromsome
273614N	4461	273614N-27p17.p1k	*Saccharomyces cerevisiae* S288c chromosome IV Tmn2p 932/939 (99%)	YPS606.chr04 891/938 (94%)	wrong chromsome
322134S	6089	322134S-2n11.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-RDN37-1 rRNA-RDN18-1 1153/1244 (93%)	NCYC110.chr12 1048/1111 (94%)	both reads were for same pSNP. However, only one is a true pSNP, therefore when false one was lost, it failed threshold of >1 variant position
322134S	6089	322134S-4d03.p1k	*Saccharomyces cerevisiae* S288c chromosome XIII hypothetical protein 572/574 (99%	RM11_1A.chr13 593/615 (96%)	both reads were for same pSNP. However, only one is a true pSNP, therefore when false one was lost, it failed threshold of >1 variant position
378604X	3614	378604X-13g08.q1k	*Saccharomyces cerevisiae* EC1118 chromosome IV Tsc13p Nop1p 1054/1086 (97%)	DBVPG1106.chr04 1040/1117 (93%)	wrong chromsome
BC187	3615	BC187-22k24.p1k	*Saccharomyces cerevisiae* EC1118 chromosome VII 891/891 (100%)	DBVPG1373.chr07 870/891 (97%)	wrong chromsome
DBVPG1106	none				
DBVPG137	4483	DBVPG1373-21d08.q1k	*Saccharomyces cerevisiae* EC1118 chromosome VII Rps2p Nab2p 813/814 (99%)		wrong chromsome

Strain	Position	Read	NCBI	SGRP	Other info
DBVPG1373	4995	DBVPG1373-25d13.p1k	*Saccharomyces cerevisiae* EC1118 chromosome VI 1135/1172 (97%)		wrong chromsome
DBVPG178	3601	DBVPG1788-15p24.q1k	*Saccharomyces cerevisiae* S288c chromosome V, complete sequence Tca17p hypothetical protein 897/932 (96%)		wrong chromsome
DBVPG1788	4930	DBVPG1788-20f14.p1k	*Saccharomyces cerevisiae* EC1118 chromosome VI210/213 (99%)		wrong chromsome
DBVPG185	none, in fact some gained				
DBVPG6040	3580	DBVPG6040-19i15.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XRnr2p Rrn7p 829/832 (99%)		wrong chromsome
DBVPG604	3614	DBVPG6040-13b22.p1k	*Saccharomyces cerevisiae* EC1118 chromosome IV Tsc13p 825/853 (97%)		wrong chromsome
DBVPG6044	1895	DBVPG6044-29f13.p1k	*Saccharomyces cerevisiae* EC1118 chromosome X Vps55p 856/884 (97%)		wrong chromsome
DBVPG604	4484	DBVPG6044-33n10.p1k	*Saccharomyces cerevisiae* S288c chromosome IV, complete sequence Tmn2p 674/680 (99%)		wrong chromsome

Strain	Position	Read	NCBI	SGRP	Other info
DBVPG6044	3607	DBVPG6044-30j11.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XIV Nrk1p Tep1p 892/902 (99%)		wrong chromsome
DBVPG676	3580	DBVPG6765-34i05.q1k	*Saccharomyces cerevisiae* EC1118 chromosome Xrnr2p EC1118_1J11_2322p 1118/1155 (97%)		wrong chromsome
DBVPG6765	4478	DBVPG6765-24m23.p1k	*Saccharomyces cerevisiae* EC1118 chromosome VII Nab2p 802/803 (99%)		wrong chromsome
DBVPG6765	7468	DBVPG6765-21l03.q1k	*Saccharomyces cerevisiae* EC1118 chromosome I 941/952 (99%)		wrong chromsome
K11	1895	K11-19c02.q1k	*Saccharomyces cerevisiae* S288c chromosome X Vps55p 900/912 (99%)		wrong chromsome
K11	3607	K11-13l04.p1k	*Saccharomyces cerevisiae* EC1118 chromosome IV Tsc13p Nop1p 906/916 (99%)		wrong chromsome
K11	4317	Non consensus readK11-10n20.q1k	*Saccharomyces cerevisiae* strain BY2986 5S ribosomal RNA gene, partial sequence; ribosomal DNA intergenic spacer 1, complete sequence; and 25S ribosomal RNA gene, partial sequence 935/947 (99%)		genuine, but turned into a SNP in the filtered as consensus reads actually match something else

Strain	Position	Read	NCBI	SGRP	Other info
K11	4317	Consensus read K11-7j06.p1k	*Saccharomyces cerevisiae* S288c chromosome XIII Tub1p 970/980 (99%)		genuine, but turned into a SNP in the filtered as consensus reads actually match something else
L_1374	4930	L_1374-3h19.p1k	*Saccharomyces cerevisiae* EC1118 chromosome VI 851/852 (99%)		wrong chromsome
NCYC110	6563	NCYC110-12d18.p1k	*Saccharomyces cerevisiae* S288c chromosome XIII, complete sequence Msn2p 937/950 (99%)		wrong chromsome
NCYC110	3609	NCYC110-12k05.q1k	*Saccharomyces cerevisiae* S288c chromosome IV, complete sequence Knh1p 882/891 (99%)		wrong chromsome
NCYC361	3615	NCYC361-23n19.p1k	*Saccharomyces cerevisiae* YJM789 mitochondrion, 983/1034 (95%)		mitochondrial DNA?
NCYC361	3603	NCYC361-16j12.p1k	*Saccharomyces cerevisiae* YJM789 mitochondrion 922/946 (97%)		mitochondrial DNA?
S288c	3603	S288c-27n23.q1k	*Saccharomyces cerevisiae* complete mitochondrial genome 892/898 (99%)		mitochondrial DNA?
SK1	1895	SK1-59n20.q1k	*Saccharomyces cerevisiae* S288c chromosome X Vps55p 828/829 (99%)		wrong chromsome
SK1	4951	SK1-5d05.p1k	Synthetic construct clone Semi-SynVIL 1145/1203 (95%)		wrong, second hit is Sc chr 6

Strain	Position	Read	NCBI	SGRP	Other info
SK1	3615	SK1-33p10.p1k	*Saccharomyces cerevisiae* S288c chromosome V Tca17p hypothetical protein 702/711 (99%)		wrong chromsome
UWOPS03_461_4	4465	UWOPS03_461_4-15h14.q1k	*Saccharomyces cerevisiae* EC1118 chromosome VII Rps2p Nab2p 892/902 (99%)		wrong chromsome
UWOPS03_461_4	4925	UWOPS03_461_4-10j20.q1k	*S. carlsbergensis* rDNA not transcribed spacer (NTS) sequence 854/866 (99%)		Consensus read, wrong, second hit is Sc chr 6, is changed to a SNP in filtered version
UWOPS03_461_4	4925	consensus UWOPS03_461_4-4n08.q1k	Synthetic construct clone Semi-SynVI 941/973 (97%)		Consensus read, wrong, second hit is Sc chr 6, is changed to a SNP in filtered version
UWOPS05_217_3	3600	UWOPS05_217_3-11j14.q1k	*Saccharomyces cerevisiae* S288c chromosome IV Tsc13p Nop1p 915/923 (99%)		wrong chromsome
UWOPS05_227_2	none				
UWOPS83_787_3	3610	UWOPS83_787_3-15m18.p1k	*Saccharomyces cerevisiae* EC1118 chromosome II EC1118_1B15_4181p Thi2p 879/911 (96%)		wrong chromsome
UWOPS83_787_3	3795	UWOPS83_787_3-1b09.p1k	*Saccharomyces paradoxus* NRRL Y-17217 genes for 25S rRNA, 5S rRNA, 18S rRNA, 5.8S rRNA, complete sequence 752/854 (88%)		right area, but matches much better to paradoxus. Contamination? Actually, if go to SGRP gbrowse it matches chr 15

Strain	Position	Read	NCBI	SGRP	Other info
UWOPS87_2421	3609	UWOPS87_2421-18i19.q1k	*Saccharomyces cerevisiae* S288c chromosome I, complete sequence 951/983 (97%)		wrong chromsome
W303	3233	W303-2c24.p1k	*Saccharomyces cerevisiae* S288c chromosome XII hypothetical protein Rrt15p 895/901 (99%)		wrong part of chromsome
W303	6567	W303-15f02.q1k	*Saccharomyces cerevisiae* S288c chromosome XIII Msn2p 897/899 (99%)		wrong chromsome
W303	4523	W303-11g03.p1k	*Saccharomyces cerevisiae* S288c chromosome XII hypothetical protein rRNA-RDN5-3 881/892 (99%)		right area, but matches the 5s repeats that are outside the rDNA array
Y9	3639	Y9-20j17.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XIV Nrk1p Tep1p 971/984 (99%)		wrong chromsome
Y12	3593	Y12-1i01.p1k	*Saccharomyces cerevisiae* S288c chromosome XI 964/976 (99%)		wrong chromsome
Y55	4951	Y55-57b02.q1k	Synthetic construct clone Semi-SynVIL 700/708 (99%)		wrong chromsome for second hit
Y55	6565	Y55-1f08.p1k	*Saccharomyces cerevisiae* S288c chromosome XIII 785/791 (99%)		wrong chromsome

Strain	Position	Read	NCBI	SGRP	Other info
YIIc17_E5	4484	YIIc17_E5-2k07.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XI Lap4p 1019/1036 (98%)		wrong chromsome
YIIc17_E5	6564	YIIc17_E5-7h21.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XIII Msn2p 1072/1089 (98%)		wrong chromsome
YJM975	4482	YJM975-14n17.q1k	*Saccharomyces cerevisiae* S288c chromosome XV Vma4p Mrs2p 838/848 (99%)		wrong chromsome
YJM975	4484	YJM975-20f06.q1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA, complete sequence 989/1038 (95%)		one is right, but the other is from wrong chromosome, therefore is lost as only 1 read covers polymorphism
YJM975	4484	YJM975-20f06.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XI Lap4p 871/882 (99%)		one is right, but the other is from wrong chromosome, therefore is lost as only 1 read covers polymorphism
YJM978	6560	YJM978-2c22.q1k	*Saccharomyces cerevisiae* S288c chromosome XIII Msn2p 875/880 (99%)		wrong chromsome
YJM978	4484	YJM978-13i01.p1k	*Saccharomyces cerevisiae* S288c chromosome IV Tmn2p 988/1012 (98%)		wrong chromsome

Strain	Position	Read	NCBI	SGRP	Other info
YJM981	4485	YJM981-16e20.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XI Gfa1p Lap4p 908/929 (98%)		wrong chromsome
YJM981	4497	YJM981-7g08.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XI Gfa1p Lap4p 934/944 (99%)		wrong chromsome
YPS128	4482	YPS128-10m02.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XIV Rpc19p Dbp2p 800/808 (99%)		wrong chromsome
YPS128	4503	YPS128-2j22.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XIV EC1118_1N9_2465p Rpc19p 782/787 (99%)		wrong chromsome
YPS606	4991	consensus YPS606-35g15.q1k	Synthetic construct clone Semi-SynVIL,896/901 (99%)		second hit is to chr 6, this was the only consensus read therefore becomes a SNP in rerun
YS4	7468	YS4-12e16.p1k	*Saccharomyces cerevisiae* EC1118 chromosome I 1114/1168 (95%)		wrong chromsome
YS4	3601	YS4-10b22.q1k	*Saccharomyces cerevisiae* S288c chromosome I Saw1p Drs2p 929/952 (98%)		wrong chromsome
YS9	none				

Table A.3.: *S. cerevisiae* pSNPs lost after filtering, and what they were identified as

Strain	Position	Read	NCBI	SGRP	Other info

Strain	Position	Read	NCBI	SGRP	Other info
273614N	7322	273614N-27p06.q1k	*Saccharomyces cerevisiae* strain CICC1308 18S ribosomal RNA gene 964/966 (99%)	YIIc17_E5. chr12 966/970 (99%)	genuine
273614N	4763	273614N-10g02.p1k	*S. carlsbergensis* rDNA not transcribed spacer (NTS) sequence 981/1010 (97%)	YPS128. chr12 956/984 (97%)	genuine
273614N	3096	273614N-30c14.p1k	*Saccharomyces cerevisiae* S288c chromosome XI rRNA-RDN37-1 rRNA-ETS2-1 1044/1054 (99%)	YS2. chr12 1041/1052 (98%)	genuine
322134S	4664	322134S-10d18.p1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal 1117/1152 (97%	YPS128. chr12 1048/1101 (95%)	genuine
322134S	3154	322134S-19i22.p1k	*Saccharomyces cerevisiae* S288c chromosome XII 915/926 (99%)	YJM981. chr12 913/924 (98%),	genuine
378604X	3902	378604X-10d17.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS2-1 RRNA-RDN5-1 1084/1160 (93%)	378604X. chr12 1036/1159 (89%)	genuine
378604X	4166	378604X-10d05.q1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal 909/932 (98%)	378604X. chr12 875/929 (94%)	genuine

Strain	Position	Read	NCBI	SGRP	Other info
BC187	3871	BC187-22a03.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS2-1 RRNA-RDN5-1 866/872 (99%)	DBVPG6765. chr12 830/870 (95%)	genuine
BC187	5457	BC187-25i17.p1k	*Saccharomyces cerevisiae* partial 5S rRNA gene, NTS2 and ETS1641/644 (99%)		genuine
DBVPG110	3659	DBVPG1106-10m13.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 rRNA-RDN37-2 1055/1099 (96%)		genuine
DBVPG1373	1426	DBVPG1373-13c12.q1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-RDN25-1 887/890 (99%)		genuine
DBVPG137	8462	DBVPG1373-26g06.q1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-ITS2-1 868/869 (99%)		genuine
DBVPG1788	none				
DBVPG185	3612	DBVPG1853-11a02.p1k	*Saccharomyces cerevisiae* strain BY21391 5S ribosomal RNA gene, partial sequence; ribosomal DNA intergenic spacer 1, complete sequence; and 25S ribosomal RNA gene, partial sequence 995/1035 (96%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
DBVPG1853	1887	DBVPG1853-10b20.q1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-ITS2-1 848/849 (99%)		genuine
DBVPG185	9067	DBVPG1853-10p07.q1k	Uncultured *Ascomycota* clone asc07069 5.8S ribosomal RNA gene, partial sequence; internal transcribed spacer 2, complete sequence; and 28S ribosomal RNA gene, partial sequence 577/579 (99%)		genuine
DBVPG6040	524	DBVPG6040-11a08.p1k	*Saccharomyces cerevisiae* S288c chromosome XII, complete sequence rRNA-RDN37-1 rRNA-RDN25-1 781/796 (98%)		genuine
DBVPG604	3697	DBVPG6040-11i14.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 rRNA-RDN37-2 787/827 (95%)		genuine
DBVPG6040	5064	DBVPG6040-10b15.p1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA 973/1026 (95%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
DBVPG604	8738	DBVPG6040-10c15.p1k	*Saccharomyces cerevisiae* strain CHY1011 18S ribosomal RNA gene, partial sequence; internal transcribed spacer 1, 5.8S ribosomal RNA gene, and internal transcribed spacer 2, complete sequence; and 26S ribosomal RNA gene, partial sequence 778/803 (97%)		genuine
DBVPG6044	679	DBVPG6044-13h01.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 RRNA-RDN25-1 936/946 (99%)		genuine
DBVPG604	5524	DBVPG6044-13a08.p1k	*Saccharomyces cerevisiae* partial 5S rRNA gene, NTS2 and ETS1, strain HD4 838/847 (99%)		genuine
DBVPG6765	1852	DBVPG6765-21e09.p1k	*Saccharomyces cerevisiae* S288c chromosome XII, complete sequence rRNA-RDN37-1 rRNA-RDN25-1 887/893 (99%)		genuine
DBVPG676	3012	DBVPG6765-12c03.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-RDN25-1 792/794 (99%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
DBVPG6765	4652	DBVPG6765-10i10.p1k	*S. carlsbergensis* rDNA not transcribed spacer (NTS) sequence 829/834 (99%)		genuine
K11	4484	K11-12m21.p1k	*S. carlsbergensis* rDNA not transcribed spacer (NTS) sequence 918/950 (97%)		genuine
K11	8686	K11-10j23.q1k	*Saccharomyces cerevisiae* S288c chromosome XII, complete sequence rRNA-RDN37-1 rRNA-RDN25-1 1034/1091 (95%)		genuine
L_1374	4484	L_1374-12h10.q1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA 878/898 (98%)		genuine
L_1374	4657	L_1374-12j16.p1k	*S. carlsbergensis* rDNA not transcribed spacer (NTS) sequence 863/872 (99%)		genuine
NCYC110	253	NCYC110-10b20.q1k	*Saccharomyces cerevisiae* S288c chromosome XII, complete sequence rRNA-RDN37-1 rRNA-RDN25-1 595/647 (92%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
NCYC110	8686	NCYC110-10b16.p1k	*Saccharomyces cerevisiae* strain Z614 internal transcribed spacer 1, partial sequence; 5.8S ribosomal RNA gene and internal transcribed spacer 2, complete sequence; and 26S ribosomal RNA gene, partial sequence 890/899 (99%)		genuine
NCYC361	3590	NCYC361-13j14.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII, EC1118_1L10 genomic Scaffold rRNA-NTS1-2 rRNA-RDN37-2 942/956 (99%)		genuine
NCYC361	4854	NCYC361-22p04.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-NTS2-1 rRNA-RDN5-1 614/626 (98%)		genuine
NCYC361	524	NCYC361-33b08.p1k	*Saccharomyces cerevisiae* S288c chromosome XII, complete sequence rRNA-RDN37-1 rRNA-RDN25-1 1082/1121 (97%)		genuine
S288c	4307	S288c-1d19.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS2-1 RRNA-RDN5-1 1091/1156 (94%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
S288c	6089	S288c-18c09.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-RDN37-1 RRNA-RDN18-1 828/831 (99%)		genuine
SK1	3177	SK1-16p22.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 rRNA-RDN37-2 865/875 (99%)		genuine
SK1	8568	SK1-10h11.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-ITS2-1 776/782 (99%)		genuine
UWOPS03_4	5526	UWOPS03_4_4-11b06.p1k	*Saccharomyces cerevisiae* partial 5S rRNA gene, NTS2 and ETS1, 989/1008 (98%)		genuine
UWOPS05_217_3	3517	UWOPS05_217_3-10c23.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 rRNA-RDN37-2 1049/1092 (96%)		genuine
UWOPS05_217_3	5131	UWOPS05_217_3-10a18.p1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal 905/930 (97%)		genuine
UWOPS05_227_2	3517	UWOPS05_227_2-10b15.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS2-1 RRNA-RDN5-1 1050/1131 (93%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
UWOPS05 _227_2	4854	UWOPS05 _227_2-10a14.q1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA 1038/1106 (94%)		genuine
UWOPS83 _787_3	3517	UWOPS83 _787_3-15p17.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 rRNA-RDN37-2 885/902 (98%)		genuine
UWOPS83 _787_3	4270	UWOPS83 _787_3-10l08.p1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA 889/917 (97%)		genuine
UWOPS83 _787_3	5818	UWOPS83 _787_3-18i11.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-ETS1-1 427/432 (99%)		genuine
UW87 _2421	1112	UWOPS87 _2421-3o08.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-RDN25-1 1006/1014 (99%)		genuine
UW87 _2421	3517	UWOPS87 _2421-12k17.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 rRNA-RDN37-2 847/870 (97%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
W303	4431	W303-12k13.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-NTS2-1 rRNA-RDN5-1 1032/1045 (99%)		genuine
W303	5601	W303-16b13.p1k	*Saccharomyces cerevisiae* partial 5S rRNA gene, NTS2 and ETS1, strain HD4 941/950 (99%)		genuine
Y9	3538	Y9-10b11.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII, EC1118_1L10 genomic Scaffold rRNA-NTS1-2 rRNA-RDN37-2 1048/1083 (97%)		genuine
Y9	8295	Y9-11b13.q1k	*Saccharomyces cerevisiae* S288c chromosome XIIrRNA-RDN37-1 rRNA-ITS2-1 878/882 (99%)		genuine
Y12	8295	Y12-10d12.q1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-ITS1-1 916/932 (98%)		genuine
Y12	4484	Y12-14i23.p1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA 868/905 (96%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
Y55	8568	Y55-10b14.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 RRNA-ITS2-1 890/903 (99%)		genuine
Y55	3177	Y55-1h09.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 rRNA-RDN37-2 779/792 (98%)		genuine
YIIc17_E5	3612	YIIc17_E5-4f08.q1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-ETS2-1 883/891 (99%)		genuine
YIIc17_E5	8738	YIIc17_E5-12g04.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-RDN25-1 970/998 (97%)		genuine
YJM975	4484	YJM975-13f12.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS2-1 rRNA-RDN5-1 1279/1379 (93%)		genuine
YJM975	3659	YJM975-11b01.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 RRNA-RDN37-2 960/967 (99%)		genuine
YJM975	6595	YJM975-19f06.p1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-RDN37-1 rRNA-RDN18-1 945/963 (98%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
YJM978	3659	YJM978-10d16.q1k	*Saccharomyces cerevisiae* EC1118 chromosome XII rRNA-NTS1-2 rRNA-RDN37-2 1014/1025 (99%)		genuine
YJM978	9026	YJM978-15h13.q1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-RDN25-1 1015/1039 (98%)		genuine
YJM978	5554	YJM978-13b08.p1k	*Saccharomyces cerevisiae* partial 5S rRNA gene, NTS2 and ETS1, strain HD4 868/875 (99%)		genuine
YJM981	2017	YJM981-14b16.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-RDN25-1 891/896 (99%)		genuine
YJM981	5554	YJM981-10m13.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-NTS2-1 rRNA-RDN5-1 1059/1095 (97%)		genuine
YPS128	none				
YPS606	2996	YPS606-15k22.q1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-RDN25-1 873/875 (99%)		genuine
YPS606	5473	YPS606-15j03.q1k	*Saccharomyces cerevisiae* partial 5S rRNA gene, NTS2 and ETS1, strain HD4 833/840 (99%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
YS4	3430	YS4-17a07.p1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA 1149/1204 (95%)		genuine
YS4	4070	YS4-12d23.q1k	*Saccharomyces cerevisiae* strain BY21391 5S ribosomal RNA gene, partial sequence; ribosomal DNA intergenic spacer 1, complete sequence; and 25S ribosomal RNA gene, partial sequence 833/848 (98%)		genuine
YS4	8702	YS4-10g18.p1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA 893/903 (99%)		genuine
YS9	1813	YS9-13c06.p1k	*Saccharomyces cerevisiae* S288c chromosome XII rRNA-RDN37-1 rRNA-RDN25-1 908/911 (99%)		genuine

Strain	Position	Read	NCBI	SGRP	Other info
YS9	3989	YS9-14g13.p1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA 1126/1201 (94%)		genuine
YS9	5329	YS9-11m17.p1k	*Saccharomyces cerevisiae* strain BY2986 35S ribosomal cistron external transcribed spacer, partial sequence; ribosomal DNA intergenic spacer 2, complete sequence; and 5S ribosomal RNA Gene 907/951 (95%)		genuine
YS9	8702	YS9-12g21.p1k	Synthetic construct clone pNOY373 35S ribosomal RNA, 18S ribosomal RNA, 5.8S ribosomal RNA, 25S ribosomal RNA, and 5S ribosomal RNA 930/948 (98%)		genuine

Table A.4.: *S. cerevisiae* pSNPs kept after filtering, and what they were identified as

Bibliography

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*, 25(17):3389–402.

Arner, E., Tammi, M. T., Tran, A.-N., Kindlund, E., and Andersson, B. (2006). DNPTrapper: an assembly editing tool for finishing and analysis of complex repeat regions. *BMC Bioinformatics*, 7:155.

Ban, N., Nissen, P., Hansen, J., Moore, P. B., and Steitz, T. A. (2000). The complete atomic structure of the large ribosomal subunit at 2.4 A resolution. *Science*, 289(5481):905–20.

Ben Ali, A., Wuyts, J., De Wachter, R., Meyer, A., and Van de Peer, Y. (1999). Construction of a variability map for eukaryotic large subunit ribosomal RNA. *Nucleic Acids Res*, 27(14):2825–31.

Ben-Shem, A., Garreau de Loubresse, N., Melnikov, S., Jenner, L., Yusupova, G., and Yusupov, M. (2011). The structure of the eukaryotic ribosome at 3.0 Ã resolution. *Science*, 334(6062):1524–9.

Bremer, H. (1975). Parameters affecting the rate of synthesis of ribosomes and RNA polymerase in bacteria. *J Theor Biol*, 53(1):115–24.

Brockman, W., Alvarez, P., Young, S., Garber, M., Giannoukos, G., Lee, W. L., Russ, C., Lander, E. S., Nusbaum, C., and Jaffe, D. B. (2008). Quality scores and SNP detection in sequencing-by-synthesis systems. *Genome Res*, 18(5):763–70.

Brown, D. D., Wensink, P. C., and Jordan, E. (1972). A comparison of the ribosomal DNA's of *Xenopus laevis* and *Xenopus mulleri*: the evolution of tandem genes. *J Mol Biol*, 63(1):57–73.

Bryant, D. and Moulton, V. (2004). Neighbor-net: an agglomerative method for the construction of phylogenetic networks. *Mol Biol Evol*, 21(2):255–65.

Buckler, E. S., Ippolito, A., and Holtsford, T. P. (1997). The evolution of ribosomal DNA: divergent paralogues and phylogenetic implications. *Genetics*, 145(3):821–32.

Caporaso, J. G., Kuczynski, J., Stombaugh, J., Bittinger, K., Bushman, F. D., Costello, E. K., Fierer, N., Pena, A. G., Goodrich, J. K., Gordon, J. I., Huttley, G. A., Kelley, S. T., Knights, D., Koenig, J. E., Ley, R. E., Lozupone, C. A., McDonald, D., Muegge, B. D., Pirrung, M., Reeder, J., Sevinsky, J. R., Turnbaugh, P. J., Walters, W. A., Widmann, J., Yatsunenko, T., Zaneveld, J., and Knight, R. (2010). QIIME allows analysis of high-throughput community sequencing data. *Nat Methods*, 7(5):335–6.

Casper, A. M., Mieczkowski, P. A., Gawel, M., and Petes, T. D. (2008). Low levels of DNA polymerase alpha induce mitotic and meiotic instability in the ribosomal DNA gene cluster of *Saccharomyces cerevisiae*. *PLoS Genet*, 4(6):e1000105.

Cavalli-Sforza, L. L. and Edwards, A. W. (1967). Phylogenetic analysis. Models and estimation procedures. *Am J Hum Genet*, 19(3 Pt 1):233–57.

Chakraborty, R., Fuerst, P. A., and Nei, M. (1980). Statistical Studies on Protein Polymorphism in Natural Populations. III. Distribution of Allele Frequencies and the Number of Alleles per Locus. *Genetics*, 94(4):1039–63.

Cole, J. R., Wang, Q., Cardenas, E., Fish, J., Chai, B., Farris, R. J., Kulam-Syed-Mohideen, A. S., McGarrell, D. M., Marsh, T., Garrity, G. M., and Tiedje, J. M. (2009). The Ribosomal Database Project: improved alignments and new tools for rRNA analysis. *Nucleic Acids Res*, 37(Database issue):D141–5.

Crosby, L. D. and Criddle, C. S. (2003). Understanding bias in microbial community analysis techniques due to rrn operon copy number heterogeneity. *Biotechniques*, 34(4):790–4, 796, 798 passim.

Dammann, R., Lucchini, R., Koller, T., and Sogo, J. M. (1993). Chromatin structures and transcription of rDNA in yeast *Saccharomyces cerevisiae*. *Nucleic Acids Res*, 21(10):2331–8.

Davey, R. P., James, S. A., Dicks, J., and Roberts, I. N. (2010). TURNIP: tracking unresolved nucleotide polymorphisms in large hard-to-assemble regions of repetitive DNA sequence. *Bioinformatics*, 26(22):2908–9.

DeSantis, T. Z., Hugenholtz, P., Larsen, N., Rojas, M., Brodie, E. L., Keller, K., Huber, T., Dalevi, D., Hu, P., and Andersen, G. L. (2006). Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Appl Environ Microbiol*, 72(7):5069–72.

Dou, J., Zhao, X., Fu, X., Jiao, W., Wang, N., Zhang, L., Hu, X., Wang, S., and Bao, Z. (2012). Reference-free SNP calling: improved accuracy by preventing incorrect calls from repetitive genomic regions. *Biol Direct*, 7:17.

Duan, Z., Andronescu, M., Schutz, K., McIlwain, S., Kim, Y. J., Lee, C., Shendure, J., Fields, S., Blau, C. A., and Noble, W. S. (2010). A three-dimensional model of the yeast genome. *Nature*.

Dvorák, J., Jue, D., and Lassner, M. (1987). Homogenization of tandemly repeated nucleotide sequences by distance-dependent nucleotide sequence conversion. *Genetics*, 116(3):487–98.

Edgar, R. C. (2004). MUSCLE: a multiple sequence alignment method with reduced time and space complexity. *BMC Bioinformatics*, 5:113.

Ehrenberg, M. (2009). Structure and function of the Ribosome: Scientific Background on the Nobel Prize in Chemistry 2009. Nobel Media AB 2013. Web. 7 Jan 2014. http://www.nobelprize.org/nobel_prizes/chemistry/laureates/2009/advanced.html.

Eickbush, T. H. and Eickbush, D. G. (2007). Finely orchestrated movements: evolution of the ribosomal RNA genes. *Genetics*, 175(2):477–85.

Elemento, O., Gascuel, O., and Lefranc, M.-P. (2002). Reconstructing the duplication history of tandemly repeated genes. *Mol Biol Evol*, 19(3):278–88.

Falcón, A. A., Chen, S., Wood, M. S., and Aris, J. P. (2010). Acetyl-coenzyme A synthetase 2 is a nuclear protein required for replicative longevity in *Saccharomyces cerevisiae*. *Molecular and Cellular Biochemistry*, 333(1-2):99–108.

Fell, J. W., Boekhout, T., Fonseca, A., Scorzetti, G., and Statzell-Tallman, A. (2000). Biodiversity and systematics of basidiomycetous yeasts as determined by

large-subunit rDNA D1/D2 domain sequence analysis. *Int J Syst Evol Microbiol*, 50 Pt 3:1351–71.

Felsenstein, J. (1985). Confidence limits on phylogenies: an approach using the bootstrap. *Evolution*, pages 783–791.

Felsenstein, J. (2004). PHYLIP (Phylogeny Inference Package) version 3.6. Distributed by the author at http://evolution.genetics.washington.edu/phylip. Department of Genome Sciences, University of Washington, Seattle.

Fogel, G., Collins, C., Li, J., and Brunk, C. (1999). Prokaryotic Genome Size and SSU rDNA Copy Number: Estimation of Microbial Relative Abundance from a Mixed Population. *Microb Ecol*, 38(2):93–113.

French, S. L., Osheim, Y. N., Cioci, F., Nomura, M., and Beyer, A. L. (2003). In exponentially growing *Saccharomyces cerevisiae* cells, rRNA synthesis is determined by the summed RNA polymerase I loading rate rather than by the number of active genes. *Mol Cell Biol*, 23(5):1558–68.

Gangloff, S., Zou, H., and Rothstein, R. (1996). Gene conversion plays the major role in controlling the stability of large tandem repeats in yeast. *EMBO J*, 15(7):1715–25.

Ganley, A. R. D., Ide, S., Saka, K., and Kobayashi, T. (2009). The effect of replication initiation on gene amplification in the rDNA and its relationship to aging. *Mol Cell*, 35(5):683–93.

Ganley, A. R. D. and Kobayashi, T. (2007). Highly efficient concerted evolution in the ribosomal DNA repeats: total rDNA repeat variation revealed by whole-genome shotgun sequence data. *Genome Res*, 17(2):184–91.

Ganley, A. R. D. and Kobayashi, T. (2011). Monitoring the Rate and Dynamics of Concerted Evolution in the Ribosomal DNA Repeats of *Saccharomyces cerevisiae* Using Experimental Evolution. *Mol Biol Evol*, 28(10):2883–2891.

Goffeau, A., Barrell, B. G., Bussey, H., Davis, R. W., Dujon, B., Feldmann, H., Galibert, F., Hoheisel, J. D., Jacq, C., Johnston, M., Louis, E. J., Mewes, H. W., Murakami, Y., Philippsen, P., Tettelin, H., and Oliver, S. G. (1996). Life with 6000 genes. *Science*, 274(5287):546, 563–7.

Haegeman, B. and Weitz, J. S. (2012). A neutral theory of genome evolution and the frequency distribution of genes. *BMC Genomics*, 13:196.

Harms, J., Schluenzen, F., Zarivach, R., Bashan, A., Gat, S., Agmon, I., Bartels, H., Franceschi, F., and Yonath, A. (2001). High resolution structure of the large ribosomal subunit from a mesophilic eubacterium. *Cell*, 107(5):679–88.

Henderson, A. S., Warburton, D., and Atwood, K. C. (1972). Location of ribosomal DNA in the human chromosome complement. *Proc Natl Acad Sci U S A*, 69(11):3394–8.

Hillier, L., Riles, L., Albermann, K., André, B., Ansorge, W., Benes, V., Brückner, M., Delius, H., Dubois, E., Düsterhöft, A., Entian, K. D., Floeth, M., Goffeau, A., Hebling, U., Heumann, K., Heuss-Neitzel, D., Hilbert, H., Hilger, F., Kleine, K., Kötter, P., Louis, E. J., Messenguy, F., Mewes, H. W., Hoheisel, J. D., and M Johnston (1997). The nucleotide sequence of *Saccharomyces cerevisiae* chromosome XII. *Nature*, 387(6632 Suppl):87–90.

Hillis, D. M. and Dixon, M. T. (1991). Ribosomal DNA: molecular evolution and phylogenetic inference. *Q Rev Biol*, 66(4):411–53.

Hillis, D. M., Moritz, C., Porter, C. A., and Baker, R. J. (1991). Evidence for biased gene conversion in concerted evolution of ribosomal DNA. *Science*, 251(4991):308–10.

Hood, L., Campbell, J. H., and Elgin, S. C. (1975). The organization, expression, and evolution of antibody genes and other multigene families. *Annu Rev Genet*, 9:305–53.

Hua, J., Craig, D. W., Brun, M., Webster, J., Zismann, V., Tembe, W., Joshipura, K., Huentelman, M. J., Dougherty, E. R., and Stephan, D. A. (2007). SNiPer-HD: improved genotype calling accuracy by an expectation-maximization algorithm for high-density SNP arrays. *Bioinformatics*, 23(1):57–63.

Humphries, N. E., Queiroz, N., Dyer, J. R. M., Pade, N. G., Musyl, M. K., Schaefer, K. M., Fuller, D. W., Brunnschweiler, J. M., Doyle, T. K., Houghton, J. D. R., Hays, G. C., Jones, C. S., Noble, L. R., Wearmouth, V. J., Southall, E. J., and Sims, D. W. (2010). Environmental context explains Lévy and Brownian movement patterns of marine predators. *Nature*, 465(7301):1066–9.

Humphries, N. E., Weimerskirch, H., Queiroz, N., Southall, E. J., and Sims, D. W. (2012). Foraging success of biological Lévy flights recorded in situ. *Proc Natl Acad Sci U S A*, 109(19):7169–74.

Huson, D. H. and Bryant, D. (2006). Application of phylogenetic networks in evolutionary studies. *Mol Biol Evol*, 23(2):254–67.

Ide, S., Miyazaki, T., Maki, H., and Kobayashi, T. (2010). Abundance of ribosomal RNA gene copies maintains genome integrity. *Science*, 327(5966):693–6.

Iqbal, Z., Caccamo, M., Turner, I., Flicek, P., and McVean, G. (2012). De novo assembly and genotyping of variants using colored de Bruijn graphs. *Nat Genet*, 44(2):226–32.

James, S. A., O'Kelly, M. J. T., Carter, D. M., Davey, R. P., van Oudenaarden, A., and Roberts, I. N. (2009). Repetitive sequence variation and dynamics in the ribosomal DNA array of *Saccharomyces cerevisiae* as revealed by whole-genome resequencing. *Genome Res*, 19(4):626–35.

Johansen, T., Carlson, C. R., and Kolsto, A. B. (1996). Variable numbers of rRNA gene operons in *Bacillus cereus* strains. *FEMS Microbiol Lett*, 136(3):325–8.

Johnson, F. B., Sinclair, D. A., and Guarente, L. (1999). Molecular biology of aging. *Cell*, 96(2):291–302.

Judd, S. R. and Petes, T. D. (1988). Physical lengths of meiotic and mitotic gene conversion tracts in *Saccharomyces cerevisiae*. *Genetics*, 118(3):401–10.

Kaeberlein, M. (2010). Lessons on longevity from budding yeast. *Nature*, 464(7288):513–9.

Katoh, K. and Standley, D. M. (2013). MAFFT multiple sequence alignment software version 7: improvements in performance and usability. *Mol. Biol. Evol.*, 30(4):772–80.

Keirle, M. R., Avis, P. G., Hemmes, D. E., and Mueller, G. M. (2011). Variability in the IGS1 region of *Rhodocollybia laulaha*: is it allelic, genomic, or artificial? *Fungal Biol*, 115(3):310–6.

Kim, Y.-H., Ishikawa, D., Ha, H. P., Sugiyama, M., Kaneko, Y., and Harashima, S. (2006). Chromosome XII context is important for rDNA function in yeast. *Nucleic Acids Res*, 34(10):2914–24.

Kiss, L. (2012). Limits of nuclear ribosomal DNA internal transcribed spacer (ITS) sequences as species barcodes for Fungi. *Proc Natl Acad Sci U S A*, 109(27):E1811; author reply E1812.

Klappenbach, J. A., Dunbar, J. M., and Schmidt, T. M. (2000). rRNA operon copy number reflects ecological strategies of bacteria. *Appl Environ Microbiol*, 66(4):1328–33.

Klappenbach, J. A., Saxman, P. R., Cole, J. R., and Schmidt, T. M. (2001). rrndb: the Ribosomal RNA Operon Copy Number Database. *Nucleic Acids Res*, 29(1):181–4.

Kobayashi, T. (2011). Regulation of ribosomal RNA gene copy number and its role in modulating genome integrity and evolutionary adaptability in yeast. *Cell Mol Life Sci*, 68(8):1395–403.

Kobayashi, T., Heck, D. J., Nomura, M., and Horiuchi, T. (1998). Expansion and contraction of ribosomal DNA repeats in *Saccharomyces cerevisiae*: requirement of replication fork blocking (Fob1) protein and the role of RNA polymerase I. *Genes Dev*, 12(24):3821–30.

Kobayashi, T., Horiuchi, T., Tongaonkar, P., Vu, L., and Nomura, M. (2004). SIR2 regulates recombination between different rDNA repeats, but not recombination within individual rRNA genes in yeast. *Cell*, 117(4):441–53.

Kumar, P. (2011). Computational Analysis of Ribosomal DNA Dynamics in Yeast. Master's thesis, School of Computing Science.

Kurtzman, C. P. and Robnett, C. J. (1998). Identification and phylogeny of ascomycetous yeasts from analysis of nuclear large subunit (26S) ribosomal DNA partial sequences. *Antonie Van Leeuwenhoek*, 73(4):331–71.

Kwan, E. X., Foss, E. J., Tsuchiyama, S., Alvino, G. M., Kruglyak, L., Kaeberlein, M., Raghuraman, M. K., Brewer, B. J., Kennedy, B. K., and Bedalov, A. (2013).

A natural polymorphism in rDNA replication origins links origin activation with calorie restriction and lifespan. *PLoS Genet*, 9(3):e1003329.

Lachance, M. A., Daniel, H. M., Meyer, W., Prasad, G. S., Gautam, S. P., and Boundy-Mills, K. (2003). The D1/D2 domain of the large-subunit rDNA of the yeast species *Clavispora lusitaniae* is unusually polymorphic. *FEMS Yeast Res*, 4(3):253–8.

Lander, E. S. and Waterman, M. S. (1988). Genomic mapping by fingerprinting random clones: a mathematical analysis. *Genomics*, 2(3):231–9.

Lang, G. I. and Murray, A. W. (2008). Estimating the per-base-pair mutation rate in the yeast *Saccharomyces cerevisiae*. *Genetics*, 178(1):67–82.

League, G. P., Slot, J. C., and Rokas, A. (2012). The ASP3 locus in *Saccharomyces cerevisiae* originated by horizontal gene transfer from *Wickerhamomyces*. *FEMS Yeast Res.*, 12(7):859–63.

Lee, Z. M.-P., Bussema, C., and Schmidt, T. M. (2009). rrnDB: documenting the number of rRNA and tRNA genes in bacteria and archaea. *Nucleic Acids Res.*, 37(Database issue):D489–93.

Li, H. and Durbin, R. (2009). Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–60.

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R., and 1000 Genome Project Data Processing Subgroup (2009). The Sequence Alignment/Map format and SAMtools. *Bioinformatics*, 25(16):2078–9.

Liao, D. (2000). Gene conversion drives within genic sequences: concerted evolution of ribosomal RNA genes in bacteria and archaea. *J Mol Evol*, 51(4):305–17.

Lindner, D. L. and Banik, M. T. (2011). Intragenomic variation in the ITS rDNA region obscures phylogenetic relationships and inflates estimates of operational taxonomic units in genus *Laetiporus*. *Mycologia*, 103(4):731–40.

Liti, G., Barton, D. B. H., and Louis, E. J. (2006). Sequence diversity, reproductive isolation and species concepts in *Saccharomyces*. *Genetics*, 174(2):839–50.

Liti, G., Carter, D. M., Moses, A. M., Warringer, J., Parts, L., James, S. A., Davey, R. P., Roberts, I. N., Burt, A., Koufopanou, V., Tsai, I. J., Bergman, C. M., Bensasson, D., O'Kelly, M. J. T., van Oudenaarden, A., Barton, D. B. H., Bailes, E., Nguyen, A. N., Jones, M., Quail, M. A., Goodhead, I., Sims, S., Smith, F., Blomberg, A., Durbin, R., and Louis, E. J. (2009). Population genomics of domestic and wild yeasts. *Nature*, 458(7236):337–41.

Liu, W., Di, X., Yang, G., Matsuzaki, H., Huang, J., Mei, R., Ryder, T. B., Webster, T. A., Dong, S., Liu, G., Jones, K. W., Kennedy, G. C., and Kulp, D. (2003). Algorithms for large-scale genotyping microarrays. *Bioinformatics*, 19(18):2397–403.

Loftus, B. J., Fung, E., Roncaglia, P., Rowley, D., Amedeo, P., Bruno, D., Vamathevan, J., Miranda, M., Anderson, I. J., Fraser, J. A., Allen, J. E., Bosdet, I. E., Brent, M. R., Chiu, R., Doering, T. L., Donlin, M. J., D'Souza, C. A., Fox, D. S., Grinberg, V., Fu, J., Fukushima, M., Haas, B. J., Huang, J. C., Janbon, G., Jones, S. J. M., Koo, H. L., Krzywinski, M. I., Kwon-Chung, J. K., Lengeler, K. B., Maiti, R., Marra, M. A., Marra, R. E., Mathewson, C. A., Mitchell, T. G., Pertea, M., Riggs, F. R., Salzberg, S. L., Schein, J. E., Shvartsbeyn, A., Shin, H., Shumway, M., Specht, C. A., Suh, B. B., Tenney, A., Utterback, T. R., Wickes, B. L., Wortman, J. R., Wye, N. H., Kronstad, J. W., Lodge, J. K., Heitman, J., Davis, R. W., Fraser, C. M., and Hyman, R. W. (2005). The genome of the basidiomycetous yeast and human pathogen *Cryptococcus neoformans*. *Science*, 307(5713):1321–4.

Long, Q., Rabanal, F. A., Meng, D., Huber, C. D., Farlow, A., Platzer, A., Zhang, Q., Vilhjálmsson, B. J., Korte, A., Nizhynska, V., Voronin, V., Korte, P., Sedman, L., Mandáková, T., Lysak, M. A., Seren, U., Hellmann, I., and Nordborg, M. (2013). Massive genomic variation and strong selection in *Arabidopsis thaliana* lines from Sweden. *Nat. Genet.*, 45(8):884–90.

Lunter, G. and Goodson, M. (2011). Stampy: a statistical algorithm for sensitive and fast mapping of Illumina sequence reads. *Genome Res*, 21(6):936–9.

Lynch, M., Sung, W., Morris, K., Coffey, N., Landry, C. R., Dopman, E. B., Dickinson, W. J., Okamoto, K., Kulkarni, S., Hartl, D. L., and Thomas, W. K. (2008). A genome-wide view of the spectrum of spontaneous mutations in yeast. *Proc. Natl. Acad. Sci. U.S.A.*, 105(27):9272–7.

McMahon, M. E., Stamenkovich, D., and Petes, T. D. (1984). Tandemly arranged variant 5S ribosomal RNA genes in the yeast *Saccharomyces cerevisiae*. *Nucleic Acids Res*, 12(21):8001–16.

McStay, B. and Grummt, I. (2008). The epigenetics of rRNA genes: from molecular to chromosome biology. *Annu Rev Cell Dev Biol*, 24:131–57.

Merker, R. J. and Klein, H. L. (2002). hpr1Delta affects ribosomal DNA recombination and cell life span in *Saccharomyces cerevisiae*. *Mol Cell Biol*, 22(2):421–9.

Montrocher, R., Verner, M. C., Briolay, J., Gautier, C., and Marmeisse, R. (1998). Phylogenetic analysis of the *Saccharomyces cerevisiae* group based on polymorphisms of rDNA spacer sequences. *Int J Syst Bacteriol*, 48 Pt 1:295–303.

Nagylaki, T. and Petes, T. D. (1982). Intrachromosomal gene conversion and the maintenance of sequence homogeneity among repeated genes. *Genetics*, 100(2):315–37.

Naumov, G. I., James, S. A., Naumova, E. S., Louis, E. J., and Roberts, I. N. (2000). Three new species in the *Saccharomyces sensu stricto* complex: *Saccharomyces cariocanus*, *Saccharomyces kudriavzevii* and *Saccharomyces mikatae*. *Int J Syst Evol Microbiol*, 50 Pt 5:1931–42.

Nickoloff, J. A., Sweetser, D. B., Clikeman, J. A., Khalsa, G. J., and Wheeler, S. L. (1999). Multiple heterologies increase mitotic double-strand break-induced allelic gene conversion tract lengths in yeast. *Genetics*, 153(2):665–79.

Nilsson, R. H., Kristiansson, E., Ryberg, M., Hallenberg, N., and Larsson, K.-H. (2008). Intraspecific ITS variability in the kingdom fungi as expressed in the international sequence databases and its implications for molecular species identification. *Evol Bioinform Online*, 4:193–201.

Ning, Z., Cox, A. J., and Mullikin, J. C. (2001). SSAHA: a fast search method for large DNA databases. *Genome Res*, 11(10):1725–9.

Nishant, K. T., Wei, W., Mancera, E., Argueso, J. L., Schlattl, A., Delhomme, N., Ma, X., Bustamante, C. D., Korbel, J. O., Gu, Z., Steinmetz, L. M., and Alani,

E. (2010). The baker's yeast diploid genome is remarkably stable in vegetative growth and meiosis. *PLoS Genet*, 6(9):e1001109.

Ohta, T. (1976). Simple model for treating evolution of multigene families. *Nature*, 263(5572):74–6.

O'Kelly, M. J. T. (2008). *Silencing and recombination in yeast ribosomal DNA*. PhD thesis, Massachusetts Institute of Technology. Dept. of Physics.

Oliveros, J. (2007). VENNY. An interactive tool for comparing lists with Venn Diagrams. `http://bioinfogp.cnb.csic.es/tools/venny/index.html`.

Olsen, G. J. and Woese, C. R. (1993). Ribosomal RNA: a key to phylogeny. *FASEB J*, 7(1):113–23.

Ozenberger, B. A. and Roeder, G. S. (1991). A unique pathway of double-strand break repair operates in tandemly repeated genes. *Mol Cell Biol*, 11(3):1222–31.

Pâques, F. and Haber, J. E. (1999). Multiple pathways of recombination induced by double-strand breaks in *Saccharomyces cerevisiae*. *Microbiol Mol Biol Rev*, 63(2):349–404.

Paredes, S., Branco, A. T., Hartl, D. L., Maggert, K. A., and Lemos, B. (2011). Ribosomal DNA deletions modulate genome-wide gene expression: "rDNA-sensitive" genes and natural variation. *PLoS Genet*, 7(4):e1001376.

Perelson, A. S. and Bell, G. I. (1977). Mathematical models for the evolution of multigene families by unequal crossing over. *Nature*, 265(5592):304–10.

Petes, T. D. (1979). Yeast ribosomal DNA genes are located on chromosome XII. *Proc Natl Acad Sci U S A*, 76(1):410–4.

Petes, T. D. (1980). Unequal meiotic recombination within tandem arrays of yeast ribosomal DNA genes. *Cell*, 19(3):765–74.

Prokopowich, C. D., Gregory, T. R., and Crease, T. J. (2003). The correlation between rDNA copy number and genome size in eukaryotes. *Genome*, 46(1):48–50.

Proux-Wéra, E., Byrne, K. P., and Wolfe, K. H. (2013). Evolutionary mobility of the ribosomal DNA array in yeasts. *Genome Biol Evol*, 5(3):525–31.

Pruesse, E., Quast, C., Knittel, K., Fuchs, B. M., Ludwig, W., Peplies, J., and Glöckner, F. O. (2007). SILVA: a comprehensive online resource for quality checked and aligned ribosomal RNA sequence data compatible with ARB. *Nucleic Acids Res*, 35(21):7188–96.

Puigbò, P., Garcia-Vallvé, S., and McInerney, J. (2007). TOPD/FMTS: a new software to compare phylogenetic trees. *Bioinformatics*, 12(23):1556–8.

Quinlan, A. R., Stewart, D. A., Strömberg, M. P., and Marth, G. T. (2008). Pyrobayes: an improved base caller for SNP discovery in pyrosequences. *Nat Methods*, 5(2):179–81.

R Development Core Team (2011). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Ramakrishnan, V. (1986). Distribution of protein and RNA in the 30S ribosomal subunit. *Science*, 231(4745):1562–4.

Ramakrishnan, V. (2002). Ribosome structure and the mechanism of translation. *Cell*, 108(4):557–72.

Ramazzotti, M., BernÃ¡, L., Stefanini, I., and Cavalieri, D. (2012). A computational pipeline to discover highly phylogenetically informative genes in sequenced genomes: application to *Saccharomyces cerevisiae* natural strains. *Nucleic Acids Res*, 40(9):3834–48.

Ratan, A., Zhang, Y., Hayes, V. M., Schuster, S. C., and Miller, W. (2010). Calling SNPs without a reference sequence. *BMC Bioinformatics*, 11:130.

Richard, G.-F., Kerrest, A., and Dujon, B. (2008). Comparative genomics and molecular dynamics of DNA repeats in eukaryotes. *Microbiol Mol Biol Rev*, 72(4):686–727.

Roberts, R. J., Carneiro, M. O., and Schatz, M. C. (2013). The advantages of SMRT sequencing. *Genome Biol*, 14(6):405.

Ross, M. G., Russ, C., Costello, M., Hollinger, A., Lennon, N. J., Hegarty, R., Nusbaum, C., and Jaffe, D. B. (2013). Characterizing and measuring bias in sequence data. *Genome Biol*, 14(5):R51.

Saitou, N. and Nei, M. (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol*, 4(4):406–25.

Schaal, B. and Learn, G. (1988). Ribosomal DNA variation within and among Plant Populations. *Annals of the Missouri Botanical Garden*, 75:1207–1216.

SGD (2013). The Saccharomyces Genome Database. `http://www.yeastgenome.org/`.

SGRP (2013). The Saccharomyces Genome Resequencing Project. `http://www.sanger.ac.uk/research/projects/genomeinformatics/sgrp.html/`.

Simon, U. K., Trajanoski, S., Kroneis, T., Sedlmayr, P., Guelly, C., and Guttenberger, H. (2012). Accession-specific haplotypes of the internal transcribed spacer region in *Arabidopsis thaliana*–a means for barcoding populations. *Mol Biol Evol*, 29(9):2231–9.

Sinclair, D. A. and Guarente, L. (1997). Extrachromosomal rDNA circles–a cause of aging in yeast. *Cell*, 91(7):1033–42.

Smith, G. P. (1976). Evolution of repeated DNA sequences by unequal crossover. *Science*, 191(4227):528–35.

Stage, D. E. and Eickbush, T. H. (2007). Sequence variation within the rRNA gene loci of 12 *Drosophila* species. *Genome Res*, 17(12):1888–97.

Stajich, J. E., Block, D., Boulez, K., Brenner, S. E., Chervitz, S. A., Dagdigian, C., Fuellen, G., Gilbert, J. G. R., Korf, I., Lapp, H., Lehväslaiho, H., Matsalla, C., Mungall, C. J., Osborne, B. I., Pocock, M. R., Schattner, P., Senger, M., Stein, L. D., Stupka, E., Wilkinson, M. D., and Birney, E. (2002). The Bioperl toolkit: Perl modules for the life sciences. *Genome Res*, 12(10):1611–8.

Stamatakis, A. (2006). RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, 22(21):2688–90.

Steinkraus, K. A., Kaeberlein, M., and Kennedy, B. K. (2008). Replicative aging in yeast: the means to the end. *Annu Rev Cell Dev Biol*, 24:29–54.

Strand, M., Prolla, T. A., Liskay, R. M., and Petes, T. D. (1993). Destabilization of tracts of simple repetitive DNA in yeast by mutations affecting DNA mismatch repair. *Nature*, 365(6443):274–6.

Stults, D. M., Killen, M. W., Pierce, H. H., and Pierce, A. J. (2008). Genomic architecture and inheritance of human ribosomal RNA gene clusters. *Genome Res*, 18(1):13–8.

Stults, D. M., Killen, M. W., Williamson, E. P., Hourigan, J. S., Vargas, H. D., Arnold, S. M., Moscow, J. A., and Pierce, A. J. (2009). Human rRNA gene clusters are recombinational hotspots in cancer. *Cancer Res*, 69(23):9096–104.

Szostak, J. W., Orr-Weaver, T. L., Rothstein, R. J., and Stahl, F. W. (1983). The double-strand-break repair model for recombination. *Cell*, 33(1):25–35.

Szostak, J. W. and Wu, R. (1980). Unequal crossing over in the ribosomal DNA of *Saccharomyces cerevisiae*. *Nature*, 284(5755):426–30.

Tammi, M. T., Arner, E., and Andersson, B. (2003). TRAP: Tandem Repeat Assembly Program produces improved shotgun assemblies of repetitive sequences. *Comput Methods Programs Biomed*, 70(1):47–59.

Tamura, K., Peterson, D., Peterson, N., Stecher, G., Nei, M., and Kumar, S. (2011). MEGA5: molecular evolutionary genetics analysis using maximum likelihood, evolutionary distance, and maximum parsimony methods. *Mol Biol Evol*, 28(10):2731–9.

Turova, T. P. (2003). Copy number of ribosomal operons in prokaryotes and its effect on phylogenic analyses. *Mikrobiologiia*, 72(4):437–52.

Uemura, M., Zheng, Q., Koh, C. M., Nelson, W. G., Yegnasubramanian, S., and De Marzo, A. M. (2012). Overexpression of ribosomal RNA in prostate cancer is common but not linked to rDNA promoter hypomethylation. *Oncogene*, 31(10):1254–63.

Unneberg, P., Strömberg, M., and Sterky, F. (2005). SNP discovery using advanced algorithms and neural networks. *Bioinformatics*, 21(10):2528–30.

Warner, J. R. (1999). The economics of ribosome biosynthesis in yeast. *Trends Biochem Sci*, 24(11):437–40.

Weiss, H. and Maluszynska, J. (2000). Chromosomal rearrangement in autotetraploid plants of *Arabidopsis thaliana*. *Hereditas*, 133(3):255–61.

Witte, C., Jensen, R. E., Yaffe, M. P., and Schatz, G. (1988). MAS1, a gene essential for yeast mitochondrial assembly, encodes a subunit of the mitochondrial processing protease. *EMBO J*, 7(5):1439–47.

Woese, C. R. (2000). Interpreting the universal phylogenetic tree. *Proc Natl Acad Sci U S A*, 97(15):8392–6.

Yin, J., Jordan, M. I., and Song, Y. S. (2009). Joint estimation of gene conversion rates and mean conversion tract lengths from population SNP data. *Bioinformatics*, 25(12):i231–9.

Yonath, A. (2005). Antibiotics targeting ribosomes: resistance, selectivity, synergism and cellular regulation. *Annu Rev Biochem*, 74:649–79.

Zamb, T. J. and Petes, T. D. (1982). Analysis of the junction between ribosomal RNA genes and single-copy chromosomal sequences in the yeast *Saccharomyces cerevisiae*. *Cell*, 28(2):355–64.

Zimmer, E. A., Martin, S. L., Beverley, S. M., Kan, Y. W., and Wilson, A. C. (1980). Rapid duplication and loss of genes coding for the alpha chains of hemoglobin. *Proc Natl Acad Sci U S A*, 77(4):2158–62.