Supporting Text

Diploid Assembly

The inputs to the diploid assembly were the sequences and base quality scores of the 2,519 contigs from the PHRAP assembly and a set of manual assembler directives. The outputs included supercontig sequences and base qualities, a "scrap" file containing sequence not assembled (mostly trimmed low-quality bases), and several descriptive and diagnostic files used in subsequent analyses.

Initially, we ran the assembler without any manual directives to produce a rough draft of the diploid genome. We examined the draft in detail for consistency with other sources of information such as physical mapping data, paired plasmid end sequences, and known *Candida albicans* sequences in GenBank. The result of this analysis was a set of manual assembler directives. Most of the directives instructed the assembler to join together PHRAP contigs that it had not joined automatically. A few prevented misassemblies that conflicted with verified map data. To produce the final diploid assembly, we ran the diploid assembler on the PHRAP contigs again, this time including the manual directives.

To produce the diploid genome sequence, the assembler went through the following sequence of steps.

Alignment and identification of repeats. Many of the PHRAP contigs were very short and contained very few reads. It appeared that most contigs containing less than four reads, and also less than 1,500 bp (regardless of quality), contained contaminant sequence, artifacts, or corrupted *C. albicans* sequence. We excluded such contigs from the diploid assembly, leaving 1,428 PHRAP contigs to be assembled. All contigs was searched against all others using NCBI BLASTN version 2.0.14, with a nucleotide mismatch penalty of -2, and other parameters adjusted to enable BLAST to cross larger gaps. All alignments that contained 200 bp or more from either of the two aligned PHRAP contigs were added to a database of alignments used throughout the assembly. Additional alignments could be added to the database if needed to support manual directives. Sequences that appeared, based on the alignment database, to have four or more approximate repeats in the PHRAP assembly were designated as high-copy repeats. The locations of all high-copy repeat sequences were stored in a repeat database. Repeat locations in the diploid and PHRAP assemblies are shown on the supercontig maps (see accompanying data files).

Identification of chimeric contig ends. The assembler identified the sequence dependent on a single read at each contig end and aligned it against the rest of the PHRAP assembly. Alignment information of this kind did not always identify chimeric contig ends with high accuracy (these are often chimeras already undetected by PHRAP), and the presence of separately assembled homologous sequences added to the difficulty. We used a conservative rule that marked chimeras only when clearly indicated. Single-coverage sequence at contig ends was identified and searched against the entire PHRAP assembly by using BLASTN. Sufficiently long BLAST alignments that were nearly exact matches, that could not be extended significantly into multiple-coverage contig sequence and that did not omit a significant fraction of high-quality bases approaching the end of the contig sequence were flagged as probable alignments of chimeric sequence. Single-coverage bases participating in such alignments were marked as chimeric, along with bases at the left end of the contig lying to the left of such bases, and bases at the right end lying to their right.

Finding terminal alignments. Each alignment in the alignment database was tested for terminality. Consider an alignment between PHRAP contigs A and B, with m unaligned bases at positions 1, 2, ..., m of contig A. We describe the statistical test for determining whether the alignment is terminal at the left end of contig A; that is, whether the failure of bases 1, 2, ..., m of contig A to be included in the alignment could have been caused by sequencing errors even if the actual sequence at those positions is still similar to sequence in contig B. The same test of

terminality, with appropriate changes, would be performed at the right end of A and at both ends of B. For simplicity of exposition, we assume that no bases at the left end of A are marked as chimeric. When chimeric bases were present, the test was applied to the nonchimeric bases only, but was otherwise unchanged.

The statistical terminality test was based on three assumptions:

(i) the sequences of true homologs are identical;
(ii) no bases are inserted or deleted by sequencing errors;
(iii) the sequence of contig B is error-free.

Assumption (i) is further discussed below. Assumptions (ii) and (iii), although not strictly true, are reasonable approximations. In the case of (iii) this is so because the low-quality sequence from the end of contig A is being compared to the higher-quality sequence internal to B.

For $1 \leq i \leq m$, let q_i be the quality of base i in contig A, and let $p_i = 10^{-q_i/10}$ be the error probability (1,2). Define a random variable X_i to be 1 if a sequencing error occurs at base position i, and 0 otherwise. If bases 1, 2, ..., m were added to the alignment, their contribution to the raw BLAST score would be $\sum_i (1-X_i) - 2X_i$. Because BLAST did not align these bases, this contribution must be negative. It follows that $S = \sum_i X_i > m/3$. Assuming that sequencing errors at different positions are independent, we have that $E(S) = \sum_i p_i$ and $\mathrm{Var}(S) = \sum_i p_i(1-p_i)$. Using these values, we can shift and rescale S to obtain a random variable Z with mean zero and variance 1:

$$Z = \frac{S - E(S)}{\sqrt{\mathrm{Var}(S)}} > \frac{m/3 - E(S)}{\sqrt{\mathrm{Var}(S)}}.$$

If the alignment is between true homologs, and if the fraction of bases of quality zero is bounded away from one, our assumptions imply that the distribution of Z approaches the standard normal for large m [e.g., by the Lyapounov Central Limit Theorem (3)]. For small values of m, the normal approximation may not be very good, but in such cases the alignment can reasonably be declared terminal regardless of the base qualities. The values of S (the number of sequencing errors) and Z are not known in practice, but the lower bound on the right of the inequality is available. Large values of this bound indicate that failure of the first m bases to align is unlikely to be explained by low sequence quality.

The foregoing discussion relies on assumption (i), which is not true in general. Because at this stage we had no data to provide a sound basis for modeling heterozygous polymorphism, we used a rule based on the above lower bound for Z with a conservative threshold in recognition of the incompleteness of the statistical model. Specifically, the alignment was classified as terminal if

(i) $m \leq 20$; or
(ii) $\mathrm{Var}(S) = 0$ (this happens if and only if all $q_i = 0$); or
(iii) $\mathrm{Var}(S) > 0$ and $\big(m/3 - E(S)\big)/\sqrt{\mathrm{Var}(S)} > 1.15$, approximately the 0.875 quantile of the standard normal distribution.

As a computational optimization, alignments with $m > 5{,}000$ were immediately declared nonterminal. The derivation of the test suggests that it may misclassify a fairly high proportion (nominally 0.125) of terminal alignments as nonterminal, but we preferred a conservative test. The problem was mitigated in practice by several factors, such as PHRAP's tendency to leave large blocks of zero-quality bases at contig ends. In the final diploid assembly, manual directives were used to correct several cases where it was evident (e.g., based on mapping data) that the test had failed to recognize terminal alignments of homologs.

Identification of alignments of repeat sequence. If the alignment was classified as terminal by the above rule, then the 200 bases at positions $m+1$, $m+2$, ..., $m+200$ in contig A

were checked against the repeat database. If any of these bases were repeat sequence, the alignment was designated as a repeat alignment. Adjacent to short repeats such as tRNAs, the required number of adjacent unique sequence bases was reduced below 200, but never to less than 100 bp. The same repeat test was applied at the other end of A, and at both ends of B, if the alignment was terminal at those locations. Repeat alignments were used to produce various diagnostic outputs, but not to perform assembly except when manually directed. We experimented with variants of the repeat rule and found that although some rule of this kind was needed, the results were not highly sensitive to minor variants of it.

Identification of potential joins and inclusions. We call an alignment between homologs like the one shown in Fig. 1*b* a join, and an alignment of the type in Fig. 1*c* an inclusion. The first step of the assembly proper was to make a list of potential joins and inclusions ("potential" because they had to pass additional tests as the assembly progressed). First, all manually directed joins and inclusions, if any, were placed on the list. Next the assembler searched the alignment database to identify additional potential joins and inclusions based on terminality and orientation.

For each alignment, the terminality test produced four yes or no results, indicating whether the alignment could be regarded as extending to either end of each of the two aligned contigs. Potential joins were identified by finding alignments that were terminal once in each aligned contig and had alignment directions in the two contigs consistent with the layout shown in Fig. 1*b*. Such joins, based on a single alignment, were called simple joins to distinguish them from other types used in more complex situations. Alignments that were terminal at both ends of one of the two aligned contigs and that were not repeat alignments, were identified as potential simple inclusions (potential join repeat alignments did go on the list and were not removed until later).

In certain regions of the genome, we expected homologous sequences to be sufficiently diverged to prevent BLAST from constructing a complete alignment of the overlapping terminal regions. Examples include the mating-type locus and several cases of transposon insertions into one homolog. To handle such regions, the assembler looked for pairs of alignments that appeared, based on orientation, terminality, and nonrepeat alignment status, to correspond to the ends of a diverged homologous region. When such alignment pairs were found, the assembler effectively constructed an end-to-end alignment of the diverged region by combining the two end alignments with a large substitution in the middle. If the combined alignment had the proper terminality and orientation, then, as in the simple case, a potential join or inclusion was added to the list. Joins and inclusions based on paired BLAST alignments were called complex to distinguish them from the simple joins and inclusions based on one alignment.

The vast majority of the assembly was guided by simple and complex joins and inclusions. Most of these were found automatically. In the final assembly, a small number were manually directed. A few exceptional situations called for two additional types of joins. Our data had a small number of gaps that were covered by GenBank sequences. We filled such gaps with *Ns*, with the number of *Ns* being estimated from the GenBank sequence. Sometimes no *Ns* were needed because our contigs already had small overlaps (too small to be included in the assembler's alignment database). This type of gap filling was performed by using special manually directed joins that we called N joins.

Another unusual situation arose when a large insertion existed in one homolog, and part of the sequence of the insertion was missing in the PHRAP assembly, because of either an actual gap in coverage, or, more likely, a repeat sequence in the insertion. An analysis like that of Fig. 1, but considerably more complicated, predicted that in this situation PHRAP would produce three contigs and two terminal alignments. There are several possible configurations with variations in details; in some cases, with an intact retrovirus on one homolog and LTR on the other, additional terminal alignments between LTR sequences complicated the picture. The diploid assembler contained code capable of performing "three-way joins" that assembled the correct diploid sequence in such cases,

with the missing segment of the insertion replaced by a sequence of 100 N bases. Analysis showed that three-way inclusions might also theoretically be needed, but none appeared to be called for in our assembly. Because these situations were both uncommon and complex (especially when LTR were involved), we concluded that three-way joins should be made only via manual directives.

Table 6 shows the number of joins and inclusions of each type that were finally used.

Identification of included PHRAP contigs. In Fig. 1*c*, we called the smaller contig ("Contig w"), on which the alignment is terminal at both ends, an included contig. Under the assumption of a diploid genome, because the included contig is aligned end to end with its homolog, it should not participate in any other alignments of homologous sequence. Included contigs therefore provide homologous sequence in localized regions but do not influence the large-scale structure of supercontigs.

The assembler identified included contigs so that they could be set aside until the supercontig structures, based on joins, were determined. In general, this was a straightforward process of finding contigs that participated in potential inclusions as the included contig. Not surprisingly, a few cases arose that behaved differently than predicted by the analysis of Fig. 1, which required special handling. Four contigs were included in more than one place. We opted to undercount polymorphism rather than overcount genes and excluded these contigs from the assembly. They appear in the scrap file. One pair of contigs aligned end to end (so that each included the other); of the two, one was of conspicuously lower quality and was placed in the scrap file.

In 23 cases, contigs identified as included also participated in join alignments. These joins were removed from the list of potential joins on the basis that the longer inclusion alignments were more likely correct; the contigs involved were then assembled on the same basis as the others. Examination of a few cases where this occurred revealed various causes, of which the most common was repeat sequence at contig ends. This was as expected given the retention of repeat alignment joins on the potential join list at this stage of the assembly.

The assembly graph. To determine the supercontig structures, the assembler constructed an undirected graph whose vertices represented the ends of the nonincluded contigs (two vertices per contig), and whose edges represented the potential joins remaining on the list [for graph theory terminology, see Bollobás (4)]. Then, in a process we called edge reduction, the assembler removed graph edges according to the following rules: (*i*) any edges sharing a common vertex with a manually directed edge were deleted; (*ii*) all edges based on repeat alignments, and not manually directed, were deleted; (*iii*) when two or more edges joined the same vertices and the associated alignments were essentially the same (this arose when the BLAST alignment of contig A against B differed slightly from the alignment of B against A), then all but one of the essential duplicates were deleted; (*iv*) if, after application of (*i*), (*ii*), and (*iii*), a vertex still had multiple edges incident with it, all such edges were deleted. In the final assembly, these rules deleted 99 of 782 potential joins. Deletion of repeat alignments eliminated 72; deletion of edges competing with manual directives, 17; and deletion of duplicate alignments, 4. Applications of rule (*iv*), indicating actual uncertainty about how to assemble, deleted only six edges. The assembler retained information on all deleted edges for use in descriptive and diagnostic output.

After edge reduction, the assembler added edges to the graph connecting each vertex with the vertex representing the other end of the same contig. We call these internal edges. After this step, every vertex was incident with exactly one internal edge, and at most one join edge, so that all vertices had degree one or two. Therefore every connected component of the graph was either a path or a cycle. The assembler performed cycle detection and contained code to break cycles by deleting one join edge in the cycle, but in our assembly no cycles arose. Thus each component of the graph was a path, and each such path represented a supercontig or homologous pair of supercontigs.

Supercontig assembly and sequence generation. Assembly of supercontigs proceeded

by searching the graph for a vertex of degree one not marked as already assembled, and following the path of alternating internal and join edges from that start vertex. Fig. 6 shows the typical situation as this process progressed. In Fig. 6, assembly has proceeded through the join of graph vertices s and t, so that both homolog sequences are known through the position marked as L. The sequence shown in gray at s has been trimmed and placed in the scrap file.

To continue, the assembler located the other end of the t–u PHRAP contig by following the internal edge leaving t, and then located the next PHRAP contig end to join by following the join edge from u to v. This was the first time the assembler encountered v's PHRAP contig, so that contig was assigned to the homolog opposite that of the t–u contig (with three-way joins, the homolog assignment is more complex). The entire sequence from v's PHRAP contig, except for trimmed bases, would appear intact in its assigned homologous supercontig.

To perform the join between u and v, bases with low PHRAP quality scores were located and removed from the alignment at both contig ends. These bases, shown in gray in Fig. 6, were trimmed off and placed in the scrap file. The trimming algorithm aimed to achieve 99% or better sequence accuracy after trimming. Care was taken not to trim at positions inside gaps in the alignment. Sequence trimmed in this way corresponds to the homozygous sequence predicted in Fig. 1*b* to occur at terminal alignment ends.

In relatively uncommon cases, one aligned PHRAP contig contained large (hundreds of bases) internal regions of very poor sequence quality. This phenomenon would be expected if two heterozygous regions were separated by a short nearly homozygous region, whose length was such as to barely allow reads extending from the heterozygous regions to meet in the middle. In such a case, one PHRAP contig would contain a good-quality version of the nearly homozygous sequence, incorporating all reads from the central region, whereas another would contain a poor one reconstructed only from low-quality read ends. The trimming algorithm was allowed to replace internal poor-quality sequence in one homolog with sequence from the other to correct quality dropouts of this type. Fig. 6 shows internal trimming in the s/t join. Internal trimming was not performed in complex or three-way joins.

Inclusions into the t–u PHRAP contig were performed after making the join between u and v. All PHRAP contigs having inclusion alignments with the t–u contig were located, positioned between the two join alignments, and assigned to the homolog opposite t–u. Inclusions were subject to trimming by the same algorithm used for joins.

At this stage, assembly through the join of u and v was complete, and the assembler continued with v in the role previously played by t. The process went on until the assembler reached the end of the supercontig pair, identified by the lack of a join edge in the assembly graph at u. The logic at the beginning and end of the supercontig pair was essentially the same as outlined here, except that at the beginning, there was no s/t join to make, and at the end there was no u/v join. Instead, the assembler removed low-quality sequence at the supercontig ends, similar to the trimming in joins and inclusions. Again, trimmed sequences were placed in the scrap file.

Once the construction of a supercontig pair was complete, the two homologous sequences and base quality scores were created; it was at this point that sequence in nearly homozygous regions was copied into both homologs as shown in Fig. 2. The assembler also produced descriptions of where each base in the supercontig pair originated in the PHRAP assembly, and diagnostics such as enumeration of any potential joins not used because of deletion during edge reduction. In the case of PHRAP contigs having no join or inclusion alignments, the assembler generated only a single sequence rather than a pair, identical to the PHRAP contig except for end trimming. Such sequence is assumed to be homozygous. One supercontig built from two PHRAP contigs connected by an N join is also homozygous and has no homologous partner.

Ploidy. Although the assembly was based on the assumption of a diploid genome, the

assembler contained code designed to detect, but not to assemble, sequence apparently present in more than two homologous copies. In Fig. 6, for example, if the alignment between u and v extended much farther to the left, so that it overlapped the s/t join alignment, the assembler forced the portion of the u/v alignment left of position L to be removed during trimming, regardless of sequence quality. Similar rules were applied to inclusions that overlapped joins or other inclusions. Diagnostic messages were issued when trimming occurred for this reason. In our assembly, a total of 2,603 bases were trimmed in this way at 91 separate locations. The number of bases discarded at any one location ranged from 3 to 383, with the majority being less than 10. These few cases are most plausibly viewed as small overlaps between nearby separated homologs, unrecognized as such by PHRAP, rather than as evidence of aneuploidy.

Map conflicts. Early in the project, a large number of genes were identified based on the sequences of individual, unassembled reads. Following the fate of the reads for these genes through the assembly assigned genes to contigs and supercontigs. Because a read is part of one or the other homologous supercontig, such genes were then assigned to both of the supercontigs. In the fully automated assembly, those genes with chromosome assignments were used to locate the supercontigs on the map and yielded a list of eight verified map conflicts. Twenty-one manual directives were created for the final diploid assembly to prohibit joins made in the automated assembly that fell between conflicting map markers. These directives eliminated six of the map conflicts. The two remaining conflicts, involving less than 1% of the assembled genome, derive from PHRAP. Correction of PHRAP's misassemblies is a finishing task that the diploid assembler did not attempt to perform. Fortunately, given the small amount of sequence involved, these conflicts were of minimal importance to the analysis of heterozygosity or *C. albicans* genes. Additional mapped single-copy genes not present in the set of annotated reads mentioned above were used to further check the results and yielded no likely conflicts.

Heterozygous Polymorphism

Detection of heterozygous polymorphisms was performed by using separate but conceptually similar methods in the nearly homozygous and heterozygous regions of the assembly. In both cases, a Bayesian statistical test using base quality scores was performed to distinguish true polymorphisms from sequencing errors. Both tests required an *a priori* per-base rate of polymorphism; we used a rate of 1% for this purpose. This was chosen to be larger than the apparent average rate for the genome as a whole, because of the tendency of polymorphisms to concentrate in some regions.

At each position where we observed a possible disagreement between homolog sequences, a polymorphism was called when the *a posteriori* probability of polymorphism was at least 99%. The choice of the *a priori* rate and the 99% cutoff together determined what level of evidence was needed to support a polymorphism call. Experimenting with cutoffs other than 99% indicated that our general conclusions were not affected by minor changes in the procedure.

Table 7 shows the composition of the polymorphism set obtained by our procedures. The complete set of polymorphisms is available as one of the accompanying data files.

Polymorphism test in nearly homozygous sequence. Detection of polymorphism in nearly homozygous regions relied on PHRAP's alignment of shotgun reads, and proceeded one base position at a time through the padded contig sequence (the sequence shown at the top line in consed). At each position, the set of all shotgun read bases aligned at that position was obtained. At a position with k-fold coverage, this resulted in a set of basecalls $b_1, b_2, \ldots, b_k$ and quality scores $q_1, q_2, \ldots, q_k$. As usual, let $p_i = 10^{-q_i/10}$ be the error probability for b_i. To avoid certain trivialities, any reads for which b_i was not one of the bases `A`, `C`, `G`, `T`, or `*` (denoting a pad in PHRAP's multiple alignment of reads), or with $q_i < 4$, was dropped from the analysis, and we assume that

no such bases occur in the set. Pad bases (*) lacked PHRED quality scores. We assigned pads the minimum of the quality scores of the two nearest nonpad bases before and after the pad. This practice does not have the degree of empirical support available for base quality scores (1,2), but it seemed to work fairly well in practice.

Suppose that at the position we are examining, the genome is homozygous and the correct genomic base is $x \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}, *\}$. Then each basecall b_i can be classified as correct or an error, depending on whether $b_i = x$. We assume that sequencing errors in different reads are statistically independent. Then, given the base qualities, the probability of the observed pattern of correct and incorrect basecalls under the assumption of homozygosity is given by

$$\log P(\text{Obs} \mid \text{Hom}(x)) = \sum_{b_i = x} \log(1 - p_i) + \sum_{b_i \neq x} \log p_i.$$

In practice, we do not know the correct basecall x, but we can easily determine the base x^* that maximizes $P(\text{Obs} \mid \text{Hom}(x^*))$, and we use it in subsequent calculations.

A similar calculation assuming heterozygosity has a few additional complications. If the genome is heterozygous with alleles y and z, $y \neq z$, we classify each basecall b_i as y, z, or an error, depending on whether $b_i = y$, $b_i = z$, or neither. This differs from the homozygous case in that we treat the two possible correct basecalls as distinguishable, but we continue not to distinguish the separate possible erroneous basecalls. We chose this approach on the grounds that erroneous basecalls carry no useful information, but that the relative frequency of two putative alleles is informative, given that they should be equally represented. With this scheme, we compute the probability of observing a y in a given read as $(1/2)(1 - p_i) + (1/2)p_i/3$. The first term is the probability that the read comes from the y allele and that no sequencing error occurs. The second term is the probability that the read comes from the z allele (probability 1/2), that a sequencing error occurs (probability p_i), and that the error converts the z to a y (probability 1/3). The probability of 1/3 was derived by a simplified calculation that excluded pads, and assumed that erroneous basecalls were chosen at random from the three possible wrong calls. Of course pads do occur, but it was not clear how to model them. Furthermore, actual deletion polymorphisms were uncommon, and the results were not very sensitive to this probability, so 1/3 seemed to be an acceptable approximation. Following all this reasoning to its conclusion, we obtain

$$\log P(\text{Obs} \mid \text{Het}(y, z)) = \sum_{b_i \in \{y,z\}} \log(\tfrac{1}{2}(1 - 2p_i/3)) + \sum_{b_i \notin \{y,z\}} \log(2p_i/3).$$

As in the homozygous case, we use the bases y^* and z^* that maximize this probability.

We use these two quantities to make a (approximate, because of the statistical estimation of x^*, y^*, and z^*) Bayesian calculation of the posterior probability of heterozygosity given the observed basecalls:

$$P(\text{Het} \mid \text{Obs}) = \frac{\alpha P(\text{Obs} \mid \text{Het}(y^*, z^*))}{\alpha P(\text{Obs} \mid \text{Het}(y^*, z^*)) + (1 - \alpha) P(\text{Obs} \mid \text{Hom}(x^*))},$$

where α is the prior probability of heterozygosity. For positions where neither y^* nor z^* was a pad (*), we used $\alpha = 0.01$. Otherwise we used $\alpha = 0.002$ to make calls for indel polymorphisms somewhat more conservative (pads being very common in PHRAP's alignments of reads). A polymorphism was called if the posterior probability was greater than or equal to 0.99.

After application of this rule at each consensus position, adjacent inserted or deleted polymorphic bases were grouped together to make single multibase indel polymorphisms. Occasionally PHRAP inserted pads in varying positions in variable-length homopolymer sequence or similar situations. The locations of such polymorphisms were (of necessity) ambiguous, and therefore polymorphic bases of this type may be slightly overcounted in the polymorphism set.

Polymorphisms in PHRAP contigs outside nearly homozygous regions. A total of 22,108 polymorphisms were found in PHRAP contigs by the rule just described. Of these, only three occurred in sequence that was trimmed during the diploid assembly. Another 375 occurred inside joins or inclusions, regions where we believe the homologs were represented by separate PHRAP contigs, so that no polymorphisms within PHRAP contigs should occur. These 375 anomalous polymorphisms did not appear to concentrate anywhere to the extent that they suggested an error in the diploid assembly. The heterozygous assembly regions account for 19% of the diploid genome, but almost 1/3 of the PHRAP assembly, because of duplication. Assuming pessimistically that all polymorphism calls in these regions are errors (some may in fact be correct if PHRAP's separation of homologs is inexact), and extrapolating the same error rate to the whole PHRAP assembly, these numbers give a worst-case estimate of about 95% for accuracy of polymorphism calls within PHRAP contigs, as opposed to the nominal 99% confidence implied by the statistics. Given that the estimate is pessimistic and the statistics involved some approximations, the polymorphism test seems to be performing as intended.

Because the heterozygous regions were examined separately in the polymorphism analysis, we excluded the 378 polymorphisms found within PHRAP contigs in heterozygous and trimmed sequence, leaving 21,730 polymorphisms from nearly homozygous regions to form part of the final polymorphism set.

Polymorphism test in heterozygous sequence. Corresponding heterozygous regions of the diploid assembly (homologs that PHRAP separated) were globally (i.e., end-to-end) aligned by the Smith-Waterman algorithm (5). Any mismatches in the alignment were tested statistically.

The test for polymorphisms in heterozygous sequence is simpler than its counterpart for nearly homozygous sequence, because it makes comparisons between assembled PHRAP sequences for two homologs, rather than large numbers of individual reads. The simplest case is that of a possible single base substitution. Let D be the event that two aligned bases disagree, let the quality scores of the two bases be q_1 and q_2, with error probabilities $p_1 = 10^{-q_1/10}$ and $p_2 = 10^{-q_2/10}$, and let $\alpha = 0.01$ be the *a priori* polymorphism rate. Treating sequencing errors in the two homologs as independent, and assuming that when an error occurs the erroneous base is selected with equal probability from the available alternatives (again relying on approximations based on ignoring pads), we obtain by straightforward computations that the probability of a disagreement due to sequencing errors if the genome is homozygous is

$$P(D \mid \text{Hom}) = p_1(1-p_2) + p_2(1-p_1) + 2p_1p_2/3$$

while the probability of disagreement if the genome is heterozygous is

$$P(D \mid \text{Het}) = (1-p_1)(1-p_2) + 2p_1(1-p_2)/3 + 2(1-p_1)p_2/3 + 7p_1p_2/9.$$

By Bayes' rule, we compute the posterior probability of polymorphism as

$$P(\text{Het} \mid D) = \frac{\alpha P(D \mid \text{Het})}{\alpha P(D \mid \text{Het}) + (1-\alpha)P(D \mid \text{Hom})}.$$

As in the nearly homozygous case, we call a polymorphism if this probability is greater than or equal to 0.99.

For indels in the alignment, deleted bases have no quality scores. As in the nearly homozygous setting, we assigned a deletion a quality score equal to the minimum of the scores of the two bases before and two bases after the deletion. Multibase indels were treated as a single polymorphism, and the quality scores for the inserted bases were summed to obtain a score for the insertion as a

whole (the goal is to obtain a posterior probability for the existence of a polymorphism, not for all inserted bases being correctly sequenced). With this approach, sufficiently large indels are always called as polymorphisms unless sequence quality is extremely poor, usually involving quality scores of zero.

Regions such as the MTL where the two homologs are very diverged are treated in the same way as all other heterozygous regions. This allows for a unified statistical treatment of polymorphism, but the individual polymorphism calls in highly diverged regions can be somewhat arbitrary, depending greatly on details such as the weights used in the alignment.

A total of 40,804 polymorphism calls were made in heterozygous regions.

Other Topics

Cross-species protein comparisons. To perform the protein comparisons reported in the paper, we obtained protein sets for *Schizosaccharomyces pombe* from the Sanger Institute (6), for *Homo sapiens* from the National Center for Biotechnology Information (7), and for *Saccharomyces cerevisiae* from the *Saccharomyces* Genome Database (8). Protein translations of our *C. albicans* reduced haploid ORF set were searched against these three protein sets using NCBI BLASTP with default parameters. Any BLAST hit achieving an E-value of 10^{-8} or better was counted as a match in the discussion in the paper.

Sequence coverage. Early in the project when the emphasis was on gene discovery based on survey sequence, a decision was made to count coverage in terms of "trimmed read length," a locally used method based on trimming low quality bases at the ends of individual reads. For consistency, we continued to use this measure throughout the project, although the coverage figures it produces are too high to give satisfactory results when used in the Lander-Waterman formulas (9) to assess assembly progress. The figure of 10.9× coverage arises from the use of this measure.

According to the more commonly used PHRED20 measure [the count of all bases scored at quality 20 or higher by PHRED (1,2)], our coverage is 7.1×. Our experience with PHRED20 is that it produces coverage figures too low to work well in Lander-Waterman calculations. Some measure lying in between PHRED20 and trimmed length would work best for these calculations. Depending on the necessarily approximate values chosen for other parameters, the Lander-Waterman formulas predict a contig count close to what we actually obtained at approximately 8× coverage.

References

1. Ewing, B., Hillier, L., Wendl, M. C. & Green, P. (1998) *Genome Res.* **8,** 175–185.
2. Ewing, B. & Green, P. (1998) *Genome Res.* **8,** 186–194.
3. Billingsley, P. (1979) *Probability and Measure* (Wiley, New York).
4. Bollobás, B. (1998) *Modern Graph Theory* (Springer, New York).
5. Smith, T. F. & Waterman, M. S. (1981) *J. Mol. Biol.* **147,** 195–197.
6. Sanger Institute, *Schizosaccharomyces pombe* Protein Set, October 2002, `www.sanger.ac.uk/Projects/S_pombe/protein_download.shtml`.
7. National Center for Biotechnology Information, Human Protein Set, January 2003, `ftp://ftp.ncbi.nih.gov/genomes/H_sapiens/protein/protein.fa.gz`.
8. *Saccharomyces* Genome Database, *Saccharomyces cerevisiae* Protein Set, September 2002, `ftp://genome-ftp.stanford.edu/pub/yeast/data_download/sequence/genomic_sequence/orf_protein/orf_trans.fasta.gz`.
9. Lander, E. S. & Waterman, M. S. (1988) *Genomics* **2,** 231–239.