**CNV methods.**

Pipeline URL: <https://github.com/theboocock/long_read_cnv>

**Variant calling.**

**Hybrid CNV algorithm (The long and short read of CNV)**

To identify copy-number variation (CNV) in the 16 yeast parent strains used in the round robin crosses, we developed a computational pipeline designed to leverage both moleculo assemblies and Illumina short-read sequencing (Figure X). The pipeline starts by filtering the moleculo assemblies to only include large scaffolds (10kb). These scaffolds were corrected using PILON, which is a tool that utilizes complementary Illumina short-read sequencing from the same sample to correct for SNPs, fix miss-assemblies, and fill gaps (Walker et al., 2014). Our CNV identification approach consisted of distinct discovery and validation phase.

**Discovery of putative CNVs.**

The filtered scaffolds were aligned to the S288c reference genome using the SpeedSeq align command and were filtered to remove any scaffolds with low alignment quality (MAPQ < 30) (Chiang et al., 2015). To identify putative CNVs, we first applied a modified version of a long-read CNV scripts, which was originally used to generate CNV calls to use as a gold standard for testing the efficiency of the short-read tool LUMPY (Layer, Chiang, Quinlan, & Hall, 2014). Briefly, these scripts parse the CIGAR string from scaffolds that aligned had multiple split alignments to the reference genome. For each CNV, we calculated the number of bases that were accounted for in the alignment. Any CNV with < -50 or > 50 unaccounted for bases were considered complex. We removed any duplications where the predicted size in the reference sequence was > -25 and < 25, which indicates that the CNV was likely an artifact of the alignment process. This process results in a list of base-pair resolved breakpoints for both duplication, deletions, inversions, translocations. This script also identified moderately sized deleted (>500bp) sections at the beginning of the 3’ telomere, and the end of the 5’ telomere. For each of these CNV, we extracted +- 300 base-pairs of sequence surrounding the left and right breakpoints. In addition, we extracted the sequence of each CNV event, for deletions, we extracted the sequence from the reference; and for duplications, we extracted the sequence from the moleculo assembly.

**Validation of CNVs using short-read Illumina data.**

The short-read Illumina datasets were aligned to the S288c reference genome with the SpeedSeq align command. These BAM files were filtered to remove any read with a low alignment quality (MAPQ < 20). For each putative CNV, we extracted all the reads that aligned +- 300 base-pairs from either breakpoint. We identified any reads that should support the CNV call, meaning it is expected to have a split alignment at the breakpoint. For reads with split alignments within 10 bases of the breakpoint, we extracted the raw reads from the FASTQ and aligned it to the breakpoint FASTA using the smith-waterman algorithm. Any alignment with an identity of less than 90% was considered to not support the call. To calculate the quality of each CNV, we modified the genotyping approach used in SVTyper (Chiang et al., 2015). Briefly, the model assumes that given the genotype the conditional distribution of supporting and non-supporting reads is binomially distributed (Equation 1). For the non-destructive events, such as duplications and inversions, the priors were set to be 0.1 for the reference allele and 0.5 for the alternate allele. For disruptive events, such as deletions, the priors were set to be 0.1 and 0.9. Bayes rule was used to calculate the posterior probability of each genotype given the supporting and non-supporting counts. These posterior probabilities were used to derive the quality score, which was the –10 log10 probability of observing an incorrect genotype call given the supporting and non-supporting counts. CNVs with a quality score of less than 50 were removed. We merged the CNVs from each strain by treating each CNV as unique and genotyping the CNV in all strains with the model described above.



Equation 1

**Inversions.**

**Functional analysis of CNVs.**

QTL enrichment –

Functional enrichment –

Compare to the domestic not domestic yeast paper.

http://www.nature.com/ng/journal/vaop/ncurrent/full/ng.3847.html

**Estimating the global allele frequency of all SNPs and INDELs.**

We obtained all publicly available yeast assemblies from the Saccharomyces genome database (SGD, <http://www.yeastgenome.org/sequence/strains/> accessed June 23, 2017). When a variant call format (VCF) file was available for a strain, we used these variants making sure to filter ones with quality score of less than 50. For strains where no VCF file was available on SGD, we first filtered the draft assemblies to remove any contigs that was less than 10kb in length. We called variants by aligning the draft assemblies to the reference genome and comparing the aligned contigs at every position. When two of more overlapping alignments disagreed on a variant call they were discarded.