Bioinformatics, 36(8), 2020, 2569–2571
doi: 10.1093/bioinformatics/btz933
Advance Access Publication Date: 13 December 2019
Applications Note

Genome analysis

HaploTypo: a variant-calling pipeline for phased genomes

Cinta Pegueroles[1,†,‡], Verónica Mixão[1,†,‡], Laia Carreté[1,‡], Manu Molina[1,†] and Toni Gabaldón [1,2,3,*,†]

[1]Centre for Genomic Regulation, The Barcelona Institute of Science and Technology, Barcelona 08003, Spain, [2]Universitat Pompeu Fabra (UPF), Barcelona, Spain and [3]ICREA, Barcelona 08010, Spain

*To whom correspondence should be addressed.
[†]Present address: Barcelona, Supercomputing Center (BSC-CNS) and Institute for Research in Biomedicine (IRB), Barcelona, Spain
[‡]The authors wish it to be known that, in their opinion, the first three authors should be regarded as Joint First Authors.
Associate Editor: Inanc Birol

Received on July 16, 2019; revised on November 28, 2019; editorial decision on December 9, 2019; accepted on December 10, 2019

Abstract

Summary: An increasing number of phased (i.e. with resolved haplotypes) reference genomes are available. However, the most genetic variant calling tools do not explicitly account for haplotype structure. Here, we present HaploTypo, a pipeline tailored to resolve haplotypes in genetic variation analyses. HaploTypo infers the haplotype correspondence for each heterozygous variant called on a phased reference genome.
Availability and implementation: HaploTypo is implemented in Python 2.7 and Python 3.5, and is freely available at https://github.com/gabaldonlab/haplotypo, and as a Docker image.
Contact: toni.gabaldon.bcn@gmail.com
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Motivation

The heterozygosity (i.e. the presence of alternative alleles at the same locus) present in diploid organisms can complicate genome analyses, particularly when the levels of heterozygosity are high. Over the last years, several bioinformatics tools have been developed to account for this sequence complexity. These include pipelines and algorithms to assist during the genome assembly process (Pryszcz and Gabaldón, 2016; Safonova *et al.*, 2015), subsequent phasing of assembled genomes (Chin *et al.*, 2016; Edge *et al.*, 2017; Pan *et al.*, 2014) or allele-specific transcriptomic analysis (Deonovic *et al.*, 2017; Romanel *et al.*, 2015). However, and to the best of our knowledge, available variant calling tools do not explicitly account for phased genomes. As a result, the user has to decide between using the combined phased haplotypes as reference and thereby losing heterozygosity information or, alternatively, using only one of the haplotypes as reference and sacrificing haplotype information. An illustrating example of such problem is studies on the heterozygous yeast pathogen *Candida albicans*. Although the diploid genome of this pathogen was phased in 2013 (Muzzey *et al.*, 2013), subsequent studies have only used one of the haplotypes (Bensasson *et al.*, 2019; Ropars *et al.*, 2018), thereby losing the valuable haplotype information. Given the increasing amount of highly heterozygous genomes, including those from hybrids (Mixão and Gabaldón, 2018), and the relevance of phased information to reconstruct their population structures and evolutionary histories, there is an urgent need for solutions that allow the exploitation of phased genomes in genomic variation analysis. To fill in this gap, we developed HaploTypo, a python-based pipeline that, in the presence of a phased reference genome, provides detailed genome variation resolved at the haplotype level. HaploTypo is not a *de novo* genome phasing tool, but a tool to phase variants in re-sequencing analysis, using information of an already phased genome, resulting in a fast and accurate assessment of heterozygosity levels and reconstruction of haplotypes.

2 Implementation

HaploTypo requires as input the phased haplotypes of a diploid genome, and filtered genomic paired-end sequencing reads or, alternatively, their alignment to each of the reference haplotypes. The pipeline is divided in four modules, which can be run in block or separately (Fig. 1). The first module aligns the genomic paired-end reads independently to each of the phased haplotypes using BWA-MEM (Li, 2013). The second module performs variant calling on the two generated alignments using GATK (McKenna *et al.*, 2010), BCFtools (Li, 2011) or FreeBayes (Garrison and Marth, 2012) followed by variant filtration. From here, variability information is obtained for each reference haplotype independently. The third module of HaploTypo implements a variant phasing algorithm that, based on the comparison of reference haplotypes and previously called variants, infers which variants correspond to each haplotype.

© The Author(s) 2019. Published by Oxford University Press.

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited. For commercial re-use, please contact journals.permissions@oup.com

Downloaded from https://academic.oup.com/bioinformatics/article/36/8/2569/5675499 by guest on 29 February 2024

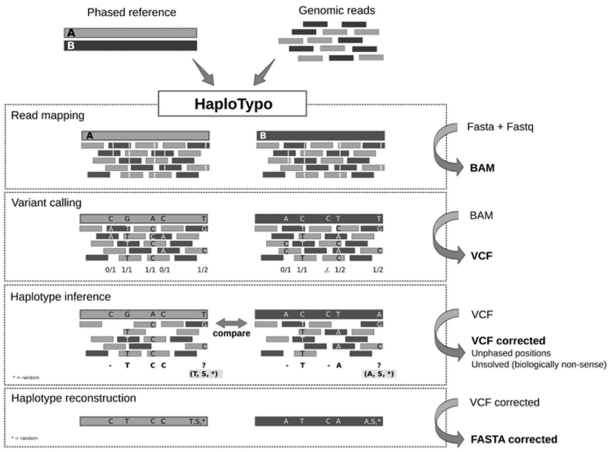

Fig. 1. Schematic representation of the four modules of HaploTypo pipeline. The steps are described in the main text and in the pipeline's manual

Phased (and unphased if required) genotypes from the two phased haplotypes are provided as independent VCF files. Additionally, unphased and unsolved positions for each haplotype are reported as bed files. A final module uses the VCF files generated in module 3 to reconstruct the haplotypes and provide them in fasta format. Detailed information on HaploTypo implementation is available in the pipeline's manual.

3 Validation and results

We validated HaploTypo using simulated phased genome sequences with known variable positions. To explore the influence of divergence between the two phased haplotypes in the downstream analysis, we simulated diploid reference genomes with haplotypes diverging 0.5, 1 or 5% at the nucleotide level. These simulated phased reference genomes were derived with fasta2diverged.py [https://github.com/lpryszcz/bin, (Pryszcz *et al.*, 2015)] from *C.albicans* haplotype A (Muzzey *et al.*, 2013). The same script was used to simulate diploid strains, which differed from the respective reference genome in 1 position per kilo-base, referred to as the 'simple' dataset. Given that, for these simulated strains, most of the polymorphisms are of the type 0/1 (where 0 is reference allele and 1 is alternative allele), we also simulated divergent strains where the polymorphisms between the two haplotypes and the reference could be 0/1 (60% of the total variation), 1/1 (38% of the total variation) or 1/2 (2% of the total variation, where 2 is an alternative allele different from 1), referred to as the 'complex' dataset. The relative proportions of the different variant types were based on real data from *C.albicans* sequenced strains (Ropars *et al.*, 2018). We next simulated sequencing reads using wgsim v0.3.1-r13 (https://github.com/lh3/wgsim). By using these simulated references and reads, we compared the performance of the HaploTypo pipeline to: (i) mapping libraries to only one of the haplotypes (standard procedure) and, (ii) mapping the libraries to the phased genome reference, which combines the two haplotypes (alternative approach). In all cases, we assessed the performance with the three mentioned variant callers.

As expected, when using a haploid reference, sensitivity varied between 96.37 and 99.38%, depending on the variant caller and the divergence between haplotypes (Supplementary Table S1). However, as discussed above, this approach serves to assess heterozygosity levels and the location of heterozygous SNPs, but this information is unphased. When using a diploid reference, the divergence between the two haplotypes highly influenced the outcome, with better results being achieved at higher nucleotide divergences, with sensitivity ranging from 6.5 to 19.3% for 0.5% divergence, from 36.7 to 66.4% for 1% divergence and from 98.3 to 98.8% for 5% divergence (Supplementary Table S1). GATK had the poorest results, specially at low divergence levels (Supplementary Table S1). It is worth noting that accuracy and specificity remained high and stable (>99% and 100% respectively) independently of the levels of divergence and the variant caller used. When using HaploTypo, reads are mapped independently to the two haploid references of a phased genome (approach with the best results, as shown above) and outputs the two haplotypes, correctly phasing >99% of the positions independently of the variant caller used, with few exceptions (Supplementary Table S2). The unphased cases always represented ambiguous situations that cannot possibly be resolved with this type of data (see manual for details), and the user can decide whether to include them in the VCF or not. In addition, HaploTypo also reports positions that have incoherent results in the two haplotypes and therefore are likely to be mapping or variant calling errors (unsolved positions, see Table 1 from the manual for details). HaploTypo benchmarking was performed on a workstation [Intel(R) Xeon(R) CPU E5-1650 v3] and 64 GB of RAM with default number of threads. The total running time ranged from 1 to 15 h, depending on the level of heterozygosity of the dataset and the variant caller (Supplementary Table S3). Hence HaploTypo is a user-friendly tool that eases variant analyses and allows to incorporate haplotype-specific information when a phased reference genome is available.

Acknowledgements

We thank all the members of Gabaldón's lab for the relevant discussions.

Funding

This work was supported by the European Union's Horizon 2020 research and innovation programme under the grant agreement ERC-2016-724173 and Marie Sklodowska-Curie grant agreements N° 642095, and 747607; the Spanish Ministry of Economy, Industry, and Competitiveness (MEIC) for the EMBL partnership, and grants 'Centro de Excelencia Severo Ochoa' SEV-2012-0208, and BFU2015-67107 co-founded by European Regional Development Fund (ERDF); the CERCA Programme/Generalitat de Catalunya; from the Catalan Research Agency (AGAUR) SGR857; and INB Grant (PT17/0009/0023—ISCIII-SGEFI/ERDF).

Conflict of Interest: none declared.

References

Bensasson,D. *et al.* (2019) Diverse lineages of live on old oaks. *Genetics*, **211**, 277–288.

Chin,C.-S. *et al.* (2016) Phased diploid genome assembly with single-molecule real-time sequencing. *Nat. Methods*, **13**, 1050–1054.

Deonovic,B. *et al.* (2017) IDP-ASE: haplotyping and quantifying allele-specific expression at the gene and gene isoform level by hybrid sequencing. *Nucleic Acids Res.*, **45**, e32.

Edge,P. *et al.* (2017) HapCUT2: robust and accurate haplotype assembly for diverse sequencing technologies. *Genome Res.*, **27**, 801–812.

Garrison,E. and Marth,G. (2012) Haplotype-based variant detection from short-read sequencing. *arXiv*, arXiv: 1207.3907.

Li,H. (2013) Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *arXiv*, arXiv: 1303.3997

Li,H. (2011) A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics*, **27**, 2987–2993.

McKenna,A. *et al.* (2010) The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res.*, **20**, 1297–1303.

Mixão,V. and Gabaldón,T. (2018) Hybridization and emergence of virulence in opportunistic human yeast pathogens. *Yeast*, **35**, 5–20.

Muzzey,D. *et al.* (2013) Assembly of a phased diploid *Candida albicans* genome facilitates allele-specific measurements and provides a simple model for repeat and indel structure. *Genome Biol.*, **14**, R97.

Downloaded from https://academic.oup.com/bioinformatics/article/36/8/2569/5675499 by guest on 29 February 2024

Pan,W. *et al*. (2014) WinHAP2: an extremely fast haplotype phasing program for long genotype sequences. *BMC Bioinformatics*, **15**, 164.
Pryszcz,L.P. *et al*. (2015) The genomic aftermath of hybridization in the opportunistic pathogen *Candida metapsilosis*. *PLoS Genet*., **11**, e1005626.
Pryszcz,L.P. and Gabaldón,T. (2016) Redundans: an assembly pipeline for highly heterozygous genomes. *Nucleic Acids Res*., **44**, e113.
Romanel,A. *et al*. (2015) ASEQ: fast allele-specific studies from next-generation sequencing data. *BMC Med. Genomics*, **8**, 9.
Ropars,J. *et al*. (2018) Gene flow contributes to diversification of the major fungal pathogen *Candida albicans*. *Nat. Commun*., **9**, 2253.
Safonova,Y. *et al*. (2015) dipSPAdes: assembler for highly polymorphic diploid genomes. *J. Comput. Biol*., **22**, 528–545.

Downloaded from https://academic.oup.com/bioinformatics/article/36/8/2569/5675499 by guest on 29 February 2024