IMPUTE 5

v1.2.0 - Aug 17, 2023

Introduction

IMPUTE 5 is a genotype imputation method that can scale to reference panels with millions of samples. This method continues to refine the observation made in the IMPUTE2 method, that accuracy is optimized via use of a custom subset of haplotypes when imputing each individual. It achieves fast, accurate, and memory-efficient imputation by selecting haplotypes using the Positional Burrows Wheeler Transform (PBWT). By using the PBWT data structure at genotyped markers, IMPUTE 5 identifies locally best matching haplotypes and long identical by state segments. The method then uses the selected haplotypes as conditioning states within the IMPUTE model.

Imp5Chunker is a program to create imputation chunks from the target and reference panel.

xcftools is a program to convert VCF/BCF reference panels in XCF file format. xcf is a file format used by IMPUTE 5 to store reference panels and allows fast read of custom regions, without the need to use compression libraries like ZLIB, and faster imputation. XCF is a open-source and refined version of the previous imp5 format, described in the IMPUTE5 manuscript.

Citation:

If you use IMPUTE 5 in your research, please cite the following publication:

Rubinacci S, Delaneau O, Marchini J (2020) Genotype imputation using the Positional Burrows Wheeler Transform. PLOS Genetics 16(11): e1009049

Documentation

Example files to test IMPUTE 5 and xcftools can be found in the *test* directory. Examples shown below assume impute5 binary is called from the appropriate directory, when not explicitly noted.

1. Overview

1.1. Running software of the IMPUTE5 suite

IMPUTE5 is available as a static binary compiled in Ubuntu x64 system. To run the program, simply run:

```
./impute5_v1.2.0_static --help
```

Please note that the name might change between versions. Please use the latest version of the software as it contains the latest bug fixes.

In the case the binary does not work on your operating system, you can use docker to run IMPUTE5. For example you can run:

```
#pull an ubuntu image
[sudo] docker pull ubuntu
#run the docker
[sudo] docker run -it --rm --name impute5 -v
<impute_folder>:/home/impute5 ubuntu
```

From the resulting terming (indicated as “/ #”) you can run IMPUTE5 by simply using:

```
home/impute5/impute5_v1.2.0
```

And run the test example by using:

```
home/impute5/impute5_v1.2.0 --h /home/impute5/test/reference.bcf --
g /home/impute5/test/target.bcf --m
/home/impute5/test/chr20.b37.gmap.gz --o home/impute5/test/imputed.bcf
--r 20:1000000-4000000
```

1.2. Imputation workflow

IMPUTE5 does not perform automatic chunking and ligation. The reason is to allow each job to run completely independently on large clusters. The typical IMPUTE5 workflow is composed by considering each chromosome independently. The following are the steps of a typical IMPUTE5 run:

1. **Chunking step**: each chromosome is divided into a small set of chunks (overlapping in the buffer regions). (Section 2)

2. *Optional but recommended*: **convert the reference panel and SNP array data in XCF** file format (section 3)

3. **Imputation** of each chunk of data: running IMPUTE5 to impute each chunk of data independently (section 4)

4. **Ligation**: ligate each chunk of imputed data together in order to create one file per chromosome. To perform this, two alternatives are possible:

    1. If the SNP array data has been phased on the whole chromsome, the ligation step reduces to a simple concatation of files. It is possible to use *bcftools concat -n* for this task, allowing very fast concatenation without recompression. (Section 5)

    2. If the SNP array data has been phased in chunks, and not ligated at the chromosome level, a legation has to be performed. For this, impute5 is required to run using the --out-buffer option, allowing to output in outpu phased SNP array data in the buffer regions. At the legation step, a program such as *GLIMPSE2_ligate* or *bcftools concat --ligate* needs to be run. The two programs are very similar and can likely be used indifferently. However, *bcftools concat –ligate* keeps the whole buffer region in memory and could be memory inefficient.

2. Chunking step

2.1. Simple run

Imp5Chunker is a software that takes two datasets (target and reference panel) to create a file containing the regions to be used for imputation.

Three main parameters are needed to run imp5Chunker:

```
imp5Chunker --h reference.bcf --g target.bcf --r 20 --o coordinates.txt
```

Where --h defines the reference panel (does not have to contain the GT field), --g defines the target and --r region of interested (usually a full chromosome) and --o the output file (a text file).

The output has the following fields:

```
Chunk ID / chromosome ID / Buffered region / Imputation region / Length / Number of target markers / Number of reference markers
```

Other parameters define the minumum chunk and buffer size and counts. The parameter `--max-window-count` forces the minumum number of genotype markers to be fixed.

2.2. Option summary

The full list of options can be obtained by running the command:

```
imp5Chunker --help
```

This should output this list of options:

Input

Option		Default value	Description
--h	STRING	-	Haplotype reference panel in VCF/BCF/XCF format (must have .vcf[.gz]/.bcf extension). The file must be indexed (tabix index). For efficiency reasons, better if only positions are defined (no GT field)
--g	STRING	-	File containing target haplotypes for a study cohort that you want to impute in VCF/BCF format (must have .vcf[.gz]/.bcf extension). The file must be indexed (tabix index).
--r	STRING	-	Region containing the whole region (typically

			a whole chromosome). Example -r 20 (whole chromosome 20).

Parameters

Option	Type	Default value	Description
--window-size	INT	5000000	Minimum Window size in bp
--buffer-size	INT	250000	Minimum buffer size in bp
--window-count	INT	-1	Minimal number of genotyped markers in the chunk (shared with the reference panel). Default means that the parameter is set to the expected number of chip markers in a chunk (#variants_shared / (length_region_shared / window_size))
--buffer-count	INT	-1	Minimal number of genotyped markers in the chunk (shared with the reference panel). Default means that the parameter is set to the expected number of chip markers in a buffer region (#variants_shared / (length_region_shared / buffer_size))
--max-window-count	INT	-1	Minimal number of genotyped markers in the chunk (shared with the reference panel). Default is set to 2*window-count

Output

Option	Type	Default value	Description
--o	STRING	-	Specifies output file name.

Other parameters

Option	Type	Default value	Description
--help	NA	-	Produces help message, listing all the accepted arguments
--l	STRING	-	Location of the log file to be written. If not specified, only console output will be generated.

2.3. Alternative: bcftools scatter

The Mocha imputation pipeline (https://github.com/freeseek/mocha/blob/master/wdl/impute.wdl) adopts an alternative to perform the chunking step by using the bcftools scatter plugin.
More information is in the wdl file provided by the Mocha pipeline (https://github.com/freeseek/mocha/blob/master/wdl/impute.wdl), more details about the bcftools scatter can be found by running:

```
bcftools +scatter
```

Running bcftools plugins requires BCFTOOLS_PLUGINS variable to be in PATH. More details here: `https://samtools.github.io/bcftools/howtos/plugins.html`

3. Reference panel format: xcftools (replaces deprecated imp5 format)

xcftools converts a dataset in VCF/BCF format to the xcf file format. A xcf file format contains a set of files with genotype information on a region within a chromosome. Typically we want to create a xcf file for each chromosome in order to perform imputation on different chunks of the chromosome. A xcf file is also complemented by an index (.bcf.csi), that allows IMPUTE 5 to have random access to the region of the chromosome.

IMPUTE5 greatly benefits of this file format to speed-up the process of imputation, and it is recommended for the reference panel file. Additionally, an xcf file can also be used for the target panel. While the efficient read of the target panel is less performance critical compared to the reference panel, importantly SHAPEIT5_phase_common allows the use of the xcf file format in output, making the pre-phasing step very easily integrated in the

New versions of the format are updated with the versioning of IMPUTE5: i**mp5 file format cannot be read by IMPUTE5 v1.2.0 or newer**. Please create a new xcf file using the latest imp5Converter version.

3.1. Simple run

To convert the full reference panel chromosome 20 to an xcf file using 8 threads (-T8) you can simply use:

```
xcftools view -i /reference.bcf -o /reference_xcf.bcf -O sh -r 20 -T8 -m 0.03125
```

The output is a site file named reference_xcf**.bcf** and an index file named reference_xcf.bcf**.csi** a genotype file reference_xcf**.bin** and a pedegree file reference_xcf**.fam**. We can now call IMPUTE 5 and pass the xcf file (for convenience the site file reference_xcf**.bcf**) as a reference panel (IMPUTE 5's --h option), and IMPUTE5 will automatically read the .csi, .fam and .bin.

Similarly a file can be converted from XCF to VCF/BCF format using:

```
xcftools view -i /reference_xcf.bcf --r 20 --o reference.bcf -O bcf -T8
```

Optionally, it is possible to convert the target panel in the xcf file format as well. For this, we use a different encoding (**-O bh**). Similarly to the reference panel option, IMPUTE5 will automatically detect if a .bcf site file is encoded in xcf file format.

3.2. Option summary

The full list of options can be obtained by running the command:

```
xcftools view --help
```

This should output this list of options:

Input Mode: view

Option		Default value	Description
--i	STRING	-	Input file in VCF/BCF format or XCF format. The file must be indexed.
--r	STRING	-	Region containing the whole imputation region (typically a whole chromosome). Example -r 20 (whole chromosome 20).

Output

Option	Type	Default value	Description
--o	STRING	-	Specifies output file name (must have .vcf[.gz]/.bcf extension).
-O	STRING	-	Output encoding. Accepted values: • **sh** (sparse haplotypes) for reference panel files. • **bh** (binary haplotypes) for SNP array phased data • **bcf** to convert XCF back to BCF.
-m	FLOAT	0.001	Sparse MAF threshold to use for the **-O sh** option. For reference panels used with IMPUTE5, please always use **-m `0.03125`**`, representing the optimal value in SNP array imputation settings.`

Other parameters

Option	Type	Default value	Description
--help	NA	-	Produces help message, listing all the accepted arguments
--T	INT	1	Number of threads used for compression/decompression.

4. IMPUTE5

4.1. Simple run

To run IMPUTE5 with default parameters, use the following command line:

```
impute5 --h reference_xcf.bcf --m chr20.b37.gmap.gz --g target.bcf --r
20:2000000-7000000 --o imputed.bcf
```

All five options are mandatory and their descriptions are:

- `--h` specifies the haplotype reference panel in VCF/BCF/XCF format (must have .vcf[.gz]/.bcf extension). The file must be indexed. The reference panel should be phased and non-missing at every position.
- `--m` specifies the fine-scale recombination map for the region to be analysed. Maps for humans can be found HERE. In the case this parameter is not defined, a constant recombination rate is assumed.
- `--g` specifies the file containing target haplotypes for a study cohort that you want to impute in VCF/BCF format (must have .vcf[.gz]/.bcf extension). The file must be indexed (tabix index). The target dataset should be phased and non-missing in the set of markers specified. Markers that are only present in the reference panel and not in the target set, are imputed.
- `--r` specifies the target region or chromosome to be imputed . Buffer parameters will expand this region, if specified.
- `--o` specifies the output filename. A proper extension is mandatory.

IMPUTE5 considers as genotype markers the markers in the intersection between `--g` and `--h`. In practice, it considers as genotype markers only the variants with chromosome ID, position, REF and ALT alleles that perfectly match between the two panels. Markers only in the reference panel are considered imputed markers and markers present only in the target panel are simply reported in output.

4.2. Log file

To record all the verbose that appear on the screen, use the `--l` option as follows:

```
impute5 --h reference_xcf.bcf --m chr20.b37.gmap.gz --g target.bcf --r
20:2000000-7000000 --o imputed.bcf --l imputed.log
```

The use of this parameter is strongly recommended.

4.3. Imputing a chunk of data

To impute the 5Mb region located in the genomic interval 2Mb-7Mb, use:

```
impute5 --h reference_xcf.bcf --m chr20.b37.gmap.gz --g target.bcf --r 20:2000000-7000000 --o imputed.bcf
```

`--r` option is mandatory. Double check that the chromosome ID matches one of those specified in the VCF/BCF file. A common mistake is to use other specifications for the chromosome different from the one specified in the VCF/BCF file. A quick way to check it would be running:

```
bcftools view -H -G target.bcf | head -n 1 | awk  '{print $1}'
```

Also, please verify that your reference and target panel present the same notation for the chromosome.

The definition of the imputation regions depends of the dataset. Typically, imputation regions ~5 Mb should be enough for the majority of applications. Increasing the imputation region could bring additional benefits at rare variants, at the cost of an increased running time.

Each chunk of imputed data is expanded by a buffer region, in order to help preventing imputation quality from deteriorating near the edges of the region. Markers in the buffer region will help the inference but do not appear in the output files. Larger buffers can improve accuracy. Value of the buffer regions can be expressed as follows:

- using the **--buffer-region**, defining a region to be used as buffer, that expands the region defined with the --r parameter:

  ```
  impute5 --h reference_xcf.bcf --m chr20.b37.gmap.gz --g target.bcf --r 20:2000000-7000000 --o imputed.bcf --buffer-region 20:1500000-7500000
  ```

4.4. Output file format

IMPUTE5 file automatically detects the format of the input and output file by the extension. Input can be specified in three different file format: VCF[.gz]/BCF/XCF. Output can be specified in three file formats: .**vcf**[.gz]/.**bcf**/.**bgen**.

4.5.1 BGEN output

You can choose the compression used by BGEN for the output file format using --bgen-compr parameter (values accepted: no,zlib,zstd).

For example, to output a BGEN file compressed using ZSTD you will use:

```
impute5 --h reference_xcf.bcf --m chr20.b37.gmap.gz --g target.bcf --r 20:2000000-7000000 --bgen-compr zstd --o imputed.bgen
```

to output a BGEN file compressed using ZLIB you will use:

```
impute5 --h reference_xcf.bcf --m chr20.b37.gmap.gz --g target.bcf --r 20:2000000-7000000 --bgen-compr zlib --o imputed.bgen
```

to output a BGEN file compressed with no compression you will use:

```
impute5 --h reference_xcf.bcf --m chr20.b37.gmap.gz --g target.bcf --r 20:2000000-7000000 --bgen-compr no --o imputed.bgen
```

to output a **phased** BGEN file compressed using ZSTD you will use:

```
impute5 --h reference_xcf.bcf --m chr20.b37.gmap.gz --g target.bcf --r 20:2000000-7000000 --bgen-compr zstd --o imputed.bgen --out-ap-field
```

4.5.2 VCF/BCF output

VCF and BCF file format contains phased genotypes in the GT field. At imputed markers the VCF/INFO field will contain:

- IMP flag, denoting that the marker is imputed;
- INFO field: containing the IMPUTE INFO score at the variant
- AF field: containing the estimated allele frequency

The VCF/FORMAT by default has:

- GT, containing the most likely genotype;
- DS, containing the genotype dosage. This can be suppressed using the -–no-out-ds-field parameter.
- GP, genotype probrabilities. This can be suppressed using the --no-out-gp-field parameter.

And finally its index, generated automatically by IMPUTE5

- `imputed.bcf.csi`. This can be suppressed using the `--no-out-index` parameter.

The `--out-ap-field` can be used to output ALT haplotype probabilities in the FORMAT/AP field.

4.6. Parallelization

4.6.1 Parallelization by chunk

IMPUTE5 parallelise by chunks so that different imputation regions can be imputed at the same time on a different process.

To do this, you just need to run a IMPUTE5 job per imputation region, by exploiting the `--r` parameter, for example running the following tow commands in parallel:

```
impute5 --h reference.bcf --m chr20.b37.gmap.gz --g target.bcf --r 20:2000001-7000000 --o imputed.00.bcf
```

```
impute5 --h reference.bcf --m chr20.b37.gmap.gz --g target.bcf --r 20:7000001-12000000 --o imputed.01.bcf
```

4.6.2 Multi-threaded parallelization

A single chunk can also be multi-threaded. Multi-threading is only performed on parts of the algorithm (e.g. HMM calculations), therefore is not as efficient as parallelization by chunk.

Multi-threaded parallelization imputes in parallel several individuals, therefore is useful is the number of target samples is large.

To run impute5 on a chunk in parallel, run:

```
impute5 --h reference.bcf --m chr20.b37.gmap.gz --g target.bcf --r 20:2000000-7000000 --o imputed.bcf --threads 4
```

This will run imputation for the chunk using four threads.

4.8 Chromosome X imputation

Since IMPUTE5 v1.1.4 it is possible to perform haploid imputation. For chromosome X imputation in the non-PAR region, it is required to split the target samples by sex, as females are diploids and males are haploid. For females standard diploid imputation is performed. For males, haploid imputation must be perfomed, therefore it is possible to inform IMPUTE5 using the --haploid option:

```
impute5 --h reference_xcf.bcf --m chr20.b37.gmap.gz --g target.haploids.bcf --r 20:2000000-7000000 --o imputed.haploids.bcf --haploid
```

4.9. Other options

IMPUTE5 implements the surfbat model and outputs haploid dosages for surrogate family based association testing. Details about the caluculations are provided the following manuscript:

SURFBAT: a surrogate family-based association test building on large imputation reference panels. AF Herzig et al. BioRxiv. doi: https://doi.org/10.1101/2023.01.10.523404

Please check the --surfbat , --surfbat-maf and --surfbat-info options for more details.

4.10. Option summary

The full list of options can be obtained by running the command:

```
impute5 --help
```

This should output this list of options:

Input

Option		Default value	Description
--h	STRING	-	Haplotype reference panel in VCF/BCF/XCF format (must have .vcf[.gz]/.bcf extension). The file must be indexed.
--m	STRING	-	Fine-scale recombination map for the region to be analyzed. If not specified, a constant recombination rate of 1cM per Mb is used.
--g	STRING	-	File containing target haplotypes for a study cohort that you want to impute in VCF/BCF format (must have .vcf[.gz]/.bcf extension). The file must be indexed (tabix index).
--r	STRING	-	Region to be imputed (replaces IMPUTE4's -int parameter). Example -r 20:1000000-5000000 (region within chromosome 20). Buffer parameters will expand this region, if specified.
--buffer-region	STRING	-	Length of buffer region (in kb) to include on each side of the analysis window specified by the -r option. Variants in the buffer regions inform the inference but do not appear in output files. If both the buffer options are used (--b and --buffer-region) only --buffer-region is used.

--sparse-maf	FLOAT	0.03125	(Expert setting) Rare variant threshold
--haploid	-	-	Specifies the target samples are haploid in the region (e.g. for males in non-PAR region on Chromosome X).

Model parameters

Option	**Type**	**Default value**	**Description**
--ne	FLOAT	100000	Effective population size.
--err-imp	FLOAT	1e-4	(Expert setting) Imputation HMM error rate

State selection

Option	**Type**	**Default value**	**Description**
--max-pbwt-depth	INT	16	Max depth of PBWT indexes to condition on
--min-pbwt-depth	INT	2	Min depth of PBWT indexes to condition on
--pbwt-cm	FLOAT	0.02	Frequency of the selection algorithm in cM
--Kpbwt	INT	1500	Max number of states to condition on
--neigh-select	NA	NA	Use only the positional prefix array to select states

Test statistics

Option	**Type**	**Default value**	**Description**
--surfbat	STRING	paired	Outputs FORMAT/SAP field (standard haploid dosages and surrogate family haploid dosages for surrogate family based association testing) and paired p-value derived from chisquare distribution with one degree of freedom. Use only if if the SNP array data has been phased on the whole chromosome as not supported during ligation.
--surfbat-maf	FLOAT	0.001	Surfbat p-value restricted to sites with MAF above specified parameter
--surfbat-info	FLOAT	0.3	Surfbat p-value restricted to sites with INFO above specified parameter

Output

Option	Type	Default value	Description
--l	STRING	-	Location of the log file to be written. If not specified, only console output will be generated.
--o	STRING	impute5.out.bcf	Specifies output file name. Accepted extensions: [.vcf[.gz],.bcf,.bgen]. If the format is in BGEN format, by default the file in unphased. To output a phased BGEN file, specify the --out-ap-field option.
--bgen-compr	STRING	zstd	Specifies the compression of the output file for BGEN file format (to be used with --o *.bgen). Accepted values: [no, zlib, zstd]
--bgen-bits	INT	8	Specifies the number of bits to be used for the encoding probabilites of the output BGEN file (to be used with --o *.bgen). Accepted values: 0< x <=32.
--no-out-index	NA	-	Skip computaton of CSI index if output is in VCF/BCF format
--no-out-ds-field	NA	-	Do not output FORMAT/DS field (Genotype probabilities) if output is in VCF/BCF format.
--no-out-gp-field	NA	-	Do not output FORMAT/GP field (Genotype probabilities) if output is in VCF/BCF format.
--out-ap-field	NA	-	Print FORMAT/AP field (ALT haplotype probabilities) if output is in VCF/BCF format.
--out-buffer	NA	-	Output SNP array variants in the buffer (necessary if the SNP array data has been phased in chunks rather than on the whole chromosome.

Other parameters

Option	Type	Default value	Description
--help	NA	-	Produces help message, listing all the accepted arguments

--threads	INT	1	Number of threads
--contigs-fai	NA	-	If specified, header contig names and their lengths are copied from the provided fasta index file (.fai) instead of being taken from the target panel (default behavior). This allows to create files with all the contigs in the header (in the case the contigs in the reference panel are limited to a single chromosome) and therefore quickly merge chromosome-level files with bcftools merge --naive
--estimate-mem-usage	NA	-	Experimental. Estimates the dynamic memory usage.
--seed	INT	42	Seed for RNG

5. Ligation step

5.1 Array phased on the whole chromosome

The simplest way to ligate imputed chunks back is using bcftools concat providing the list of files in the right order:

```
bcftools concat -n -f list.txt -Ob -o ligated.bcf
```

More details about bcftools concat can be found in the here: http://samtools.github.io/bcftools/bcftools.html#concat

In the case your output is in **BGEN** file format, you can use the cat-bgen format: https://enkre.net/cgi-bin/code/bgen/wiki/cat-bgen

5.2 Array phased in chunks (not recommended)

The simplest way to ligate imputed chunks back is using GLIMPSE2_ligate or bcftools concat --ligate. **Please note that IMPUTE5 had to be run using the --out-buffer parameter**, so that there are shared variants between chunks to determine the appropriate phasing

```
GLIMPSE2_ligate --input list.txt --output ligated.bcf --region 20
```

```
or
```

```
bcftools concat --ligate -f list.txt -Ob -o ligated.bcf
```

Contact

Please email srubinac@broadinstitute.org or join the OXSTATGEN mailing list to post any question

https://www.jiscmail.ac.uk/cgi-bin/webadmin?A0=OXSTATGEN