

Documentation for fastPHASE 1.2

Algorithm by Paul Scheet* and Matthew Stephens

13 November, 2006

Contents

1	Introduction	3
1.1	What is new in this version?	3
2	Getting started	4
2.1	Installation	4
2.2	Usage	4
3	Input file format	5
4	Output	7
5	Available options	8
5.1	Input & output options (-n, -o)	8
5.2	Controlling the algorithm (-T, -C, -H, -i)	8
5.3	Determination of number of clusters (-K[...])	9
5.4	Utilizing known haplotypes (-b, -B)	10
5.5	Incorporation of subpopulation labels (-u)	11
5.6	Sampling and simulating (-s, -U)	11
5.7	Scanning for genotype errors (-e[...])	12
5.8	Estimating haplotype frequencies (-F)	16
5.9	Bracketing uncertain genotypes in output (-q<number>)	16
5.10	Options not intended for general use	16

*for correspondence: email: PSCHEET@ALUM.WUSTL.EDU

6	Regarding the analysis of large amounts of data	18
7	How to cite this program	19
8	Acknowledgements	19
9	Obtaining the software	20

1 Introduction

fastPHASE is software that implements methods for estimating missing genotypes and reconstructing haplotypes from unphased SNP genotype data of unrelated individuals. The methods are based on a cluster model for haplotypes, which is described in Scheet and Stephens (2006). Parameters of the model are first estimated with an EM algorithm; then conditional on these parameters, missing genotypes and haplotypes are inferred.

Additionally, the software is capable of performing the following tasks:

1. scanning the data for SNPs with elevated signal for genotyping errors (methods described in a forthcoming manuscript),
2. sampling haplotypes conditional on the observed genotypes,
3. simulating haplotypes *unconditional* on the observed data but using the observed genotypes to first fit a parametric model, and
4. estimating sample haplotype frequencies.

Please check that you have the latest version of the software by visiting

<http://depts.washington.edu/fphase>

1.1 What is new in this version?

- Methods for detecting genotype errors are available (section 5.7).
- Defaults are slightly changed to accommodate larger data sets.
- Added `-q<number>` option to put brackets in output around genotypes for which the posterior probability of the most-likely genotype is less than `<number>`.

2 Getting started

2.1 Installation

Presumably, you have downloaded a `fastphase.VERSION.linux.tar.gz` file. (Alternatively, you downloaded an executable file for Solaris, Darwin¹, or for Microsoft Windows².) Then do the following:

```
gunzip fastphase.VERSION.linux.tar.gz
tar -xvf fastphase.VERSION.linux.tar
```

This will produce a folder similar in name to `fastphase.VERSION.linux`. Contained in this new directory are the following:

- `fastPHASE`
- `fastphase.inp` an example input file
- `fastphase_haplotypes.inp` an example input file of known haplotypes
- `fastphase_subpoplabels.inp` file of “subpopulation” labels for individuals in `fastphase.inp`
- `fastphase_subpoplabels2.inp` contains enough labels to run with `-b -ufastphase_subpoplabels2.inp` flags.

To do a short run and test that the program is working on your system:

```
./fastPHASE -T1
```

2.2 Usage

You may be reminded of some available options from the command line by running

¹Darwin is the Mac OS X flavor of UNIX. To use fastPHASE on a Mac, you’ll need to open a “terminal” session by launching the Terminal application. This is typically in **Applications** → **Utilities**.

²To use fastPHASE on a PC running Microsoft Windows, launch a MS-DOS window by going to **Start** → **Run** and type “cmd”. You may have to find another way to unzip and untar any compressed files you downloaded.

```
./fastPHASE -h
```

which will give the current usage.

Please note that **there should be no spaces between an option flag and any number or filename** that follows. In the following (for example)

```
./fastPHASE -T10 -ulabels.txt -oMyresults test.inp
```

I'm specifying that I'd like only 10 starts/runs of the EM algorithm (instead of 20). Additionally, here I've chosen to use *Myresults* as a prefix for the main output files that fastPHASE creates. Finally, I'm supplying a file *labels.txt* which contains subpopulation labels for the individuals in my input file *test.inp*.

3 Input file format

The input filename may be supplied by the user after any options; otherwise, fastPHASE looks for a file in the current directory named *fastphase.inp*. Information required in the input file includes: the number of diploid individuals to be analyzed, the number of SNP sites in the data, and the genotypes for each individual. Optionally, the file may specify a label for each individual (in fact, this is assumed). Additionally, the file may contain the relative physical positions of the SNP markers and may contain a line of 'S' characters. The physical positions may be utilized in conjunction with a different model option. The S-line is allowed only for compatibility with PHASE input files, as fastPHASE works only with SNP data in its current implementation.

An individual's unphased genotypes are to be provided on 2 lines using any characters other than the question mark (?), with ? denoting missing alleles. Spaces and tabs may separate the allele characters. The input file may be represented as follows:

```
no.individuals
no.SNPsites
P pos(1) pos(2) ... pos(no.SNPsites) <optional line>
SSS...SSS <optional line>
ID (1)
genotypes(1-a)
genotypes(1-b)
```

```

ID (2)
genotypes(2-a)
genotypes(2-b)
.
.
.
ID (no.individuals)
genotypes(no.individuals-a)
genotypes(no.individuals-b)

```

where the quantities are the following:

- `no.individuals` An integer specifying the number of individuals who have been genotyped.
- `no.SNPsites` An integer specifying the number of SNP sites at which each individual has been typed.
- `P` The character P (upper case).
- `pos(i)` A number indicating the position of site i . The sites must be in their physical order along the chromosome (i.e. these positions must be increasing). With the default settings, this information is ignored, and so this “P-line” may be omitted.
- `SSS...` A line of `S` characters (optional).
- `ID(i)` A character string, i.e. label, for individual i . If you do not wish to specify a label for each individual, omit them and use the `-n` option.
- `genotypes(i-a)` and `genotypes(i-b)` Genotypes for individual i .

For example, consider this small example input file, `test.inp`:

```

3
4
# id 1
1a11
0t01
# id 2
1t11

```

```
0a00
# id 3
?a01
?t10
```

A few comments:

- In the above example there are 3 individuals typed at 4 SNP sites. The first individual is heterozygous at 3 sites and homozygous for 1 at the last site, the second individual is heterozygous at all 4 sites, and the third individual contains missing data at the first site.
- Above, the second SNP is coded with `a/t`, but for each SNP site, any 2 characters may be used for the two SNP alleles.
- When giving the genotypes, you may place spaces between adjacent alleles, e.g.

```
? t 1 0
```

would be acceptable for the last line.

- fastPHASE does not explicitly or comprehensively check the format of the input file; therefore, it is recommended that you verify that the output (in the `_genotypes.out` or `_hapguess.switch.out` files) is consistent with your input data.

4 Output

Output files for inferred haplotypes or imputed genotypes contain two lines per given diploid individual, with the order of individuals corresponding to that supplied in the input file. In addition, summary information is given, such as a recapitulation of some of the parameters from the fastPHASE run, and the command line supplied by the user. The `_switch.out` file contains estimates which attempt to minimize the *switch* error, and the `_indiv.out` file contains estimates which attempt to minimize *individual* error (see Stephens and Donnelly, 2003, for a review of these error measures). If haplotypes are not estimated (by supplying a negative integer following `-H`), fastPHASE creates a `_genotypes.out` file, containing the original data and estimated unphased genotypes which were denoted as missing in the input file.

5 Available options

You may be reminded of available options by supplying the `-h` option.

5.1 Input & output options (`-n`, `-o`)

- `-n` *id lines omitted in input file.* The input file consists of only 2 lines per diploid individual (in addition to the “header” information such as the number of individuals, number of SNPs, etc.).
- `-o` *specifying output file prefix.* For example, `-oMyresults` would result in the output file: `Myresults_hapguess_switch.out`

5.2 Controlling the algorithm (`-T`, `-C`, `-H`, `-i`)

- `-T<number>` *number of random starts of the EM algorithm.* The default is 20, although 10 starts produced results close to those from 20 in our tests (in half the time, of course).
- `-C<number>` *the no. of iterations of the EM algorithm.* For our tests, the default of 25 was sufficient. However, you may see a small gain in accuracy from more iterations on larger datasets (I have only limited data on this). This value has worked well for data from 60 unrelated individuals typed at 42,000 SNPs.
- `-H<number>` *set the number of haplotypes sampled from the “posterior” distribution obtained from a particular random start of the EM algorithm.* Since the default number of EM runs is 20, `-H50` would produce 1000 samples of consecutive 2-site haplotypes, which we have observed to be sufficient for minimizing switch error. If you wish to minimize individual error (less relevant for datasets with many SNPs), you may want to set this value higher (e.g. 200). If your only interest is inferring missing genotypes, you can turn off haplotype estimation with `-H-4` (or some negative integer), to save computation time.
- `-i` *estimate haplotypes by minimizing individual error.* If you specify the `-i` option, an additional file is printed for estimated haplotypes: `fastphase_hapguess_indiv.out`, where “fastphase” may be replaced with text supplied after the `-o` flag.

5.3 Determination of number of clusters (-K[...])

The default action for fastPHASE is to determine the number of haplotype clusters via a cross-validation procedure (see Scheet and Stephens, 2006). **This default is new in version 1.1 and for smaller data sets (containing at most 500 SNP loci) this add considerably to the running time!** To turn this off and select the default from previous versions (which was 10), use -K10. **Use of the -K option followed by an integer takes precedence over the -K options below which control the cross-validation algorithm.** The number of clusters selected by the procedure is given in the output file (unless the -Z option is given to simplify the output). The default range of 5, 10, 15 may not include large enough values for complex data sets, with large numbers of individuals, or for data with large amounts of haplotypic diversity. The user should be aware of this, and set appropriate limits using the controls below.

The cross-validation procedure consists of searching over a range of values for the number of clusters K . To accomplish this, fastPHASE applies missingness (masks data) to a portion of the observed data and, for several values of K , makes a best-guess for the missing genotypes. This process is repeated multiple times, each time choosing a different portion of the observed data (all individuals but only up to a certain number of SNP loci), and the value of K is chosen which produced the lowest overall error rate.

- -Ku<number> or -KU<number> *upper limit for no. of clusters.* Largest value of K considered during cross-validation.
- -Kl<number> or -KL<number> *lower limit for no. of clusters.* Smallest value of K considered during cross-validation.
- -Ki<number> or -KI<number> *interval between values for number of clusters.* This controls the difference between values of K considered for cross-validation. For example, if the lower and upper limits for K are 5 and 14, respectively, the -Ki3 option would require fastPHASE to consider values 5, 8, 11, and 14.
- -Ks<number> or -KS<number> *no. of times masking process applied.* Default³ is 10.

³With this and all defaults, use the -h (help) option to verify default values. Default values printed in this documentation may not reflect actual values implemented in fastPHASE.

- `-Km<number>` or `-KM<number>` *no. of SNP loci used for cross-validation.* Default is 500. If data consist of fewer than 500 sites, all will be used.
- `-Kp<float>` or `-KP<float>` *rate at which missingness (masking) applied.* Default is .1.

An example usage for these options is:

```
-KL6 -KU12 -Ki2 -Ks50 -Km1000 -Kp.05
```

which would tell fastPHASE to do the following 50 times: randomly select at most 1000 consecutive SNPs; mask approximately 5% of the observed genotypes among all individuals at 1000 SNPs; consider 6, 8, 10, and 12 for the number of clusters; impute missing genotypes at each value of K and tabulate errors.

Results from this K -selection process are written to a file with a `_kselect` extension. This file includes the parameters from the procedure, as well as a list of values of K considered and the corresponding number of genotypes imputation errors (and error rate).

5.4 Utilizing known haplotypes (`-b`, `-B`)

Known haplotypes may be supplied in a separate file if the `-b<filename>` flag is specified. *Only entire haplotypes may be specified – this is NOT equivalent to the `-k` flag in PHASE.* The input file for known haplotypes should be as follows:

```
no.haplotypes
ID hap 1
haplotypes(1)
ID hap 2
haplotypes(2)
.
.
.
haplotypes(no.haplotypes)
```

where the above quantities are the following:

- `no.haplotypes` Integer specifying the number of haplotypes.
- `haplotypes(i)` Alleles for haplotype i , consisting of `no.SNPsites` characters that correspond to the same characters used in the main input file.

Please note that this file does not contain its own line for the number of SNP sites. This is assumed to be the same as for the main input file for unphased genotypes.

If all of the genetic data is of haplotypes (i.e. all of data is phase-known) and you want to estimate parameters and simulate or estimate missing alleles, you may notify fastPHASE that your main data file contains haplotype data with the `-B` flag. The haplotypes may be listed one after the other (without “id labels”) in conjunction with the `-n` option, or grouped into “pseudo-individuals” with one “id line” separating pairs of haplotypes.

5.5 Incorporation of subpopulation labels (-u)

Subpopulation labels may be supplied with the `-u<filename>` flag. The information must be supplied on *one line* with different integers for different labels. (All values okay except `-9999`.) Here are example contents of such a file, which would correspond to the `test.inp` file above.

```
5 2 5
```

This specifies that the first and third individuals were sampled from the same subpopulation. *If known haplotypes are supplied, fastPHASE expects additional labels, one for each haplotype.* So if labels are provided for the analysis of 3 diploid genotypes and 2 haplotypes, there should be 5 integers on one line in the labels file.

5.6 Sampling and simulating (-s, -U)

- `-s<number>` *sample haplotypes given observed unphased genotypes.* Instead of producing point estimates of haplotypes or genotypes, you may wish to obtain multiple samples of the haplotypes (and genotypes where missing values were observed), given the observed data. Using this option will produce `<number>` diplotypes (phased haplotypes) for each diploid individual in the sample, $\frac{\text{<number>}}{\text{no. starts (-T)}}$ per random start

of the EM (see section 5.2). Additionally, if there are (phase-known) haplotypes in your sample, `<number>` haplotypes will be output. These haplotypes are printed to a file with a `_sampledHgivG.txt` extension. For each sample from the posterior distribution, the haplotypes are given in the same order as the corresponding unphased genotypes in the input file.

- `-U<number>` *sample haplotypes **unconditional** on observed data.* The model is fit (for each run of EM) and `<number>` haplotypes are simulated from the fitted model, unconditional on the observed genotype data.

5.7 Scanning for genotype errors (`-e[...]`)

The `-e` option tells fastPHASE that the true genotypes have been observed with some error; that is, there may be particular SNPs at which an elevated level of genotyping error exists. This may be due to actual errors in the genotype calls, or possibly due to the segregation of copy-number polymorphisms, such as a deletion, in the population at some SNPs.

This method is in development but the basic idea is straightforward. fastPHASE attempts to infer the true (unobserved) genotypes, given the observed genotypes and some error rate. The error rate, along with other parameters of the model (which inform fastPHASE as to the expected levels of linkage disequilibrium; LD) are estimated in the EM algorithm.

Then, conditional on these estimates and the observed data, statistics may be computed at certain SNP markers to summarize the evidence for genotyping errors at these marker loci. In particular, the two main statistics are a *likelihood ratio* (LR) and the total number of expected errors. Other statistics are presented, as well, to allow flexibility for researchers wishing to investigate evidence for errors in a range of data sets and computational demands.

Since the types of data to which these methods may be applied are potentially quite varied, it is difficult to set appropriate defaults. At this time, we recommend an initial scan of the distribution of estimated total expected error values across the SNPs. These values are output in the file described below (`_errhat.txt`). Based on these values, the scanning criteria may be altered with the `-et<number>` option (below).

The basic switch `-e` turns on this modelling process. Additional arguments are available and can be called with letters *immediately following* the `-e`. These are given below, in the form they should be entered on the command line.

- `-em<letter>` error model

- `-ema` *allelic* 2-parameter error model (default). This default error model is recommended for general use. However, we give its form, along with several other models, in case users want to experiment with their data. Here is the default “allelic” error model:

		<i>g_{OBS}</i>			
		AA	AT	TT	
<i>g_{TRUE}</i>	AA	$1 - \epsilon$	ϵ	0	where g_{OBS} are the observed
	AT	$\frac{\epsilon^*}{2}$	$1 - \epsilon^*$	$\frac{\epsilon^*}{2}$	
	TT	0	ϵ	$1 - \epsilon$,	

genotype data and g_{TRUE} are the true (unobserved) genotypes, and ϵ and ϵ^* are error parameters.

- `-emb` *simple* 2-parameter error model:

		<i>g_{OBS}</i>			
		AA	AT	TT	
<i>g_{TRUE}</i>	AA	$1 - \epsilon$	$\frac{\epsilon}{2}$	$\frac{\epsilon}{2}$	
	AT	$\frac{\epsilon^*}{2}$	$1 - \epsilon^*$	$\frac{\epsilon^*}{2}$	
	TT	$\frac{\epsilon}{2}$	$\frac{\epsilon}{2}$	$1 - \epsilon$,	

- `-ems` *simple* error model:

		<i>g_{OBS}</i>			
		AA	AT	TT	
<i>g_{TRUE}</i>	AA	$1 - \epsilon$	$\frac{\epsilon}{2}$	$\frac{\epsilon}{2}$	
	AT	$\frac{\epsilon}{2}$	$1 - \epsilon$	$\frac{\epsilon}{2}$	
	TT	$\frac{\epsilon}{2}$	$\frac{\epsilon}{2}$	$1 - \epsilon$,	

- `-eo` Only estimate error parameters and calculate total expected errors.

(Do not evaluate each SNP by calculating LR statistics.) Relevant output will be contained in the `_errhat.txt` file (see below).

- **-et<number>** Threshold for scanning criteria: total expected number of errors. If the total expected error for a SNP exceeds **<number>**, then LR-based criteria (see below) are evaluated at this SNP. (This criterion is evaluated separately for each EM run.)
- **-ex<number>** Threshold for scanning criteria: maximum expected error from *one* individual among *all* individuals. If the maximum expected error for a SNP exceeds **<number>**, then LR-based criteria (see below) are evaluated at this SNP. (Evaluated separately for each EM run.)
- **-es<filename>** Read SNP sites to evaluate from a file. File format must be as follows: two lines with the first line containing the number of SNPs to evaluate, and the second containing the SNP indices (counting from 1) with spaces between them. This overrides the **-ex** and **-et** threshold criteria above and evaluates each SNP in the file during each run of the EM.

The output from a scan for genotype errors are contained in 2 files with the suffixes `_errhat.txt` and `_errscan.txt`. The first (`_errhat.txt`) contains a summary of the error model as well as the total expected number of errors *for each SNP in the data set*. The second (`_errscan.txt`) contains a more detailed description of the evidence for genotype errors for only those SNPs which showed putative evidence of error (based on the above criteria) for at least one run of the EM (or for those SNPs listed in a supplied file with the **-es<filename>** option).

Here is a description of the output (column headings in bold) for the `_errhat.txt` file.

1. **SNP.index** Index of the SNP (counting from 1).
2. **phys.pos** Physical position of the SNP (if given in the input file).
3. **pr.hom2hom** Estimated homozygote “fidelity”, i.e. the probability that a major (minor) homozygote is observed, given the true genotype is a major (minor) homozygote.
4. **pr.het2het** Estimated heterozygote “fidelity”, i.e. the probability that a heterozygote is observed, given the true genotype is a heterozygote.

5. **total.expected.errors** Estimated total expected number of errors.

Here is a description of the output (column headings) for the `_errscan.txt` file.

1. **SNP.index** Index of the SNP (counting from 1).
2. **phys.pos** Physical position of the SNP (if given in the input file).
3. **log(LR).avg**⁴ This is the sum of the $\log_e(LR)$ over those starts for which at least one of the screening criteria for evaluating this SNP was met (see above), divided by *the total number of EM starts*. Essentially, for computational convenience, we make the assumption that the LR is 1 for runs (EM starts) which did not present evidence of error at this SNP based on the screening criteria. This will underestimate the true average of the $\log(LR)$ value.
4. **log(LR).max** The natural logarithm of the *largest* LR encountered in the T random starts of the EM algorithm.
5. **log(LR).avg** Similar in spirit to **log(LR).avg**; however, here the average is taken on the probability ratio (LR), and presented as the natural log.
6. **total.expected.errors.avg** The expected error for each individual is used for screening which SNPs to evaluate in the scan for errors. *This is the total expected number of errors* across all individuals, averaged over all the EM runs. This value tends to be high at SNPs with moderate amounts of genotyping error, especially at sites with a CNP or deletion which may affect the genotype calls for a larger number of individuals.
7. **max.error.from.1ind** The maximum expected error from *one* individual among *all* individuals for the EM start corresponding to the largest likelihood (in some sense the “most likely” model configuration).
8. **prop.error.from.1ind** The proportion of the total expected error (for this EM start) attributable to the individual who contributed the largest value of expected error.

⁴Due to computational techniques which *approximate* the true LR, its value can be less than 1, thus the $\log(LR)$ can be less than 0.

9. **individ.contrib.max.error** The individual (counting from 1) with the largest contribution to `total.expected.errors.avg` in this “most likely” EM start/run. (These last two or three criteria can help identify one individual with a different genetic background from other members of the population or from whom poor quality samples were obtained.)

5.8 Estimating haplotype frequencies (-F)

Sample haplotype frequencies may be estimated by Monte Carlo methods with `-F`. Samples from the posterior distribution of haplotypes given the observed genotypes are drawn for each run of the EM algorithm. After all runs are completed, the sampled haplotypes are tabulated and frequency estimates are produced. Haplotypes are sampled from the observed genotypes 5,000 times per diploid individual. To increase this number (for more precise estimates), supply a larger number immediately after `-F`.

Haplotypes with corresponding estimated frequencies are output to a file with the `_freqs` suffix. If subpopulation labels are supplied with the `-u` option, the `_freqs` file will contain separate haplotype frequency estimates for each subpopulation, as well as estimates assuming a single panmictic population.

5.9 Bracketing uncertain genotypes in output (-q<number>)

As in PHASE, using the `-q<number>` option displays brackets in output around genotypes for which the posterior probability of the most-likely genotype is less than `<number>`. The default is 0, so that no brackets are produced.

5.10 Options not intended for general use

Many of these options are not well documented, and some may not be well-tested, especially when used in certain combinations with each other. Please email pscheet@alum.wustl.edu with any questions.

- `-Z` *print simplified output format*. Produces a simple output, with 2 lines per individual, without “id” lines, subpopulation labels or summary information from the run.

- **-M<number>** *modelling options*. The default model is to allow values for α and r to vary across the chromosome or genomic region (see Scheet and Stephens, 2006). Alternatively, these parameters can be assumed constant across the chromosome. That is, $\alpha_s = \alpha_t$ and $r_s = r_t$ (for all $s, t = 1, \dots, M$). To specify a “constant α , constant r ” model, use **-M1**; for “constant α only”, use **-M2**; and for “constant r only”, use **-M3**. **If using models which assume constant values of r or α , supply the relative physical positions of SNPs via the P line** (see Section 3). During the averaging of results over multiple starts of the EM algorithm, haplotype and genotype inference results may be averaged over different models. To use a “constant α , constant r ” model for half the runs, use **-M41**; to use “constant α only” for half the runs, use **-M42**.
- **-g** *turn off genotype imputation*. If you wish to infer phase only (not actually infer the missing genotypes), or some specific simulation tasks.
- **-H<negative-number>** *turn off haplotype inference*. You can turn off haplotype estimation with **-H-4** (or some negative integer), to save computation time, if your only interest is inferring unphased genotypes. (If used in combination with **-g**, no main output file will be produced; only summary/ extraneous files.)
- **-S<number>** *set the seed for random number generation*. The value of **<number>** should be a positive integer. If not supplied, seed will be taken from the system clock.
- **-S<negative-number>** *read parameter values from file*. If a negative integer is supplied, e.g. **-S-7**, the parameter values are initiated by reading the values from files. These filenames are “hard-wired” to be **<input-prefix>** with the following suffixes appended: **_alphahat.txt**, **_thetahat.txt**, **_rhat.txt**. For example, if you had files named

run35_thetahat.txt, **run35_alphahat.txt**, and **run35_rhat.txt**,

you would supply “run35” as the input-prefix parameter with the **-I** (capital *i* for input) option, i.e. **-Irun35**

- `-Pp` (or currently `-p` as well) *print estimated parameter values*. Can be used to capture parameter values obtained at the end of the final run of the EM algorithm. The filenames produced are the same as above, for use with `-S<negative-number>`.
- `-Pzp` *print cluster membership probabilities* for all SNPs between and including `begin-SNP` and `end-SNP` (see below). These are the posterior probabilities that an individual derives its genotypes at a particular SNP from a particular pair of haplotype clusters (see Scheet and Stephens, 2006). Because this output may be large⁵, it is broken up into multiple files, one for each individual (with extension `._<individual.number>`.) The format for each file is

Individual SNP `cluster.prob(1,1)` `cluster.prob(K,K)`

and the order of the cluster pairs is as follows:

$\{1, 1\}, \{1, 2\}, \dots, \{1, K\}, \{2, 2\}, \{2, 3\}, \dots, \{2, K\}, \{3, 3\}, \dots, \{3, K\}, \dots, \{K-1, K-1\}, \{K-1, K\}, \{K, K\}$.

- `-Pzb<number>` *begin-SNP*. First SNP (1-indexed) to be included in the output for `-Pzp`.
- `-Pze<number>` *end-SNP*. Last SNP (1-indexed) to be included in the output for `-Pzp`. (Both `-Pzb<number>` and `-Pze<number>` must be supplied or probabilities will be printed for all SNPs.)

6 Regarding the analysis of large amounts of data

There are no built-in limits to the amount of data which can be analyzed. The algorithm is linear in both the number of individuals and number of SNPs. Some considerations are:

⁵This option should be used in conjunction with `-T1`. The probabilities are printed for each random start but with the same filename and so will overwrite results from earlier starts with the same output-prefix (see `-o<name>` option).

- The maximum number of characters per line which can be read by fastPHASE is currently set to 500,000. So if you have data at a very large number of sites, you may break these up over multiple lines. You should do so in the same fashion as you would with one line, only adding a few breaks where necessary. That is, initially assign phase arbitrarily, giving one “haplotype” on multiple lines and then the other haplotype.
- Since with hundreds of thousands of loci on even a hundred individuals, fitting the model for multiple starts will take perhaps a day or two. To reduce computation time, you might consider reducing the number of starts to 5 or 10, e.g. with `-T10`.
- Depending on the data, you might consider reducing the value of K . We have found `-K8` to be suitable under a variety of scenarios; however, this was often with data consisting of at most a few hundred individuals. The K -selection procedure chooses for the number of clusters, the value which produced the lowest error rate. In practice, these error rates may be very similar, in which case it would be prudent to choose the smallest value of K which still seems reasonable. You may first run the K -selection procedure with:

`-KL6 -KU14 -Ki2`

for example, and then hit `CTRL-c` to kill the process after it chooses a value of K . Then, you can examine the output in the `_Kselect.txt` file and see if a smaller value of K (than the maximum examined) is suitable based on the genotype imputation error. This value could then be set manually with `-K<number>`.

7 How to cite this program

In publications which use results from the use of fastPHASE, please cite Scheet and Stephens (2006).

8 Acknowledgements

The executable for Microsoft Windows was created with DJGPP, which is available from: <http://www.delorie.com/djgpp/>

9 Obtaining the software

fastPHASE is available for download from:

<http://www.stat.washington.edu/stephens/software.html>

or

<http://depts.washington.edu/fphase>

References

Scheet, P. and M. Stephens (2006). A fast and flexible statistical model for large-scale population genotype data: Applications to inferring missing genotypes and haplotypic phase. *American Journal of Human Genetics* 78, 629–644.

Stephens, M. and P. Donnelly (2003). A comparison of Bayesian methods for haplotype reconstruction from population genotype data. *American Journal of Human Genetics* 73(5), 1162–1169.