COMPASS[1]:

COalescence siMulation Program Allowing Serial Samples

Version 1.0.1

Mattias Jakobsson[2]
Department of Evolutionary Biology
Uppsala University

March 17, 2010

The *COMPASS* software is available at
`http://www.egs.uu.se/evbiol/Research/JakobssonLab/compass.html`[3]

[1]Jakobsson, M (2009) *COMPASS*: a program for generating serial samples under an infinite sites model. *Bioinformatics* 25: 2845-2847.

[2]Comments on *COMPASS* can be sent to mattias.jakobsson@ebc.uu.se

Contents

1 Introduction

The program *COMPASS* can generate samples that have been collected at various points in time. The samples are generated using coalescence simulations permitting various demographic scenarios. *COMPASS* uses an infinite sites model (Kimura, 1969) to generate polymorphism data for the samples. By generating serially sampled population-genetic data, *COMPASS* provide means of investigating the properties of data that has been collected at different time-points, and potentially, aid in interpreting the results from empirical data, for example polymorphism data collected from both present-day samples and historic samples.

Ancient DNA can be useful to detect and date population dynamics and patterns and processes in the history of populations and species (see e.g. Willerslev & Cooper, 2005; Hofreiter, 2008). The the number of ancient DNA studies have been increasing over the last few years and spectacular data from for example Neanderthals, cave bear and Mammoth have been generated (Krings *et al.*, 1997; Noonan *et al.*, 2006, 2005; Miller *et al.*, 2008).

Rodrigo & Felsenstein (1999) extend the standard coalescent model by considering serially sampled gene copies. The idea of serial samples have been exploited in the *BEAST* software (Drummond *et al.*, 2002) to estimate demographic parameters of populations or species using data from multiple time-points. Another software that simulates data from serial samples is *Serial SimCoal* by Anderson *et al.* (2005). These two softwares have different primary aims – estimation of demographic parameters and simulating data (although simulated data can be used for inference too) – and both programs have different limitations and strengths (see e. g. Anderson *et al.*, 2005). However, none of the the programs above use an infinite sites model (Kimura, 1969), and in some circumstances the infinite sites model may be appropriate and/or straightforward to use, for example, when comparing simulated data to population-genetic SNP data. Regardless of model details, simulations of serially sampled population-genetic data can provide means of better understanding the population-genetic properties of data from multiple time-points and it may also be used as part of an analysis framework.

The program *COMPASS* generates samples (and polymorphism data assuming an infinite sites model, under a coalescent model allowing serially sampled gene copies and permitting various demographic scenarios. *COMPASS* can generate many independent replicate samples under various assumptions about sample times, populations sizes and population size changes. The samples are generated using standard coalescent approaches where the random genealogy of the sample is first generated, followed by randomly "dropping" mutations to the genealogy (Kingman, 1982; Hudson, 1990; Nordborg, 2001; Hein *et al.*, 2005). An infinite sites model (Kimura, 1969) is assumed so that every mutation give rise to a new variable site. To allow serial samples, the basic genealogy-generating algorithm have been modified to allow "new" lineages to be added when the sample-times are passed (backwards in time).

The output from the program *COMPASS* is very similar to the output from the program

ms by Hudson (2002). This similarity will (hopefully) minimize the need to transform output from *COMPASS* to fit programs and scripts written to handle output of *ms*. Users familiar with *ms* users will also find that options in *COMPASS* behave in the same way as they do in *ms*. A program note (Jakobsson, 2009) describing *COMPASS* was published in *Bioinformatics*.

2 Getting started

Pre-compiled executables for *COMPASS* to run under Unix/Linux, MacOS X and Windows are available from the *COMPASS* website. The program is written in C++, and some functions from the GNU Scientific Library are used (a free library under the GNU General Public Licence, `http://www.gnu.org/software/gsl/`), including the random number generator. The *COMPASS* source code is available on request.

2.1 Availability

Pre-compiled executables for Linux/Unix, MacOS X, and Windows are available at:
`http://www.egs.uu.se/evbiol/Research/JakobssonLab/compass.html`
When using *COMPASS*, please cite:

Jakobsson, M. (2009). *COMPASS*: a program for generating serial samples under an infinite sites model. *Bioinformatics* 25: 2845-2847.

2.2 Installing *COMPASS*

The executable comes in a gzipped tar file. In Unix/Linux (and in MacOS X, from a command prompt), extract the appropriate `.tar.gz` file by typing: `tar -xvzf COMPASS_Linux.xx.tar.gz`, where `xx` is the version number. This will create a new directory called `COMPASS_Linux.xx`.

In Windows, extract the file `COMPASS_Windows.xx.zip`. This will create a directory called `COMPASS_Windows.xx`.

2.3 Running *COMPASS*

In Unix (and in MacOS X, from a command prompt), the program is executed by typing `./COMPASS` followed by a number of command line arguments (described in sections 3 and 4). If no arguments are given after typing `./COMPASS`, the program will provide a usage summary.

In Windows, *COMPASS* is run from a command prompt. In the command prompt (which can be accessed by going to the START menu, clicking on Run, and typing `cmd`), move to the directory where *COMPASS* is located (by typing `cd c:Program Files\COMPASS_Windows.1.0` on our machine). The program is then executed by typing `COMPASS` followed by a number of command line arguments (described in sections 3 and 4). If no arguments are given after typing `COMPASS`, the program will provide a usage summary.

2.4 The basic command line

To run *COMPASS*, a few arguments are always needed. The simplest usage of *COMPASS* is:

```
COMPASS numbSamples numbReps -t theta -h t numSerSamples
```

where **numbSamples** is the total number of gene copies in the sample, and **numbReps** is the number of independent replicate samples to generate (note that we omit "`./`" for simplicity, to run *COMPASS* on a UNIX/LINUX/MacOS X machine, continue to use "`./`" or use an alias). The third argument **-t** is the option for setting the mutation rate $\theta = 4N_0u$ to **theta**, where N_0 is the diploid population size and where u is the mutation rate for the entire locus. The fifth argument **-h** is the option for setting the sample-time **time** and the number of sampled gene copies **numbSerSamples** from that particular time . The two first arguments (**numbSamples** and **numbReps**) are required and must appear in the given order. Following these two arguments, at least one of the the options **-t theta**, **-s segSites** or **-T** must be given, where the **-s** option fixes the number of sites to **segSites**, and the **-T** option will make *COMPASS* print the gene tree. The **-h** option is needed, at least once, for *COMPASS* to run. If the **-h** option is given only once, **numSerSamples** must equal **numSamples**. Except for the two first options (**numbSamples** and **numbReps**), options can appear in any order.

2.5 Random number generator

COMPASS uses the Tausworthe random number generator (function "gsl_rng_taus" in the Gnu Scientific Library; L'Ecuyer, 1996). The program looks for the file "**seedcompass**", containing a single integer, to seed the random number generator. If no file named "**seedcompass**" is found, *COMPASS* will use the system time to seed the random number generator. In all cases, the seeds are printed on the second line of the output. If one desires to generate the exact same simulation and set of samples, one simply replaces the integer in the the "**seedcompass**" file with the seed from the output. When *COMPASS* finishes a simulation, it will print the last random number to the file "**seedcompass**" (and if no such file exists, *COMPASS* creates one).

3 Usage

In this section, the usage of *COMPASS* is described. All options available for *COMPASS* behave in the same way as in the program *ms* (Hudson, 2002), except the **-h** option that is unique to *COMPASS*.

3.1 Historic samples

To generate samples that have been sampled at some specific time-point, the **-h** option is used. For example,

```
COMPASS 10 1 -s 1 -h 0.0 5 -h 1.0 5
```

will generate one sample with exactly one segregating site, of 5 gene copies sampled at present, and 5 gene copies sampled 1 unit of time before present (measured in $4N_0$ generations). In the output, the samples closest to the present (time 0) will be printed first (one line for each gene copy), followed by increasingly older samples. Thus, the input order on the command line will not affect the output order of the samples. In the output from the example above, the first five samples represent the sampled gene copies from the present (time 0) and the following five samples represent the sampled gene copies from 1 unit of time before present.

Note that the sum of all samples (both historic samples and samples from the present) must equal the total number of samples (**numbSamples**), which is indicated by the the first argument following the program name on the command line. Thus, at least one serial sample is always needed to run *COMPASS*.

3.2 Mutation rate

To set the mutation rate, the **-t** option is used. The **-t** flag shall be followed by the scaled mutation rate **theta** ($\theta = 4N_0u$) for the entire locus, where N_0 is the diploid population size and where u is the mutation rate per generation for the entire locus. For example, the following command

```
COMPASS 10 2 -t 4.0 -h 0.0 10
```

will generate two independent samples, each consisting of 10 gene copies sampled at present (time 0) and with the mutation rate θ set to 4.0.

3.3 Fixed number of sites

To generate samples with a fixed number of segregating sites, the **-s** option is used, followed by an integer (**segSites**) that indicates the number of segregating sites. By using this option,

the (fixed) number of sites are added randomly to each generated genealogy. Note that this procedure is not equivalent to generating samples assuming some value of θ and conditioning on a fixed number of segregating sites.

If both the **-t** option and the **-s** option is given, *COMPASS* will print an extra line for each sample after the `segsites:` line. This line begins with `prob:` followed by the probability of observing exactly the specified number of segregating sited given the gene tree and the specified value of θ.

3.4 Instantaneous population size change

To change the population size instantaneously, the **-eN** command is used. The flag shall be followed by the time of the size change, **t**, and the new population size, **size**, relative to the initial population size N_0 so that the new size is **size**$\times N_0$. The growth rate is set to zero at the time **t**. For example, the following command

```
COMPASS 10 1 -t 4.0 -h 0.0 10 -eN 1.0 .1
```

will generate one sample of 10 gene copies sampled at present (time 0), with the mutation rate θ set to 4.0, and with the population size being set to a tenth of the initial size one unit of time before present.

3.5 Exponential growth

The option **-eG** is used to specify exponential population growth (or shrinkage). The flag shall be followed by the time **t** when starts to grow/shrink (backwards in time) and the rate of change **alpha**. The population size is given by $N(t) = N_0 \exp^{-\alpha t}$, where t is the time before present measured in units of $4N_0$ generations, and α is the (exponential) growth rate. The following command

```
COMPASS 10 1 -t 4.0 -h 0.0 10 -eG 1.0 .2
```

will generate one sample of 10 gene copies sampled at present (time 0), with the mutation rate θ set to 4.0, and with the population shrinking at a rate 0.2 backwards in time, starting from one unit of time before present.

A negative value of α means that the population has been shrinking forwards in time (and growing backwards in time). Note that a negative α quickly leads to a very large population. The user must prevent the potential situation where the population size becomes so large than no coalescent event ever occur by additional demographic events in the past.

3.6 Printing gene trees

To print the gene tree for the sampled gene copies, the **-T** option is used. The following example will print the gene tree of 10 gene copies sampled at present:

```
COMPASS 10 1 -T -h 0.0 10
```

The resulting gene tree, printed in Newick format, represents the history of the 10 gene copies. The branch lengths are measured in units of $4N_0$ generations, and the sampled gene copies are labeled from 1 to **numbSamples**. The labeling within a serial sample is arbitrary, but the labeling has meaning across serial samples (see Section 3.1). The gene trees output by *COMPASS* can be used by other programs to generate other types of data. For example, sequence data can be generated using the program *seq-gen* by Rambaut & Grassly (1997), see the manual of *ms* for an explicit example on how to use the gene tree information to generate sequence data.

3.7 Output

The output from *COMPASS* is almost identical to the output from *ms*; the only difference is that one random number is given on the second line (instead of three for *ms*). For example, the following command:

```
COMPASS 10 1 -s 3 -h 0.0 5 -h 1.0 5
```

will produce the output:

```
COMPASS 10 1 -s 3 -h 0.0 5 -h 1.0 5
3042483524

//
segsites: 3

positions: 0.596281 0.657415 0.903938
100
100
101
100
100
100
100
100
010
100
```

(the exact random number may be different). The first line shows the command used to generate the sample. The second line outputs the random number seed. Following those two lines, each sample is printed after a line with "`//`" (when the tbs argument is used, numbers will appear after the "`//`", see section 3.9). For each sample, the "`//`"-line is followed by a line containing the text `segsites:` followed by the number of segregating sites. If the gene tree is to be printed (see section 3.6), it would appear directly after the "`//`"-line and before the "`segsites:`"-line. Following the "`segsites:`"-line is a line with the text `positions:`, which is followed by the positions of each site on a 0 to 1 scale (excluding the endpoints). The positions are randomly drawn from a uniform distribution. Finally, the haplotypes for each sampled gene copy are printed as 0 or 1. The ancestral state is 0 and the derived state is 1. When a sample has no polymorphic sites, both the "`positions:`"-line and the haplotypes are omitted.

3.8 Discarding low-frequency sites

With the option **-F** the user can discard all sites that have fewer than some number of copies (**minFreq**). The value of **minFreq** shall be an integer and it must be at least 2 and less than half the total number of sampled gene copies.

3.9 The "tbs" argument

The **tbs** ("to be specified") argument permits the user to give each replicate sample a different set of parameter values. On the command line, type **tbs** for each parameter value that should be specified later. For each replicate sample, the program will read the parameters from `stdin`. Most numerical parameters can be set with the **tbs** argument, except **numbSamples**, **numbReps**, and the number of sites specified by the **-s** option. For example, if the file `thetaAndSampleTime` contains the following two lines:

```
2.0 1.0
3.0 1.5
```

the following command:

```
COMPASS 7 2 -t tbs -h 0.0 3 -h tbs 4 < thetaAndSampleTime
```

would generate the output below:

```
COMPASS 7 2 -t tbs -h 0.0 3 -h tbs 4
3126284070

// 2.0 1.0
segsites: 5

positions:  0.233388 0.516891 0.63312 0.764503 0.957429
01011
01010
01010
01000
01000
10000
10100

// 3.0 1.5
segsites: 7

positions:  0.0411697 0.217953 0.423548 0.511322 0.52281 0.834232
0.911647
0111100
0111011
0111100
0000000
0000000
0000000
1000000
```

The first sample was generated with $\theta = 2.0$, three gene copies sampled at present, and four gene copies sampled at 1.0 time units before present. The second sample was generated with $\theta = 3.0$, three gene copies sampled at present, and four gene copies sampled 1.5 time units before present. The parameter values read from the file `thetaAndSampleTime` are printed after the "`//`" for each replicate. The values are read sequentially from `stdin`. If the number of imported arguments runs out before the number of replicate samples (**numbReps**) have been produced, the program will exit. The **-f** option cannot be used with the **tbs** arguments.

4 Summary of command line options

COMPASS reads the arguments provided by the user at the command line. *COMPASS* can also read most arguments from a file, see the **-f** option below.

-f filename (string)
Read arguments from file **filename** (maximum length of 99 characters). The arguments given at the command line (both before and after the **-f** argument) are also used by *COMPASS*. The **-f** argument can be given any where after the two initial arguments, the number of samples, **numbSamples**, and the number of replicates, **numbReps**.

-t theta (real)
Sets the value of $\theta = 4N_0u$ to **theta**.

-s segSites (int)
Sets the number of segregating sites to **segSites**. If both the **-s** option and the **-t** option are given, *COMPASS* will output the probability of exactly **segSites** given θ.

-T (void)
Print gene trees (in Newick format). Either the **-s** option, the **-t** option, or the **-T** option is needed to run *COMPASS*.

-eG t alpha (real, real)
Sets the growth rate to **alpha** at time **t**.

-eN t size (real, real)
Sets the population size to **size**$\times N_0$ at time **t**.

-F minFreq (integer)
Print only sites with the minor allele frequency of at least **minFreq**. Note that **minFreq** is in $[2, \text{floor}(\textbf{numbSamples}/2)]$.

-h t numbSerSamples (real, integer)
This option adds **numbSerSamples** samples at time **t**. Note that at least one serial sample is needed (set **t** to zero to indicate "the present"). Note also that the sum of all samples (added via the **-h** option) must equal **numbSamples** (the first argument given at the command line).

5 Examples

5.1 Multiple historic samples and constant population size

This example shows a model of constant population size where 5 samples (of varying number of gene copies) have been taken at different points in time, including at present (fig. 1). The following command would generate 10 replicates from the model and the particular sampling scheme:

```
COMPASS 38 10 -t 2.0 -h 0.0 18 -h 0.5 5 -h 1.0 7 -h 1.5 4 -h 2.0 4
```

where $\theta = 2.0$, 18 gene copies are sampled at present, 5 gene copies are sampled $2.0N_0$ generations before present, 7 gene copies are sampled $4.0N_0$ generations before present, 4 gene copies are sampled $6.0N_0$ generations before present, and 4 gene copies are sampled $8.0N_0$ generations before present. In the output, the first 18 haplotypes represent the sample at present, the following 5 haplotypes represent the sample $2.0N_0$ generations before present, and so on.

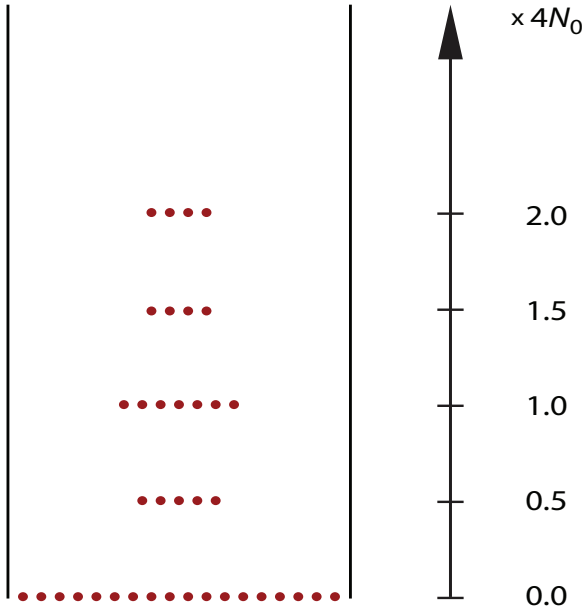

Figure 1: An example of a population with constant size and where five samples of 18, 5, 7, 4, and 4 gene-copies (shown as red filled circles) have been taken at 5 different points in time (0.0, 0.5, 1.0, 1.5, and 2.0 time units ago).

5.2 Two sample-times, one before and one after a bottleneck

The following example shows a scenario where two samples have been taken from a population, one sample of 15 gene copies is taken at present and one sample of 10 gene copies is taken $4.0N_0$ generations before present (fig. 2). The population has gone through a bottleneck – during this time the population size was a tenth of its current size – from $0.8N_0$ generations ago, to $3.2N_0$ generations ago. The mutation rate (θ) is set to 2.0. The following command will generate 1000 replicate samples from this scenario:

```
COMPASS 25 1000 -t 2.0 -h 0.0 15 -h 1.0 10 -eN 0.2 0.1 -eN 0.8 1.0
```

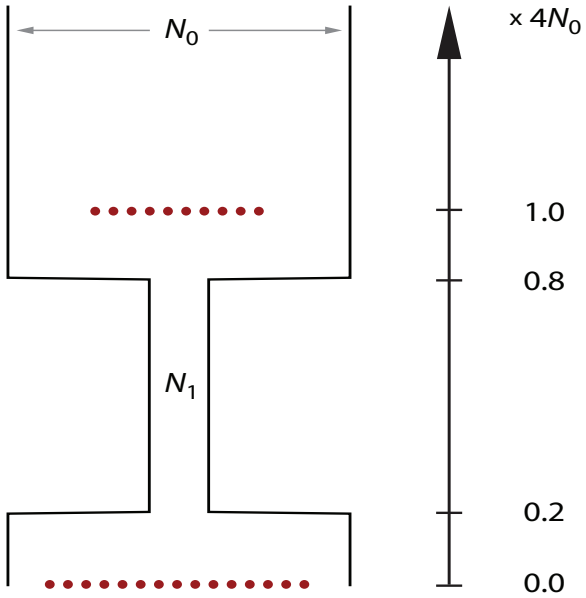

Figure 2: An example of a population history where the population has gone through a bottleneck between $0.8N_0$ and $3.2N_0$ generations ago, and the population size N_1 was a tenth of the size at present (N_0) during the bottleneck. Two samples have been taken from the population, one sample of 15 gene copies at present and one sample of 10 gene copies $4.0N_0$ generations before present.

5.3 Three sample-times and a size change followed by growth

We can simulate data from a scenario of a population that instantaneously decreased to 1/2 of the ancestral population size 100,000 years ago, and that started growing 60,000 years ago to reach 4 times the ancestral population size at present (fig. 3). Assuming that the population size at present is $N_0 = 40,000$ and that the generation time is 25 years, the population size at the start of the growth is $N_1 = 1/8 \times N_0$ and the ancestral population size $N_2 = 1/4 \times N_0$.

The times T_1 and T_2 corresponds to $T_1 = 0.015 \times 4N_0$ $(60,000/[25 \times 4 \times 40,000] = 0.015)$ and $T_2 = 0.025 \times 4N_0$ generations $(100,000/[25 \times 4 \times 40,000] = 0.025)$. We compute the growth parameter α by solving the following equality for α; $N_1 = N_0 e^{-\alpha t}$, where t is time scaled in units of $4N_0$. We have $\alpha = -\log(1/8)/0.015 \approx 138.6$. If we assume that we are modeling a piece of non-recombining DNA of 1,000 base pair, and that the mutation rate per generation and base pair is $u = 10^{-8}$, then the scaled mutation rate for the entire piece of DNA is $\theta = 4N_0 \times 1000u = 1.6$.

Based on the assumptions about population size, generation time, and mutation rate, the following command would simulate 1,000 replicates of 25 samples collected at present, 15 samples collected 60,000 years ago, and 10 samples collected 100,000 years ago:

```
COMPASS 50 1000 -t 1.6 -h 0.0 25 -h 0.015 15 -h 0.025 10 -eG 0.0 138.6 -eN 0.015
0.125 -eN 0.025 0.25
```

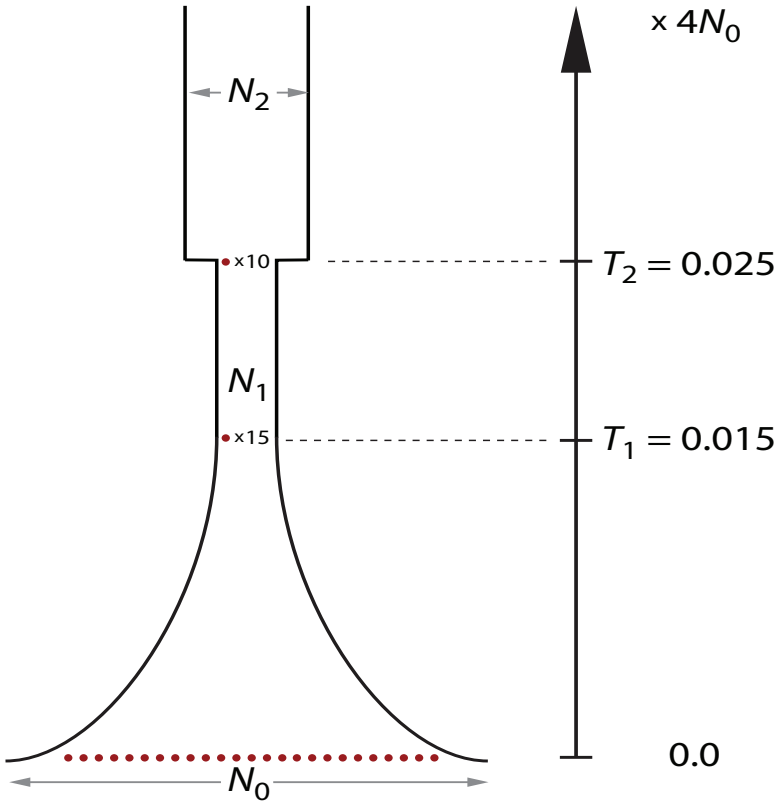

Figure 3: A demographic history of a population that (forward in time) at time T_2 decreased in size to a half of its ancestral population size, and at time T_1, the population started growing exponentially reaching a size four times as large as the ancestral size at present. Three samples have been taken from the population, one sample of 25 gene copies at present, one sample of 15 gene copies at time T_1, and one sample of 10 gene copies at time T_2.

6 Version changes

Changes from previous versions of the *COMPASS* software are noted here.

6.1 Version 0.9 (April 29, 2009)

- Initial release of the *COMPASS* software.

6.2 Version 1.0 (November 9, 2009)

- Printing bug fixed: When both the "-s" and the "-t" options were given, and if the "-s" option appeared before the "-t" option, no line indication probability was printed.
- Termination bug fixed: When only one chromosome was sampled at time zero, the program complained and exited. This bug is now fixed.
- Printing bug fixed: When no chromosomes were sampled at time zero, and the genealogy was printed ("-T" option), branch length could potentially become < 0. This bug is now fixed, and a warning appears if there are no chromosomes sampled at time zero.

6.3 Version 1.0.1 (March 17, 2010)

- Printing bug fixed: The program printed input-settings at the beginning of the output, such as "hs[j]: xx". This bug is fixed.

Acknowledgments

I thank Emma Svensson, Patrik Båtelsson and Pontus Skoglund for testing the beta version of the software.

References

Anderson, C. N. K., Ramakrishnan, U., Chan, Y. L., and Hadly, E. A. 2005. Serial SimCoal: a population genetics model for data from multiple populations and points in time, *Bioinformatics* **21**, 1733–1734.

Drummond, A. J., Nicholls, G. K., Rodrigo, A. G., and Solomon, W. 2002. Estimating mutation parameters, population history and genealogy simultaneously from temporally spaced sequence data, *Genetics* **161**, 1307–1320.

Hein, J., Schierup, M. H., and Wiuf, C. 2005. "Gene Genealogies, Variation and Evolution", Oxford University Press, Oxford.

Hofreiter, M. 2008. Long dna sequences and large data sets: investigating the quaternary via ancient dna, *Quaternary Science Reviews* **27**, 2586–2592.

Hudson, R. R. 1990. Gene genealogies and the coalescent process, *Oxford Surv. Evol. Biol.* **7**, 1–44.

Hudson, R. R. 2002. Generating samples under a Wright-Fisher neutral model of genetic variation, *Bioinformatics* **18**, 337–338.

Jakobsson, M. 2009. *COMPASS*: a program for generating serial samples under an infinite sites model, *Bioinformatics* **25**, 2845–2847.

Kimura, M. 1969. The number of heterozygous nucleotide sites maintained in a finite population due to steady flux of mutations., *Genetics* **61**, 893–903.

Kingman, J. F. C. 1982. On the genealogy of large populations, *J. Appl. Prob.* **19A**, 27–43.

Krings, M., Stone, A., Schmitz, R. W., Krainitzki, H., Stoneking, M., and Pääbo, S. 1997. Neandertal DNA sequences and the origin of modern humans, *Cell* **90**, 19–30.

L'Ecuyer, P. 1996. Maximally equidistributed combined tausworthe generators, *Mathematics of Computation* **65**, 203–213.

Miller, W., Drautz, D. I., Ratan, A., Pusey, B., Qi, J., Lesk, A. M., Tomsho, L. P., Packard, M. D., Zhao, F., Sher, A., Tikhonov, A., Raney, B., Patterson, N., Lindblad-Toh, K., Lander, E. S., Knight, J. R., Irzyk, G. P., Fredrikson, K. M., Harkins, T. T., Sheridan, S., Pringle, T., and Schuster, S. C. 2008. Sequencing the nuclear genome of the extinct woolly mammoth, *Nature* **456**, 387–390.

Noonan, J. P., Coop, G., Kudaravalli, S., Smith, D., Krause, J., Alessi, J., Platt, D., Paabo, S., Pritchard, J. K., and Rubin, E. M. 2006. Sequencing and analysis of Neanderthal genomic DNA, *Science* **314**, 1113–1118.

Noonan, J. P., Hofreiter, M., Smith, D., Priest, J. R., Rohland, N., Rabeder, G., Krause, J., Detter, J. C., Paabo, S., and Rubin, E. M. 2005. Paleontology: Genomic sequencing of pleistocene cave bears, *Science* **309** (5734), 597–600.

Nordborg, M. 2001. Coalescent theory, *in* "Handbook of Statistical Genetics" (D. J. Balding, M. Bishop, and C. Cannings, eds), chapter 7, pp. 179–212, Wiley, Chichester, UK.

Rambaut, A. and Grassly, N. C. 1997. Seq-gen: An application for the monte carlo simulation of dna sequence evolution along phylogenetic trees, *Computer Applications in the Biosciences* **13**, 235–238.

Rodrigo, A. G. and Felsenstein, J. 1999. Coalescent approaches to HIV population genetics, *in* "The Evolution of HIV" (K. A. Crandall, ed), pp. 233–272, Johns Hopkins University Press, Baltimore.

Willerslev, E. and Cooper, A. 2005. Ancient dna, *Proceedings of the Royal Society B: Biological Sciences* **272**, 3–16.