SBVB: Sequence Based Virtual Breeding

A flexible, efficient gene dropping algorithm to simulate sequence based population data and complex traits.

Miguel Pérez-Enciso,
(miguel.perez@uab.es)

With collaborations from N. Forneris, G. de los Campos and A. Legarra

Purpose

SBVB is a sequence based population simulator. Its goal is to simulate complex traits and genotype data starting with a vcf file that contains the genotypes of founder individuals and following a given pedigree. The main output are the genotypes of all individuals in the pedigree and/or molecular relationship matrices (GRM) using all sequence or a series of SNP lists, together with phenotype data. The program implements very efficient algorithms where only the recombination breakpoints for each individual are stored, therefore allowing the simulation of thousands of individuals very quickly. Most of computing time is actually spent in reading the vcf file. Future developments will optimize this step by reading and writing binary mapped files. The vcf file may not contain missing genotypes and is assumed to be phased.

Main features

- ✓ Any number of traits.
- ✓ Any number of QTNs, trait specific.
- ✓ Any number of epistatic pairs, trait specific.
- ✓ Can generate correlated allelic effects and frequencies.
- ✓ Efficient algorithms to generate haplotypes and sample SNP genotypes.
- ✓ Computes genomic relationship matrices for any number of SNP arrays simultaneously.
- ✓ Any number of chromosomes, allows for sex chromosomes and varying local recombination rates, that can be sex specific.

Installation

The source code, manual and examples can be obtained from

https://github.com/mperezenciso/sbvb0

To compile:

```
gfortran -O3 kind.f90 ALliball.f90 aux_sub11.f90 sbvb.f90 -o sbvb -lblas
```

or

```
make
```

To install in /usr/local/bin

```
sudo make install
```

The program requires blas libraries but these are standard in any unix or OS mac system. I have tested SBVB only in linux with gfortran compiler; intel ifort seems not working, but gfortran in mac OS looks ok.

Usage

To run (assuming vcf is compressed):

```
zcat file.vcf.gz | perl vcf2tped2.pl -hap | cut -d ' ' -f 1,4- | \
sbvb -isbvb.par
```

Where sbvb.par is the parameter file (details follow). The intermediate steps are simply for SBVB to read genotypes in suitable format, that is,

```
allele1_snp1_ind1 allele2_snp1_ind1 allele1_snp1_ind2 allele2_snp1_ind2 …
allele1_snp2_ind1 allele2_snp2_ind1 allele1_snp2_ind2 allele2_snp2_ind2 …
```

with alleles coded as 0/1. To run SBVB with the same random seed:

```
… | sbvb –isbvb.par –seed iseed
```

where `iseed` is an integer number.

Parameter file

The parameter file controls all SBVB behavior. It consists of a list of sections in UPPER CASE (in any order) followed in the next line by the required data, e.g.,

```
QTNFILE
sbvb.qtl
```

tells the program that QTN specifications are in `sbvb.qtl` file. Comments can be mixed starting with # or ! A full list of options in the parameter file is in Appendix 1. In the following, we list the main ones.

Specifying genetic architecture

If more than one trait is generated, then use

```
NTRAIT
ntraits
```

in parameter file. Otherwise this section is not needed. SBVB requires the user to provide the list of causal SNPs (QTNs) as specified in QTNFILE section. The format of the QTN file is

```
ichr ipos
```

or

```
ichr ipos  add_eff_trait1 add_eff_trait2 ...
```

or

```
ichr  ipos   add_eff_trait1  dom_eff_trait_1  add_eff_trait2  dom_eff_trait2 ...
```

separated by spaces, and where ichr is chromosome and ipos is position in base pair, add_eff is additive effect (ie, half the difference between 11 and 00 homozygotes), and dom_eff is the heterozygous effect.

WARNING: QTN position must coincide with one SNP position in the vcf file, otherwise it is not considered.

If QTN effects are not provided, they can be simulated specifying

```
QTNDISTA
u lower_bound upper_bound | n mu var | g s b
```

and

```
QTNDISTD
u lower_bound upper_bound | n mu var | g s b
```

where 'u' means effects are sampled from a uniform (lower_bound, upper_bound), 'n' from a normal distribution and 'g' from a gamma. For a gamma distribution, you can specify the probability p that a derived allele decreases the phenotype with

```
PSIGNQTN
p
```

The default value is 50%. By default, effects are sampled independently of frequency, but it is possible to generate a correlation (rho) with

```
RHOQA
rho
```

This option can be useful to simulate past selection. SBVB allows incorporating epistatic effects with EPIFILE section. The format of the epifile is

```
trait  ichr1  ipos1  ichr2  ipos2 e00 e01 e02  e10 e11 e12  e20 e21 e22
```

or (only if ntrait=1)

```
ichr1  ipos1  ichr2  ipos2 e00 e01 e02  e10 e11 e12  e20 e21 e22
```

with (R means reference allele and A, alternative allele):

Locus 1	Locus 2		
	RR	RA	AA
RR	e00	e01	e02
RA	e10	e11	e12
AA	e20	e21	e22

The narrow sense heritability is specified as

```
H2
h2
```

or alternatively, the broad sense heritability (using `H2G`). **Only the genotypes from the base population (in the vcf file) are used to adjust heritability.** The phenotype of individual i (y_i) is simulated from

$$y_i = \sum_{j=1}^{Q} \alpha_{ij} a_j + \sum_{j=1}^{Q} \delta_{ij} d_j + \sum_{l=1}^{Q} \sum_{m=1}^{Q} \varphi_{ilm}\, e_{lm} + \varepsilon_i$$

where Q is the number of QTNs, α_{ij} is an indicator variable taking values -1 and 1 if the genotype for the j-th QTN is RR and AA, respectively, 0 otherwise; aj is the additive effect of j-th QTN (as specified in QTN file or generated with QTNDISTA section); δ_{ij} is an indicator variable taking values 1 if the genotype for the j-th QTN is heterozygous, 0 otherwise; dj is the dominant effect of j-th QTN (as specified in QTN file or generated with QTNDISTD section). The epistatic components, if present, are similarly added, with φ_{ilm} indicator variable having a 1 with the corresponding genotype at both pairs of loci of the individual, elm being the effect defined in EPIFILE. Note therefore that **epistatic components as defined in QTNFILE do not override additive and dominant effects and care must be exerted to model epistasis as exactly as you wish**. Finally, the residual εi is sampled from a N(0,ve), where ve is adjusted given either H2 or H2G using the genotypes from the base population.

For multiple traits, the fields h2 or h2g, rhoqa, and qtldista and qtldistd must be repeated, eg, for two traits:

```
H2
0.5
0.23

RHOQA
0
-0.4

QTNDISTA
u   -0.2 0.2
g     1   0.5
```

Pedigree file (PEDFILE)

The format is

```
id  id_father id_mother [sex]
```

where all ids must be consecutive integers, 0 if father or mother unknown, sex is optional (1 for males, 2 for females) and only needed if sex chr is specified. The number of individuals in the vcf file must be specified with section

```
NBASE
nbase
```

in the parfile. The pedigree file must contain the first rows as

1 0 0
2 0 0

...
nbase 0 0

that is, those in vcf file are assumed to be unrelated.

Recombination map files

By default, SBVB assumes a cM to Mb ratio of 1. This ratio can be changed genomewide with `CM2MB` section in the par file. In addition, local recombination rates can be specified with the MAPFILE section. The mapfile takes format

```
ichr last_bp local_cm2mb
```

where local_cm2mb is the recombination rate between last_bp and previous bound (1 bp if first segment) , or

```
ichr    last_bp    local_cm2mb_males     local_cm2mb_females
```

The maximum number of chromosomes allowed by default is 23; should you require more, then section MAXNCHR must be included. SBVB permits sex chromosomes, the sex chromosome must be declared with SEXCHR section. Then, sex 1 is assumed to be the heterogametic sex, and a sex column should be present in the PEDFILE.

WARNING: chromosome ids must be integer consecutive numbers, even for the sex chr if present.

SNP files

SBVB can compute the genomic relationship matrix for all sequence data, and/or specific SNP subsets to mimic different genotyping arrays. Several SNP lists can be analyzed in the same run repeating the SNPFILE section in the par file. Each SNP file has the same format as the QTN file, i.e., chromosome and base pair position.

Output

The program writes some general info on the screen, and the following files

- **OUTYFILE** format (contains phenotypes and breeding values):

  ```
  id    y    (add[i],i=1,ntrait)    (add+dom[i], i=1, ntrait) \
             (add+dom+epi[i], i=1, ntrait)
  ```

  where add is the first sum in eq. [1] above, dom the second term and epi is the third term. For several traits, first are printed all add effects for every trait, next add+dom, and ending with add+dom+epi.

- **OUTQFILE** format (contains QTN info):

  ```
  ichr  pos  freq_base  freq  (add[i], dom[i], va[i]; i=1,ntrait)
  ```

  where ichr is chromosome, pos is QTN bp position, freq_base is frequency in vcf file, freq is frequency along the pedigree, plus additive, dominant effects and add variance ($2pq\alpha^2$) contribution for each locus by trait.

- **OUTGFILE** format (contains GRM, one per SNPFILE plus sequence)
  A matrix of n × n, where n is the number of individuals in the pedigree. As many outgfiles as snpfiles are written with subscripts .1, .2 etc. .0 corresponds to sequence. To avoid using sequence, `NOSEQUENCE` in parfile.

- **OUTMFILE** format (contains genotypes for evey SNP file and sequence, in plink format optionally using OUTPLINK in parfile). As many outmfiles as snpfiles are written with subscripts .1, .2 etc. .0 corresponds to sequence. To avoid using sequence, `NOSEQUENCE` in parfile

Outqfile, outqtn, GRM and marker files are written only if the respective sections OUTQFILE, OUTGFILE and OUTMFILE appear in the par file. Note in particular that OUTMFILE with sequence can be **huge!** To avoid printing sequence info, use

```
NOSEQUENCE
```

in the parfile. To compress marker output, include `GZIP` in parfile.

Restart the program keeping the same haplotypes

Sometimes one can be interested in running the same experiment but with different genetic architectures or different SNP arrays. SBVB offers two convenient ways to do this as it may keep track of haplotypes so exactly the same genetic structure is preserved, `RESTART` and `RESTARTQTL`.

- With `RESTART`, haplotypes, phenotypes and QTN effects are preserved. This is useful to implement selection.
- With `RESTARTQTN`, haplotypes are preserved but phenotypes and QTN effects are sampled again. `RESTARQTN` can be used to run different genetic architectures in the same haplotypes so results can be exactly comparable across models.

The program then writes a *hap file that contains all haplotype structure the first time is run. When SBVB is called again with say another `SNPFILE`, then individuals have the same haplotypes as in previous runs and a new GRM can be generated with the new SNP file. An important application is to run **selection**. In fact, SBVB can be run with different pedigree files and the `RESTART` option. SBVB generates only new haplotypes for those individuals not in current hap file. In a selection scheme, the user should add a new generation pedigree to current pedfile with the offspring of selected individuals. In the new run, SBVB generates haplotypes and phenotypes for the new offspring.

IMPORTANT:

- **The hapfile is used only if RESTART is included in parfile. If no hapfile is present, a new one is generated the first time. You can check that RESTART is in use checking, e.g, that all phenotypes are the same in different runs of SBVB.**
- **RESTARTQTN is logically not suitable for selection, since effects are sampled anew in each run.**

Expanding the base population

Very often, complete sequence is available only for very few individuals. SBVB implements an automatic option to generate additional individuals by randomly crossing the available ones and

random breeding for a pre specified number of generations. To use this feature, the pedigree file must contain larger number of individuals with unknown parents than in the vcf file. For instance, assume your vcf file contains only four individuals and the pedfile is

```
1    0    0
2    0    0
3    0    0
...
20   0    0
21   1   12
...
```

Then individuals 5-20 are generated by randomly crossing 1-4 ids, from id 21 onwards, normal pedigree gene dropping is implemented. The option is

```
EXPAND_BASEPOP
ntgen   nfam
```

which means that the new individuals are generated by crossing nfam individuals of the vcf file for ntgen generations.

Examples

Folder Examples contains an example consisting of 100 SNPs from the X chromosome of 205 lines from drosophila reference panel (http://dgrp2.gnets.ncsu.edu). The description of the files is

Base genotypes

test.vcf: original vcf file

test.gen: results from `cat test.vcf | perl vcf2tped2.pl -hap | cut -d ' ' -f 1,4-`

One trait (`cat test.gen | sbvb -i test.par`)

test.par: par file

test.qtn: list of causal SNPs, additive effects are sampled from a gamma

test.epi: epi file with two interactions (commented out in test.par by default)

test.chip: a list of SNPs from a given array

test.outy: phenotype and breeding values

test.outq: QTN effects

test.outm* : genotypes data

test.outg: GRMs

Two traits (`cat test.gen | sbvb -i test.par`)

test2.par

test2.qtn

test2.epi

test2.outq

...

Citation

M. Pérez-Enciso, N. Forneris, G. de los Campos, A. Legarra. An evaluation of sequence-based genomic prediction in pigs using an efficient new simulator. Submitted.

APPENDIX 1: Full list of commands in parameter file

```
# sbvb (Sequence Based Virtual Breeding)
# comments can be included as this
! or as this
# USAGE:
#    zcat file.vcf.gz | perl vcf2tped2.pl -hap | cut -d ' ' -f 1,4- | ./sbvb -isbvb.par
#
# WARNING: chromosomes ids in vcf file must be consecutive integer numbers
# No alphanumeric characters are allowed
NTRAIT      !--> [1]
ntrait
MAXNCHR     !-> max no. of chromosomes [23]
maxnchr
SEXCHR      !--> chr id (number) of sex chromosome,
            !    males(sex=1) are assumed to be the heterogametic sex, chr Y is not considered
sexchr

QTLFILE     !--> file with qtl posns (chr& bp) add &dom effects can be defined in cols 3 & 4
qtlfile
EPIFILE
epifile
PEDFILE
pedfile
SNPFILE    !--> file with genotyped snps: chr, bp, can be repeated
snpfile
MAPFILE    !-->recomb map file: chr, basepos, cm2Mb [cm2Mb_sex2]
mapfile
HAPFILE    !--> hap structure so program can be restarted with RESTART
hapfile
OUTPLINK   !--> prints mkr in plink tpedformat
OUTGFILE   !--> GRM outfile
outgfile
OUTQFILE   !--> output qtl file
out_q_file
OUTYFILE   !--> y outfile
outyfile
OUTMFILE   !--> output file with mkr data
outmfile
GZIP       !--> compress output files

NBASE      !-->nind which genotypes are read from STDIN
nbase
H2         !--> heritability
h2         !    repeated if multiple traits
H2G        !--> broad heritability
h2g        !    repeated if multiple traits
RHOQA      !--> desired correlation between allele effect and frequency
rhoqa      !    repeated if multiple traits
SIGNQTN    !--> P of derived allele being deleterious (only with gamma) [0.5]
p_sign_qtl
QTLDISTA   !--> QTL add effects are sampled from a distribution: u(niform), g(amma), n(ormal)
[u, l_bound, u_bound] | [n, mu, var] | [g, s, b]              !    repeated if multiple traits
QTLDISTD   !--> QTL dom effects are sampled from a distribution
[u, l_bound, u_bound] | [n, mu, var] | [g, s, b]              !    repeated if multiple traits

CM2MB      !--> cM to Mb rate, default cm2mb [1.0]
cm2mb
MXOVER     !--> Max no xovers, default 3
mxover

RESTART     !--> prepares files for new run of sbvb
RESTARTQTL  !--> restart qtl effects but keeps haplotype structure
NOPRINTHAP  !--> does not print hap file, eg, if no new haplotypes have been generated
NOSEQUENCE  !--> does not use sequence for GRM,
EXPAND_BASEPOP  !--> breeds new base individuals involving random mating for ntgen generations
                !    from nfam families
ntgen  nfam
```