POPMIN
Numerical optimization of population sizes in marker-assisted backcross programs
USER'S MANUAL

Frédéric Hospital (fred@moulon.inra.fr)

July 3, 2002

1 Introduction

In a backcross breeding program aimed at introgressing a 'target' gene T from a 'donor' line into the genomic background of a 'recipient' line, an important issue is to reduce the length of the intact chromosomal segment of donor type dragged along around the target gene (linkage drag, see Figure 1), because this segment hosts most of the unwanted donor genes still segregating in the population after a few generations (Stam and Zeven 1981; Young and Tanksley 1989; Naveira and Barbadilla 1992; Hospital 2001). This reduction can be achieved by selecting for individuals that are heterozygous at the target locus, and homozygous for recipient type alleles at two markers flanking the target locus on each side (such individuals are termed 'double homozygotes' herein).

The probability to obtain such double homozygote individuals (probability of success) depends on the distances between the target gene and the flanking markers, on the number of successive backcross (BC) generations that are to be performed (total duration of the breeding program), and on the number of individuals that are genotyped at each generation (population sizes). For a better reduction of linkage drag, flanking markers should be chosen as closely linked to the target locus as possible (Hospital 2001). But, the probability to obtain double homozygote individuals for close markers in one single BC generation is very low. Hence, it is generally preferable to perform selection on at least two successive BC generations, allowing for example to select first for a single homozygote on one side of the target, then for a single homozygote on the other side (Young and Tanksley 1989). Moreover, genotyping effort is even reduced when considering more than two BC generations, and when populations sizes at each generation are optimized simultaneously (Hospital 2001).

In order to minimize genotyping efforts, it is then necessary to compute minimal population sizes, *i.e.* the minimal number of individuals that should be genotyped at each generation so that at least one double homozygote is obtained by the end of the program. These computations over several successive BC generations are more complex than in the case of one single BC generation, because recombination between the target gene and each flanking marker can take place in any generation to provide a double homozygote genotype. Such computations must be performed numerically.

popmin is an interactive program designed for fast and easy numerical computation of such minimal population sizes in backcross programs. It is based on the theoretical framework

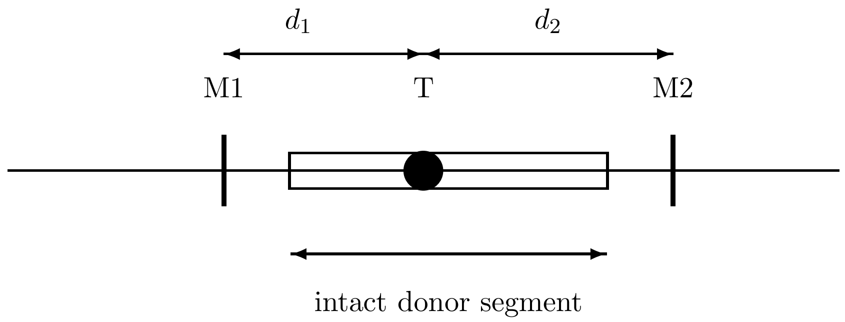

Figure 1: Linkage drag and positions of target and markers on a chromosome.

provided in Hospital (2001). The user should refer to that paper for more details. The theoretical framework assumes that, at each BC generation, one single individual is selected based on its genotype at the two flanking markers. The program provides the minimal number of individuals that should be genotyped at each generation, such that at least one individual with desired genotype is obtained by the end of the program. On an average, more than one individual may be obtained, but formally minimal population sizes such that at least $k > 1$ individuals are obtained are not considered here.

2 Methods outline

We assume that the target gene is flanked by one marker on each side (M1 and M2, see Fig. 1) at known distances d_1 and d_2 (in centiMorgans map units). Haldane mapping function is assumed throughout. We assume $d_1 \leq d_2$. We consider five possible genotypes at these three loci (see table 1). A backcross breeding program is performed during t_{max} generations or less (*i.e.*, t_{max} is the upper limit on the duration of the BC scheme, defined a priori). At each intermediate BC$_t$ generation ($t \leq t_{max}$), a total of n_t individuals (backcross progenies) are first screened for the presence of the target gene, then genotyped at the flanking markers.

		Genotype		
Index	Rank	M1	T	M2
G0	1	R/R	D/R	R/R
G1	2	R/R	D/R	D/R
G2	3	D/R	D/R	R/R
G3	4	D/R	D/R	D/R
G4	-	-/R	R/R	-/R

Table 1: Selected genotypes. R: recipient allele ; D: donor allele

One single individual is then selected based on its genotype at the two flanking markers in the following order of priority (see Rank in Table 1): 1) G0, double homozygote ; 2) G1, single homozygote for the closest marker ; 3) G2, single homozygote for the most distant marker ; 4) G3. If only genotype G4 is present in the population, the BC scheme is failed. If a double homozygote G0 is selected at a given generation t, then the BC scheme is interrupted (success), and the total number of individuals genotyped is

$$N_t = \sum_{k=1}^{t} n_k \tag{1}$$

In other cases, the selected individual is back-crossed to the recipient parental line to produce the $n_{(t+1)}$ individuals at the next $\mathrm{BC}_{(t+1)}$ generation. The probabilities S_t of success at each generation $t \leq t_{max}$ are computed by recursion (see Hospital 2001). Averaging over all possibilities, the mean number of genotypings to obtain a double homozygote by the end of the program is

$$E[N] = \sum_{t=1}^{t_{max}} N_t \, S_t \tag{2}$$

Note that here, $E[N]$ is not divided by the sum of the probabilities of success, conversely to what was done in Hospital (2001). The program optimizes the set of population sizes $\{n_t \; ; \; 1 \leq t \leq t_{max}\}$ such that

$$\text{AND} \left\{ \begin{array}{ll} E[N] & \text{is minimal} \\ \left(\sum_{t=1}^{t_{max}} S_t \right) & \geq (1 - \alpha) \end{array} \right. \tag{3}$$

where α is the risk (set by the user) that the BC scheme is not successful by t_{max}. In practice, α is only the risk that the BC scheme has to be pursuied one or more additional generation, because the probability of failure (obtaining a G4 genotype) is always close to zero.

3 Installation

popmin is written in ANSI C and runs under Unix, Linux and Windows (DOS). Complete source code along with executables and documentation can be downloaded free of charge at `http://moulon.inra.fr/~fred/programs/popmin`.

3.1 Unix/Linux

1. Copy the following files in your directory:
   Makefile
   cokus.c
   interf.h
   popmin.c
   siman.c

2. compile the program by simply running 'make'

3. run the program. See online instructions by typing 'popmin -h' or read the following sections

3.2 Windows/DOS

1. copy the file popmin.exe in your directory
2. run popmin.exe in an MS-DOS command window

NOTE: if you have a C language compiler on your Windows machine, you can also compile and run popmin as described in the Unix/Linux section above

4 Running the program

The running command is:

```
popmin [options] risk tmax d1 d2
```

The parameters `risk`, `tmax`, `d1`, `d2` and the main options are described in the following sections.

5 Parameters

`risk`: The risk (double) of failure of the backcross program, so that (1-risk) is the probability of obtaining at least one double homozygote genotype G0 in at most `tmax` BC generations (see α in section 2, equation 3)

`tmax`: The maximal duration (in generations, integer) of the backcross program (see t_{max} in section 2)

`d1, d2`: The distance (in Haldane CentiMorgans, double) from the target gene to the 'left' and 'right' markers, resp. ; `d1` must be less than or equal to `d2` (if not, interchange markers) (see d_1 and d_2 in fig. 1 and section 2)

6 Options

6.1 Population sizes options

Population sizes are computed using various methods. In all cases (except the `-t` option) the program determines the set of population sizes values $\{n_t \; ; \; 1 \leq t \leq t_{max}\}$ that satisfies equation (3) in section 2.

6.1.1 constant (-c)

The simplest method is 'constant' (-c option) were population sizes are optimized under the additional constraint $\{n_t = n \; ; \; \forall t\}$, *i.e.* population sizes are the same at any BC generation. This is done by an exhaustive search with a single loop on n. This one-dimensional search is fast, yet it does not provide the best results, since it is known that for the same overall probability of success, a better reduction of $E[N]$ is obtained by allowing population sizes to vary, and in practice to genotype *more* individuals in advanced BC generations ('variable' population sizes, see Hospital 2001).

In the case of variable population sizes, the search for optimal values is more complex, in particular for $t_{max} > 2$. Two methods are available.

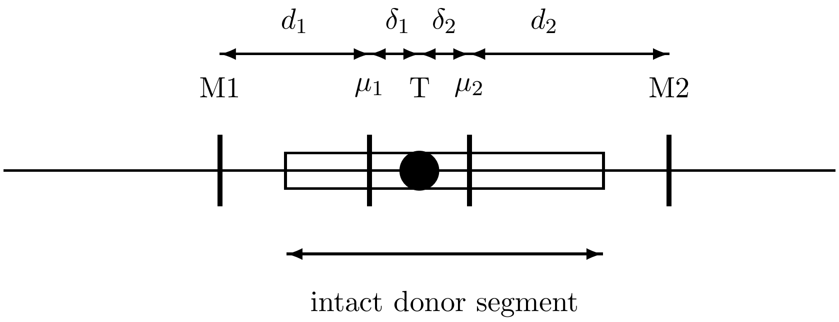

Figure 2: Same as figure 1 plus the 'foreground selection' markers μ_1 and μ_2.

6.1.2 siman (-s)

The first method is the 'siman' method (-s option), where the numerical optimization is performed using the 'simulated annealing' algorithm (Press *et al.*, 1992). This algorithm is implemented using the **siman** package extracted from GSL, the GNU Scientific Library. This is a fast and reliable search.

6.1.3 exhaustive (-e)

The second method performs an exhaustive search ('exhaustive' method; -e option). Here, all possible values of population sizes n_t ($t \leq t_{max}$) are investigated in turn. This method may be quite long for $t_{max} \geq 3$ and should be reserved for particular purposes (*e.g.*, validation of **siman** parameters).

The checks that we performed using either the 'siman' or 'exhaustive' methods showed that the 'siman' search provided the same optimal population size values as the 'exhaustive' search in the vast majority of cases, and much faster. The internal parameters of both the 'siman' and 'exhaustive' search methods were optimized with respects to realistic input parameter values to **popmin**. However, specific options make it possible to modify the internal parameters of the search methods (see online help). Note that this requires some expertise.

For 'constant', 'siman', and 'exhaustive' methods the optimization criterion (*i.e.*, the quantity to minimize) is $E[N]$ as defined in equation (3). Optimization for another criterion may be performed after simple modification in the source code (indicated by inside comment).

6.1.4 test (-t)

In addition to the optimization methods, the 'test' method (-t option) does not compute optimal population sizes, but simply computes the probabilities of success S_t at different generations for particular population sizes (entered interactively by the user). This makes it easy to use **popmin** as a simple numerical implementation of the recursion equations of Hospital (2001), in order to check probabilities of success for given duration and population sizes. Note that in this case the conditions of equation (3) might not be fulfilled.

6.2 Target gene option (-P)

By default, it is assumed that the target gene is at a single locus, which can be identified unambiguously (see Hospital, 2001, Definitions section, for more details), so that the probability of transmission of the target gene to a backcross progeny is $P = 1/2$. This default value can be altered using the -P option, in the case where the target gene is identified indirectly by 'foreground selection' markers (μ_1 and μ_2 at distances δ_1 and δ_2 (in Haldane centiMorgans) from the target, respectively, see figure 2), in order to take account of possible recombination between those markers. In such case, one should set P=$1/2(1-\rho_1)(1-\rho_2)$, with $\rho = 1/2(1-e^{-2\,\delta/100})$ the recombination rate corresponding to δ. Note that in this case, d_1 and d_2 have different meanings (see Hospital, 2001, for more details).

6.3 Initial genotype (-I or -i)

By default, it is assumed that the backcross program starts ($t = 0$) with a single individual that is heterozygous for the donor allele at both the target gene and the two flanking markers (typically, an F_1 hybrid between the donor and the recipient parental lines). This corresponds to genotype index G3 in Table 1. Using the -I or -i option, it is possible to tell **popmin** to start with another initial genotype. For example `popmin -I 2` would start with initial genotype G2, *i.e.* marker M2 is already homozygous. This is useful to optimize the remaining of an already started backcross program, for example when a single homozygous individual for one of the two markers has already been selected at a previous BC generation. -i is the same as -I except that genotype index is entered interactively upon request, not on the command line.

Other (less useful) options are available but not described here, see the online help (`popmin -h`) for more details.

7 Outputs

Options can be combined together, so that for example the command: `popmin -cs .01 3 5.1 10.2` computes 'constant' and 'siman' minimal population sizes over three backcross generations for flanking markers at 5.1 cM and 10.2 cM on each side of target gene, respectively, with an overall probability of success of .99 over the three generations. The corresponding results are given in Figure 3.

First of all, the output recalls the input parameters, then it displays the optimal population sizes computed for the requested options, where the equivalence between column headings in the output and notations used in section 2 is as follows:

Gener	Population size	Cumulated pop. size	Probability of success	Cumulated probability
t	n_t	$N_t = \sum_{k=1}^{t} n_k$	S_t	$\sum_{k=1}^{t} S_k$

$$\text{Average cumulated population size} = E[N] = \sum_{t=1}^{t_{max}} N_t\, S_t$$

The output in Fig. 3 illustrates the interest of using variable, rather than constant, population sizes. Although the cumulated population size (total number of individual genotyped) is the same, the average cumulated population size $E[N]$ is reduced from 126 to 104 when using variable population sizes (siman option), because the probabilities of success are different.

One can also use **popmin** to investigate different strategies based on different total number of BC generations (t_{max}). For example, for flanking markers located at 2 cM from the target on each side, one can compare three strategies, such that the probability to obtain a

```
---------------
popmin results:
---------------

input parameters:
---------------------------------------------
d1 =    5.100000 cM     r1 = 0.048485
d2 =    10.200000 cM    r2 = 0.092269
tmax =  3
risk =  0.010000
P =     0.500000
starting genotype = 3  ( M1--T--M2 = D--D--D )
---------------------------------------------

constant:
-------------------------------------------------------------------
Gener   Population      Cumulated       Probability     Cumulated
        size            pop. size       of success      probability
-------------------------------------------------------------------
BC1     66              66              0.137393        0.137393
BC2     66              132             0.787688        0.925082
BC3     66              198             0.065520        0.990601
-------------------------------------------------------------------
Average cumulated population size = 126.015690
-------------------------------------------------------------------

siman:
-------------------------------------------------------------------
Gener   Population      Cumulated       Probability     Cumulated
        size            pop. size       of success      probability
-------------------------------------------------------------------
BC1     38              38              0.081575        0.081575
BC2     47              85              0.697224        0.778799
BC3     113             198             0.211205        0.990005
-------------------------------------------------------------------
Average cumulated population size = 104.182543
-------------------------------------------------------------------
```

Figure 3: Example of output produced by the command `popmin -cs .01 3 5.1 10.2`

```
---------------
popmin results:
---------------

input parameters:
---------------------------------------------
d1 =    2.000000 cM     r1 = 0.019605
d2 =    2.000000 cM     r2 = 0.019605
tmax =  2
risk =  0.010000
P =     0.500000
starting genotype = 3  ( M1--T--M2 = D--D--D )
---------------------------------------------

siman:
-------------------------------------------------------------------
Gener   Population      Cumulated       Probability     Cumulated
        size            pop. size       of success      probability
-------------------------------------------------------------------
BC1     290             290             0.054214        0.054214
BC2     499             789             0.935792        0.990006
-------------------------------------------------------------------
Average cumulated population size = 754.062104
-------------------------------------------------------------------
```

Figure 4: Example of BC2 strategy `popmin -s .01 2 2 2`

double homozygote by the end of the breeding scheme is always above 99%. The minimum number of individuals that should be genotyped to obtain a double homozygote in BC1 (BC1 strategy) is 23961, obviously far too many. Running `popmin -s .01 2 2 2` shows that over two generations (BC2 strategy, Fig. 4) the optimal population sizes are 290 individuals in BC1, and 499 in BC2. Finally, running `popmin -s .01 3 2 2` shows that over three generations (BC3 strategy, Fig. 5), the optimal population sizes are 117 individuals in BC1, 171 in BC2, and 370 in BC3. However, with the BC3 strategy, in 72% of the cases the double homozygote (success) is obtained in BC2 with genotyping a total of only 288 individuals. Only in 25% of the cases should the program be really conducted until generation BC3. Hence, averaging over all possibilities, the mean number of genotypings to obtain a double homozygote with the BC3 strategy is only 374, to be compared with an average of 754 with the BC2 strategy. Hence, unless a rapid success is really mandatory, allowing a not-too-low risk of failure in early BC generations (risk of not obtaining a double homozygote at that generation), can permit a drastic reduction of genotyping costs and should be recommended, which is converse to what is generally advocated.

Obviously, the strategy and number of individuals to be genotyped should be reconsidered at each generation once the actual genotype of the individual selected is known. This is also possible using **popmin** with the -I option (see above).

```
---------------
popmin results:
---------------

input parameters:
---------------------------------------------
d1 =    2.000000 cM     r1 = 0.019605
d2 =    2.000000 cM     r2 = 0.019605
tmax =  3
risk =  0.010000
P =     0.500000
starting genotype = 3  ( M1--T--M2 = D--D--D )
---------------------------------------------

siman:
-------------------------------------------------------------------
Gener   Population      Cumulated       Probability     Cumulated
        size            pop. size       of success      probability
-------------------------------------------------------------------
BC1     117             117             0.022237        0.022237
BC2     171             288             0.717442        0.739678
BC3     370             658             0.250324        0.990002
-------------------------------------------------------------------
Average cumulated population size = 373.937861
-------------------------------------------------------------------
```

Figure 5: Example of BC3 strategy `popmin -s .01 2 2 2`

References

Frisch M, Bohn M , and Melchinger AE, 1999. Minimum sample size and optimal positioning of flanking markers in marker-assisted backcrossing for transfer of a target gene. Crop. Sci. 39: 967–975.

Hospital F, 2001. Size of donor chromosome segments around introgressed loci and reduction of linkage drag in marker-assisted backcross programs. Genetics 158: 1363–1379.

Naveira H and Barbadilla A, 1992. The theoretical distribution of lengths of intact chromosome segments around a locus held heterozygous with backcrossing in a diploid species. Genetics 130: 205–209.

Press WH, Teukolsky SA, Vetterling WT and Flannery BP, 1992. Numerical Recipes in C The Art of Scientific Computing, Second Edition. Cambridge, UK: Cambridge University Press.

Stam P and Zeven AC, 1981. The theoretical proportion of the donor genome in near-isogenic lines of self-fertilizers bred by backcrossing. Euphytica 30: 227–238.

Young ND and Tanksley SD, 1989. RFLP analysis of the size of chromosomal segments retained around the *tm-2* locus of tomato during backcross breeding. Theor. Appl. Genet. 77: 353–359.