GOplan Manual

GOplan is an R package designed to manage animal breeding programs, including the breeding program of the core population and the whole breeding strategy of the crossbreeding system. Moreover, it allows optimization of the crossbreeding system.

The package has three main functions. Parameter “prm_path” is the full path of the input parameter file:

runCore(prm_path) is the function designed to assess various breeding schemes for the nucleus herds, you can compare the genetic progress as well as the monetary profit under different key elements in constructing populations such as the productive lifetime of breeding stocks, mating ratio, mating methods, the number of dams, cull rate, the method of breeding value estimation, the number of individuals selected for phenotyping or genotyping, and so on. It returns five aspects of results, the realized genetic progress, the final population mean of traits, the decline of genetic variance, the inbreeding increment, and the economic profit. Widely used BV estimation methods including BLUP , GBLUP, and ssGBLUP (single-step GBLUP) are supported by this function, which can be executed through calling the DMU software. Additionally, to improve the flexibility of mating methods, users can invoke external mating programs to generate their own mating plans by specifying the program's path.

runWhole(prm_path) designed to easily evaluate the performances of crossbred animals and the economic profit of complex crossbreeding programs with built-in frameworks. It allows the evaluation of different crossbreeding systems and can evaluate the effectiveness of crossbreeding program with different population structures by returning the predicted value of hybrids’ phenotype and economic profit.

runOpt(prm_path) is used to search for an optimal combination of parameter values which maximizing the economic profits of crossbreeding programs. By defining some uncertain parameters and giving reasonable variation ranges, it uses the Bayesian Optimization method to draw up a superior solution under fixed iteration times.

Description of parameter file:

All needed information is determined by editing the parameter files. It is worth noting that the parameters you do not need should be specified by 0, and do not use any other separators except blank to separate different numbers, or it may cause unknown errors. Do not change the sequence of each parameter.

The parameter file contains six parts, each part is illustrated below.

### Analyse Parameters ###

This part is used to determine some basic information when running all three functions.

parameters	value	explanation
QUICK	FALSE	#FALSE, TRUE
Method	blup	#blup, ssgblup, gblup
EarlySelect	FALSE	#FALSE, TRUE
Lit_pro	0	#the percentage of pre-selected litters, range from 0 to 1
Mate	rand	#Mate method: rand, MC or the path of extra program
maxF	-999	#the maximum kinship coefficient between parents
Ctype	33	#type of crossbreeding: 1, 21, 22, 31, 32, 33, 41, 42
Nrep	50	#number of repetitions
Time	20	#time budget of breeding cycle
Ncores	10	#number of threads
nChr	18	#number of simulated chromosomes
nSnpPerChr	3000	#number of SNP per chr
nQtlPerChr	100	#number of QTL per chr
out_path		#path of output file

QUICK: when set to TRUE, the program would simplify the simulation of nucleus population by replacing the process of breeding value estimation with simulation. It would just calculates the first generation's estimated breeding value (EBV) (estimate the breeding value using DMU, then use the correlation between EBV and TBV as the accuracy of breeding value estimation), and use Cholesky method to simulate the other generations' breeding value according to the estimation accuracy. We assume TBV and EBV ~ N (u, s^2), so the equation of simulating EBV is:

$$EBV_{sim} = u + r * (TBV - u) + \sqrt{1 - r^2} * e * s$$

Where EBV_{sim} is the vector of simulated EBV of all individuals, u is the mean value of all individuals' TBV, r is the accuracy (Pearson's correlation coefficient between TBV and EBV), TBV is the vector of true breeding value, e is a randomly generated vector that follows a normal distribution ~ N (0,1), s is the standard deviation of TBV.

The QUICK option can save time and get a result quickly, if you just want to have a qualitative understanding of which breeding program is better and do not care about the exact profit, or when the **Method** is set to 'gblup' or 'ssgblup', we suggest set QUICK to TRUE.

Method: the program now supports three methods to evaluate the breeding value, and call DMU to run it. blup: best liner unbiased prediction, use only pedigree information; **ssgblup**: single-step blup, use both genomic and pedigree information; gblup: genomic blup, use only genomic information.

EarlySelect: FALSE, whether do pre-select for litters. If TRUE, GOplan would select some litters according to their litter index, which is the average of the father's breeding value and the mother's breeding value. Litter with high litter index would be selected.

Lit_pro: the the percentage of pre-selected litters, range from 0 to 1.

Mate: there are four options for users to choose, rand (random mating), HOMO: positive assortative mating, HETER: negative assortative mating, MC (minimal-coancestry mating, using annealing algorithm to find the solution); or you can specify the full path of your own mating program, while our package would write a file called "plans.txt" containing four columns, id of dam, id of sire, relationship coefficient and the expected breeding value of offspring out. See implementation information of MC in the additional file.

maxF: the maximum kinship coefficient between parents. Males and females with kinship coefficients greater than this value will not mate. When set to -999, males and females are mated without any restriction.

Ctype: crossbreeding type, see detail in Figure 1, 1 means single population, 2* means tow-way crossbreeding (B is the sire line, A is the dam line), 3* means three-way crossbreeding (C is the sire line, B is the sire of dam line, A is the dam of dam line), and 4* means four-way crossbreeding (D is the sire of sire line, C is the dam of sire line, B is the sire of dam line, A is the dam of dam line).

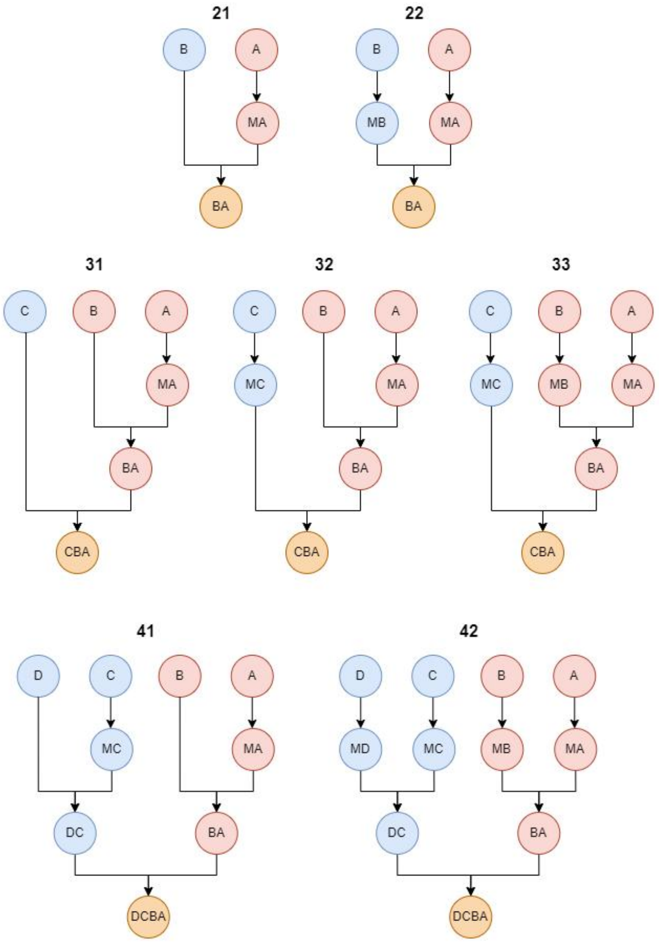


Figure1: the sketch map of different crossbreeding types.

Nrep: the number of repetitions. Final results would be the average of all repetitions.

Time: time budget of breeding cycle.

Ncores: number of threads. Our package supports multi threaded running. But please make sure that **Ncores** do not exceed the maximum available thread in your system.
nChr: number of simulated chromosomes. Change this parameter to adapt to different breeds.
nSnpPerChr: number of SNP per chromosome.
nQtlPerChr: number of QTL per chromosome.
out_path: path of output file, ended with “/”.

VARIABLES

This part can either be specified or ignored
The number of variables does not have limitation, but more variables means more combinations, which may cause long time running. We recommend to set less than three variables each time, one variable one row, with all the levels listed separated by blank.
Available variables for ***runCore(prm_path):*** Ys, Yd, Sor, nfam, nfam_F, nfam_M, ngeno, nGeno_F, nGeno_M. Using nfam or ngeno means adjust nfam_F and nfam_M or nGeno_F and nGeno_M simultaneously.
Available variables for ***runWhole(prm_path):*** Asec, Bsec, Csec, Dsec, YABd, YCDs, SorAB, SorCD, SorP

### Optimize Parameters ###

This part used to modify the range of each optimize parameter. If you don’t need some parameters, just delete the row of it. When use ***runCore(prm_path), runWhole(prm_path),*** ignore this section.

parameters	value	value	explanation
Pars	lower	upper	
Asec	2	6	#longevity of mutiplier A
Bsec	1	4	#longevity of mutiplier B
Csec	2	6	#longevity of mutiplier C
Dsec	1	4	#longevity of mutiplier D
YABd	2	6	#longevity of hybrid dam AB
YCDs	1	4	#longevity of hybrid dam CD
NAd	600	3000	#female size of nucleus A
NBd	300	1000	#female size of nucleus B
NCd	300	1000	#female size of nucleus C
NDd	300	1000	#female size of nucleus D

### Population Structure ###

This part used to specify the structure of population in the breeding program.

parameters	value	value	value	value	explanation

dam0	2				#age of dams when first born
sire0	2				#age of sires when first born
YABd	6				#longevity of hybrid dam AB
cull_AB	0	0	0		#cull rates of each age in AB
YCDs	4				#longevity of hybrid sire CD
cull_CD	0	0	0		#cull rates of each age in CD
SorAB	100				#Mating ratio when AB cross
SorCD	100				#Mating ratio when CD cross
SorP	100				#Mating ratio of terminal cross
ABSR	0.85				#survival rate of AB
CDSR	0.85				#survival rate of CD
PSR	0.85				#survival rate of products
N_P	0				#expected number of final products
nPoints	0				#number of selected points each iteration
nIter	0				#number of iterations
Breed	A	B	C	D	
n_female	500	500	500	0	#female size
Sor	50	50	50	0	#mate ratio
Ys	2	2	2	0	#longevity of sires
Yd	4	4	4	0	#longevity of dams
n_progeny	12	12	12	0	#litter size
SR	0.85	0.85	0.85	0	#survival rate of offspring
nSec	3000	0	0	0	#size of mutiplier
Ysec	6	2	2	0	#longevity of mutiplier
nPedG	3	0	0	0	#the number of generations traced for generating pedigree
nGenoG	3	0	0	0	#the number of generations traced for generating genotype file
nSel_F	2	0	0	0	#the number of female selected per litter
nSel_M	2	0	0	0	#the number of male selected per litter
nfam_F	2	2	2	0	#number of phenotyping females per litter
nfam_M	2	2	2	0	#number of phenotyping males per litter
nGeno_F	0	0	0	0	#number of genotyping females per litter
nGeno_M	0	0	0	0	#number of genotyping males per litter

### Breeds Details ###

This part used to specify the information of each breed. You must list the information of all breeds.

parameters	value	value	value	value	value	explanation
Breedname	A					
cull_s	0					#cull rates of each age in sires
cull_d	0	0	0			#cull rates of each age in dams
cull_sec	0	0	0	0	0	#cull rates of each age in multipliers
trait	JZRL					#names of considering traits
SexLimit	0					#0: non-limiting trait, 1: trait that only for male, 2: trait only for female
mean	180					#means of traits
var	100					#phenotype variances of traits
heri	0.3					#heritability of traits
weigh	-1					#weight coefficients of traits
analyse	1					#analyse method
## phenotype covariance ##						
JZRL	1					
Breedname	B					
cull_s	0					
cull_d	0	0	0			
cull_sec	0	0	0	0	0	
trait	JZRL					
SexLimit	0					
mean	170					
var	100					
heri	0.3					
weigh	-1					
analyse	1					
## phenotype covariance ##						
JZRL	1					
Breedname	C					
cull_s	0					
cull_d	0	0	0			

cull_sec	0	0	0	0	0	
trait	JZRL					
SexLimit	0					
mean	160					
var	100					
heri	0.3					
weigh	-1					
analyse	1					
## phenotype covariance ##						
JZRL	1					

analyse: "1" means single trait, other same numbers mean multiple traits. For example, if there are three traits, T1, T2 and T3, T1 need to be estimated by single trait, while T2 and T3 need to be estimated using multi-trait model. Then we set analyse: 1, 2, 2.

## phenotype covariance ##: this part determines the covariance of traits. For example, when there are three traits, T1, T2 and T3. Then it needs to be adjusted as below:

T1	1	0	0
T2	0	1	0
T3	0	0	1

Note: the column names are the same as the row names, though we omit them.

The parameter number of cull rate should always be one less than its corresponding longevity. For example, if the Yd of breed C is 4, then the cull_d of C must have three values (0, 0, 0). Because when parent exceed their longevity, they would be all culled.

Basic Economic Parameters

This part defines the economic information of the breeding program.

parameters	value	value	explanation
trait	0	JZRL	
jb_cost	906	1.5	#basic cost per individual
dam_cost	1500	0	#dam cost per individual
sire_cost	1000	0	#sire cost per individual
other_cost	10	0	#other cost each season
ind_sale	2040	0	#income of selling a product
cull_sale	1500	0	#income of culling
sire_sale	3000	0	# income of selling a sire

dam_sale	2000	0	# income of selling a dam
meas_cost	20		#phenotyping cost per individual
geno_cost	200		#genotyping cost per individual

trait: first column fixed with 0, means the initial value of all economic parameters. Then list the traits concerned, each trait corresponds to a column, means the monetary change when each trait changes one unit. The value can be negative.

jb_cost: the basic cost for raising an individual.

dam_cost: cost for keep a dam in the population each season.

sire_cost: cost for keep a sire in the population each season.

other_cost: total other cost each season.

ind_sale: income of selling a product.

cull_sale: income of culling a breeding stock.

sire_sale: income of selling a sire, a male who had been measured but not be selected can be sold as a sire.

dam_sale: income of selling a dam, a female who had been measured but not be selected can be sold as a dam.

meas_cost: phenotyping cost per individual.

geno_cost: genotyping cost per individual

Running Example & Output File

runCore() would return three files, coreOut.xlsx, coreOut_detail.xlsx and detailInfo.txt.

coreOut.xlsx has six columns listed below, and it is the results after the breeding process running after all breeding seasons.

G	Relative genetic progress
Inb	Inbreeding increase
Profit	The economic profit
Phenotype	Predicted mean phenotype
Vg_Phenotype	Population genetic variance decrease of phenotype
(Last column)	The name of variation combination

coreOut_detail.xlsx records the information of each breeding season, and each type of information is saved in a separate sheet.

varg	Population genetic variance of each phenotype
acc	Estimation accuracy of each phenotype’s breeding value

acc_index	Estimation accuracy of the aggregate selection index
index	The average aggregate selection index of population
dfInb	Inbreeding of population
pop0_G	Predicted mean phenotype

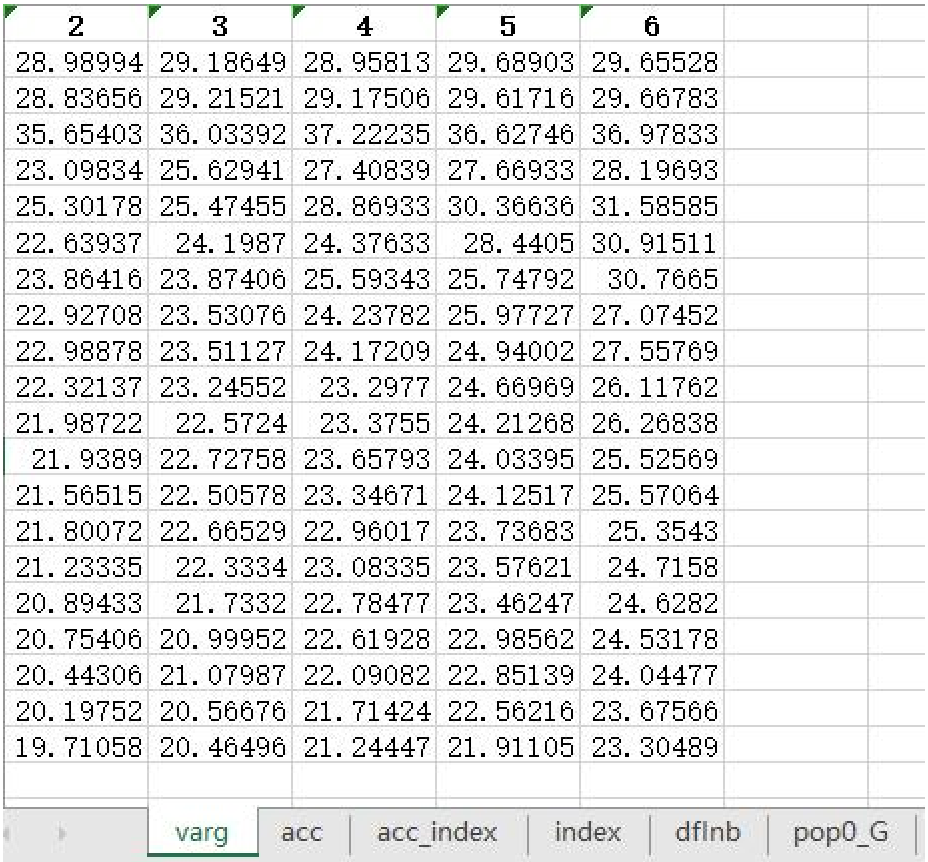
detailInfo.txt shows information about the population structure and breeding details under different variation combinations.

Here, we give a example of running breeding comparison with variables **Yd** in nucleus population. The parameter file was **prm_Core.txt** in folder example_prm: , and the results are shown below:

coreOut.xlsx :

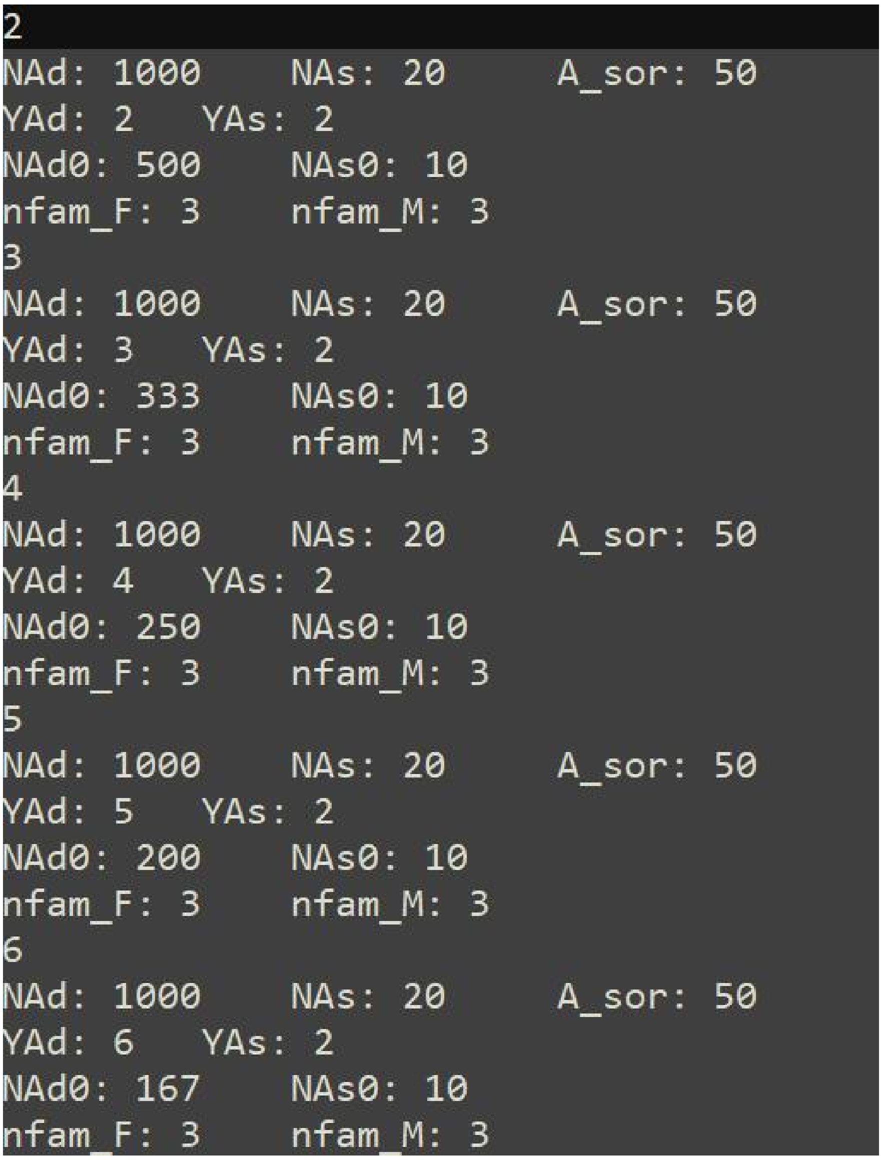
G	Inb	Profit	JZRL	Vg_JZRL	Yd
0.240	0.1045	11353818	136.85	-6.35	6
0.241	0.1020	11341171	136.64	-7.77	5
0.250	0.1077	11332108	135.01	-7.71	4
0.253	0.1101	11298887	134.40	-8.72	3
0.258	0.1137	11224855	133.65	-9.27	2

coreOut_detail.xlsx :



2	3	4	5	6
28.98994	29.18649	28.95813	29.68903	29.65528
28.83656	29.21521	29.17506	29.61716	29.66783
35.65403	36.03392	37.22235	36.62746	36.97833
23.09834	25.62941	27.40839	27.66933	28.19693
25.30178	25.47455	28.86933	30.36636	31.58585
22.63937	24.1987	24.37633	28.4405	30.91511
23.86416	23.87406	25.59343	25.74792	30.7665
22.92708	23.53076	24.23782	25.97727	27.07452
22.98878	23.51127	24.17209	24.94002	27.55769
22.32137	23.24552	23.2977	24.66969	26.11762
21.98722	22.5724	23.3755	24.21268	26.26838
21.9389	22.72758	23.65793	24.03395	25.52569
21.56515	22.50578	23.34671	24.12517	25.57064
21.80072	22.66529	22.96017	23.73683	25.3543
21.23335	22.3334	23.08335	23.57621	24.7158
20.89433	21.7332	22.78477	23.46247	24.6282
20.75406	20.99952	22.61928	22.98562	24.53178
20.44306	21.07987	22.09082	22.85139	24.04477
20.19752	20.56676	21.71424	22.56216	23.67566
19.71058	20.46496	21.24447	21.91105	23.30489

detailInfo.txt:



```
2
NAd: 1000    NAs: 20     A_sor: 50
YAd: 2   YAs: 2
NAd0: 500    NAs0: 10
nfam_F: 3    nfam_M: 3
3
NAd: 1000    NAs: 20     A_sor: 50
YAd: 3   YAs: 2
NAd0: 333    NAs0: 10
nfam_F: 3    nfam_M: 3
4
NAd: 1000    NAs: 20     A_sor: 50
YAd: 4   YAs: 2
NAd0: 250    NAs0: 10
nfam_F: 3    nfam_M: 3
5
NAd: 1000    NAs: 20     A_sor: 50
YAd: 5   YAs: 2
NAd0: 200    NAs0: 10
nfam_F: 3    nfam_M: 3
6
NAd: 1000    NAs: 20     A_sor: 50
YAd: 6   YAs: 2
NAd0: 167    NAs0: 10
nfam_F: 3    nfam_M: 3
```

runWhole() would print out four files.

WholeOut.xlsx contains predicted phenotype of final hybrids and economic profit under different breeding programs.

*out.xlsx contains information about different subpopulations' phenotype in each breeding season.

detailInfo.txt is the same as described before.

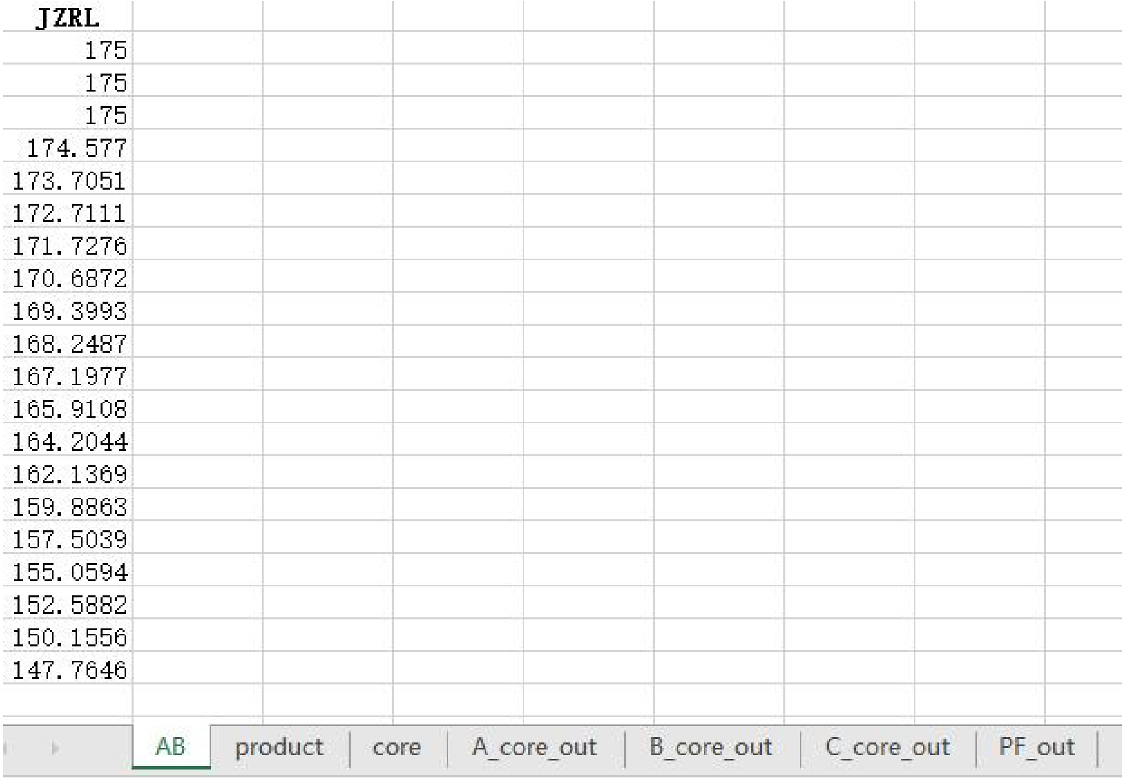
p.csv is the P matrix used in gene flow method.

Here, we give a example of running crossbreeding comparison with variables **SorP** . The parameter file was **prm_Whole.txt** in folder example_prm: , and the results are shown below:

WholeOut.xlsx:

JZRL	Profit	SorP
138.81	1154.09	50
138.37	1155.14	75
138.26	1155.604	100
137.94	1156.187	125

***out.xlsx:**



JZRL
175
175
175
174.577
173.7051
172.7111
171.7276
170.6872
169.3993
168.2487
167.1977
165.9108
164.2044
162.1369
159.8863
157.5039
155.0594
152.5882
150.1556
147.7646

The rusults of ***runOpt()*** is in the output log file, and we provide a function ***getOptRes(log_name, out_path)*** to extracting the results from the log. ***log_name*** is the full name of log, and the ***out_path*** is the path to write results out. This function return a file called **mboRes.csv**.

Additional file

1 Implementation of minimal-coancestry mating (MC) using annealing algorithm

a) Generate an initial mate plan stochastically, and calculate the overall coefficient of relationship: E0, set T=1.0, Nre = 0, Neva = 0, where T is the annealing temperature, Nre is the number of plan's replace times, Neva is the number of evaluation times;

b) Change the plan by randomly selecting two mate pairs, and exchanging the dams, the new plan's Ei is equal with E0 + δ, where $\delta = a_{s1d2} + a_{s2d1} - a_{s1d1} - a_{s2d2}$, a_{s1d2} is the coefficient of relationship between sire 1 and dam 2 (Fig.2);

c) If δ<0, accept the new plan, or accept it with the probability of $e^{(-\delta/T)}$;

d) Update the T value with 10% reduction each time when either the plan has been replaced 10*nmax times or has already changed plan 100*nmax times, where nmax is the maximum between the number of dams and the number of sires;

e) Stop if there is no improvement between the last and the second last change of T, or repeat the above steps.

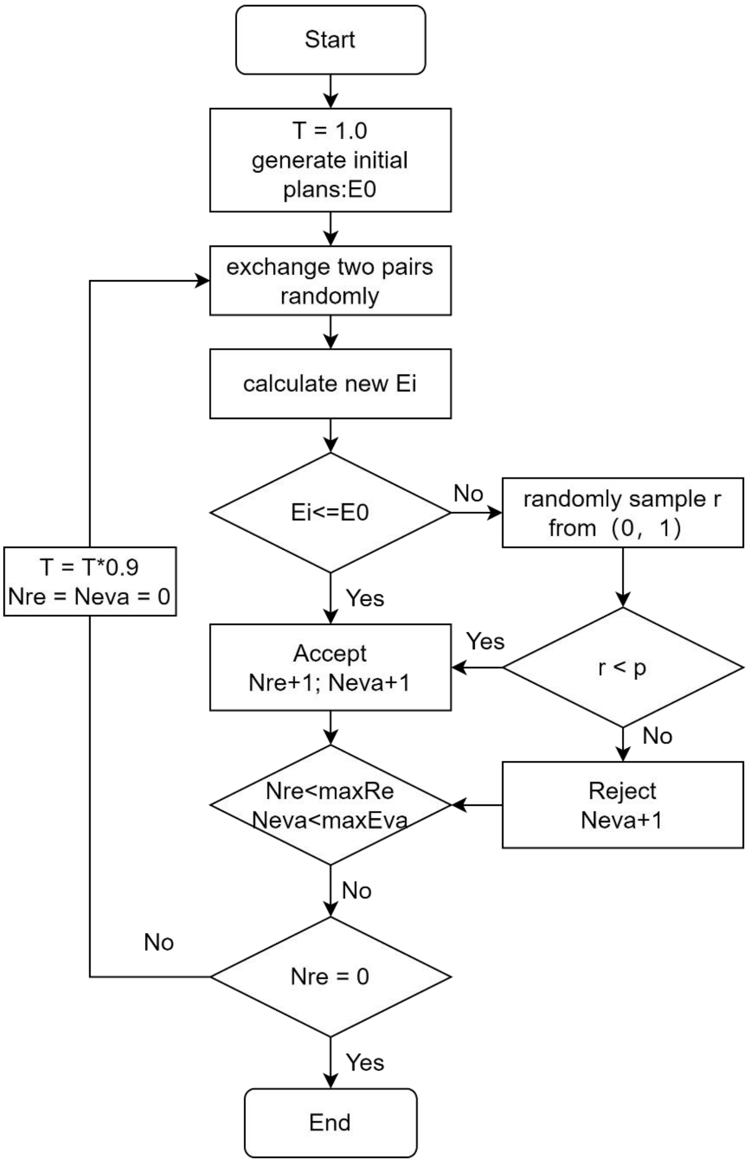


Fig. 1 A diagram illustrate the whole process of MC mate.

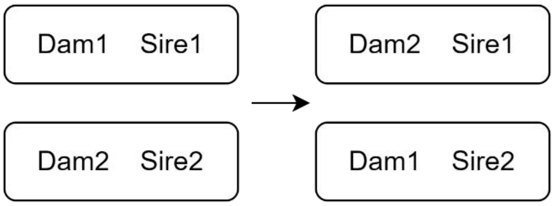


Fig. 2 Shows exchange of two mate pairs.

2 Calculation of monetary profit

In this package, we calculate the profit of each season separately and do not consider about the discount.

Main steps:

a) Calculate the phenotype progress in time i, and figure out each economic part's value;

b) Calculate the number of individuals of different types:

TYPE	CALCULATION METHOD
Weaning Number	The number of all dams * litter size
Culling Number	Total number of culled animals
Measuring Number	Total number of measured animals
Genotyping Number	Total number of genotyped animals
Sold Dams	Measuring Number of female - number of culled dams in core population or multiplier
Sold Sires	Measuring Number of male - number of culled sires in core population or multiplier
Product	The number of products' dams * litter size * survival rate
All Product	Weaning number * survival rate – max(Culling Number, (Measuring Number + number of culled animals of products' parents))

c) Calculate the economic parts:

ECONOMIC PART	CALCULATION METHOD
All Basic Cost	Weaning Number * jb_cost'
All Sire Cost	Total number of sires * sire_cost'
All Dam Cost	Total number of dams * dam_cost'
All Measure Cost	Measuring Number * meas_cost'
All Genotyping Cost	Genotyping Number * geno_cost'
Income of selling products	All Product * ind_sale'
Income of Selling Dams	Sold Dams * dam_sale'
Income of Selling Sires	Sold Sires * sire_sale'
Income of Culling Animals	Culling Number * cull_sale'

d) Profit = (Income of selling products + Income of Selling Sires + Income of Selling Dams + Income of Culling Animals) - (All Basic Cost + All Sire Cost + All Dam Cost + All Measure Cost + All Genotyping Cost + other_cost)