

**Which recurrent selection scheme to improve
mixtures of crop species?
Theoretical expectations**

Jean-Paul Sempoux,^{,1} Héloïse Giraud,^{*,2} and Isabelle Litrico**

**SUPPLEMENTAL MATERIAL
FILE S1
NONLINEAR CONSTRAINED OPTIMIZATION**

*INRA, Centre Nouvelle-Aquitaine-Poitiers, UR4 (UR P3F - Unité de Recherche Pluridisciplinaire Prairies et Plantes Fourragères), CS80006, 86600 Lusignan, France

¹Corresponding author: INRA, Centre Nouvelle-Aquitaine-Poitiers, UR P3F, CS80006, 86600 Lusignan, France. Phone: +33-5-49556027. E-mail: jean-paul.sempoux@inra.fr

²Present address: BASF BBCC Innovation Center, Technologiepark 101, 9052 Gent, Belgium

Nonlinear constrained optimization

Finding the best set of index weights of observed species contributions to the performance of the mixture with Selection for General Mixture Ability (SGMA) in two species using an algorithm of nonlinear constrained optimization

1. Definition of the optimization problem The index for the selection process in species 1 is set as $I_{1r.} = \alpha_{11}x_{11r.} + \alpha_{21}x_{21r.}$

Assuming that the variance-covariances of the mixture model effects are genetically additive, the expected responses of species contributions to the performance of the mixture to selection using $I_{1r.}$ as selection criterion are:

$$\Delta G_{x_{11}}^G = \theta_1 \phi_1 \frac{i_1}{\sigma_{I_{1r.}}} (\alpha_{11} \sigma_{v_1}^2 + \alpha_{21} Cov(a_1, v_1))$$

and

$$\Delta G_{x_{21}}^G = \theta_1 \phi_1 \frac{i_1}{\sigma_{I_{1r.}}} (\alpha_{11} Cov(v_1, a_1) + \alpha_{21} \sigma_{a_1}^2).$$

Similarly, for the selection process in species 2, the index is set as $I_{2s.} = \alpha_{22}x_{22s.} + \alpha_{12}x_{12s.}$

Under the same assumption, the expected responses of species contributions to the performance of the mixture to selection using $I_{2s.}$ as selection criterion are:

$$\Delta G_{x_{22}}^G = \theta_2 \phi_2 \frac{i_2}{\sigma_{I_{2s.}}} (\alpha_{22} \sigma_{v_2}^2 + \alpha_{12} Cov(a_2, v_2))$$

and

$$\Delta G_{x_{12}}^G = \theta_2 \phi_2 \frac{i_2}{\sigma_{I_{2s.}}} (\alpha_{22} Cov(v_2, a_2) + \alpha_{12} \sigma_{a_2}^2).$$

Expected responses to selection cumulated over the two parallel selection processes are:

$$\begin{aligned} \Delta G_{x_1}^G &= \Delta G_{x_{11}}^G + \Delta G_{x_{12}}^G, \\ \Delta G_{x_2}^G &= \Delta G_{x_{21}}^G + \Delta G_{x_{22}}^G \end{aligned}$$

and

$$\Delta G^G = \Delta G_{x_1}^G + \Delta G_{x_2}^G,$$

where ΔG^G is the expected response to selection of the performance of the mixture cumulated over the two selection processes and $\Delta G_{x_1}^G$ and $\Delta G_{x_2}^G$ are the cumulated expected responses of contributions to the performance of the mixture of species 1 and 2, respectively. The problem is to find the set of the four index weights that maximizes the cumulated expected response to selection of the mixture performance ΔG^G among those sets meeting the desired ratio of cumulated expected responses of species contributions $(\Delta G_{x_1}^G, \Delta G_{x_2}^G) = c(k1, k2)$.

2. Solving the problem with the Optimization Toolbox from Matlab R2015a We assume that $\theta_1\phi_1 = \theta_2\phi_2 = 1$. The problem is solved using the function *fmincon*.

Let set the following notations:

$$\text{Sigmax1sq} = \sigma_{v_1}^2$$

$$\text{covx1v1} = \text{Cov}(v_1, a_1)$$

$$\text{Sigmaa1sq} = \sigma_{a_1}^2$$

$$w11 = \sigma_{v_1}^2 + \frac{1}{M_1}\sigma_{e_{11}}^2$$

$$w21 = \sigma_{a_1}^2 + \frac{1}{M_1}\sigma_{e_{21}}^2$$

$$w31 = \text{Cov}(v_1, a_1) + \frac{1}{M_1}\text{Cov}(e_{11rm}, e_{21rm})$$

$$\text{Sigmax2sq} = \sigma_{v_2}^2$$

$$\text{covx2v2} = \text{Cov}(v_2, a_2)$$

$$\text{Sigmaa2sq} = \sigma_{a_2}^2$$

$$w22 = \sigma_{v_2}^2 + \frac{1}{M_2}\sigma_{e_{22}}^2$$

$$w12 = \sigma_{a_2}^2 + \frac{1}{M_2}\sigma_{e_{12}}^2$$

$$w32 = \text{Cov}(v_2, a_2) + \frac{1}{M_2}\text{Cov}(e_{22sm}, e_{12sm})$$

$$\text{SigmaI1sq} = \sigma_{I_{1r.}}^2$$

$$\text{DeltaGx11} = \Delta G_{x_{11}}^G$$

$$\text{DeltaGx21} = \Delta G_{x_{21}}^G$$

$$\text{SigmaI2sq} = \sigma_{I_{2s.}}^2$$

$$\text{DeltaGx22} = \Delta G_{x_{22}}^G$$

$$\text{DeltaGx12} = \Delta G_{x_{12}}^G$$

3. See the Matlab scripts on the two next pages

Main Matlab script

```
clear
%DATA INPUT parameters
%the parameters are declared as global because they are used in the function to
%optimise and in the constraint
global i1 Sigmav1sq cova1v1 Sigmaa1sq w11 w21 w31;
i1=value;
Sigmav1sq=value;
cova1v1=value;
Sigmaa1sq=value;
w11=value;
w21=value;
w31=value;

global i2 Sigmav2sq cova2v2 Sigmaa2sq w12 w22 w32 ;
i2=value;
Sigmav2sq=value;
cova2v2=value;
Sigmaa2sq=value;
w12=value;
w22=value;
w32=value;

global k1 k2 ;
k1=value;
k2=value;

alpha0=[1/2,1/2,1/2,1/2];
% this is a starting guess of the solution,
%in order
options = optimoptions(@fmincon,'Algorithm','sqp');
%options for the @fmincon MATLAB function
[alpha,fval] = fmincon(@F,alpha0,[],[],[],[],[],[],@contrainte,options);
%@F is the function to minimise
%alpha0 is the starting guess
%@contrainte is the constraint

'Order : alpha11, alpha21, alpha22, alpha12'
alpha %the solution
fval % value of the function at optimal point

%normalization of the solution alpha
alphabis=alpha;
alphabis(1)=1;
alphabis(2)=alpha(2)/alpha(1);
alphabis(3)=1;
alphabis(4)=alpha(4)/alpha(3);
alphabis

%Computation of the 4 DeltaG
Sigma1sq=alpha(1)^2*w11+alpha(2)^2*w21+2*alpha(1)*alpha(2)*w31;
Sigma1=sqrt(Sigma1sq);
DeltaGx11=(i1/Sigma1)*(alpha(1)*Sigmav1sq+alpha(2)*cova1v1)
DeltaGx21=(i1/Sigma1)*(alpha(2)*Sigmaa1sq+alpha(1)*cova1v1)

Sigma2sq=alpha(3)^2*w22+alpha(4)^2*w12+2*alpha(3)*alpha(4)*w32;
Sigma2=sqrt(Sigma2sq);
DeltaGx22=(i2/Sigma2)*(alpha(3)*Sigmav2sq+alpha(4)*cova2v2)
DeltaGx12=(i2/Sigma2)*(alpha(4)*Sigmaa2sq+alpha(3)*cova2v2)
```

Matlab script defining the function to optimize

```
function y=F(alpha)
%the function to optimise
%the input is the vector alpha of size 4
%in order alpha11, alpha21, alpha22, alpha12
global i1 Sigmav1sq cova1v1 Sigmaa1sq w11 w21 w31;
global i2 Sigmav2sq cova2v2 Sigmaa2sq w12 w22 w32 ;

Signal1sq=alpha(1)^2*w11+alpha(2)^2*w21+2*alpha(1)*alpha(2)*w31;
Signal1=sqrt(Signal1sq);
DeltaGx11=(i1/Signal1)*(alpha(1)*Sigmav1sq+alpha(2)*cova1v1);
DeltaGx21=(i1/Signal1)*(alpha(2)*Sigmaa1sq+alpha(1)*cova1v1);

Signal2sq=alpha(3)^2*w22+alpha(4)^2*w12+2*alpha(3)*alpha(4)*w32;
Signal2=sqrt(Signal2sq);
DeltaGx22=(i2/Signal2)*(alpha(3)*Sigmav2sq+alpha(4)*cova2v2);
DeltaGx12=(i2/Signal2)*(alpha(4)*Sigmaa2sq+alpha(3)*cova2v2);

y=-(DeltaGx11+DeltaGx21+DeltaGx22+DeltaGx12);
```

Matlab script defining the constraint

```
function [c,ceq] = contrainte(alpha)
%the constraint function
%the input is the vector alpha of size 4
%in order alpha11, alpha21, alpha22, alpha12
global i1 Sigmav1sq cova1v1 Sigmaa1sq w11 w21 w31;
global i2 Sigmav2sq cova2v2 Sigmaa2sq w12 w22 w32 ;
global k1 k2 ;
% Nonlinear inequality constraints
c = [];
% Nonlinear equality constraints
Signal1sq=alpha(1)^2*w11+alpha(2)^2*w21+2*alpha(1)*alpha(2)*w31;
Signal1=sqrt(Signal1sq);
DeltaGx11=(i1/Signal1)*(alpha(1)*Sigmav1sq+alpha(2)*cova1v1);
DeltaGx21=(i1/Signal1)*(alpha(2)*Sigmaa1sq+alpha(1)*cova1v1);

Signal2sq=alpha(3)^2*w22+alpha(4)^2*w12+2*alpha(3)*alpha(4)*w32;
Signal2=sqrt(Signal2sq);
DeltaGx22=(i2/Signal2)*(alpha(3)*Sigmav2sq+alpha(4)*cova2v2);
DeltaGx12=(i2/Signal2)*(alpha(4)*Sigmaa2sq+alpha(3)*cova2v2);

ceq = k1*(DeltaGx21+DeltaGx22)-k2*(DeltaGx11+DeltaGx12);
```