

OptOrch:

A modular optimization toolkit for forest tree seed orchard deployment

- From genomic inputs to practical seed orchard decisions

Ye-Ji Kim, Václav Bittner, Kyu-Suk Kang, Christi Sagariya, and Milan Lstibůrek

Presented by. Ye-Ji Kim, PhD

Postdoctoral Researcher
Department of Genetics and Physiology of Forest Trees
Czech University of Life Sciences Prague

Outline

1. Introduction

- 1) Background – why seed orchard deployment matters
- 2) Motivation and objectives – OptOrch overview

2. Methods

- 1) Core mathematical formulation
- 2) Model extensions
- 3) Simulation setup

3. Results

- 1) Computational performance with parameter tuning of solver
- 2) Optimal contributions under simulated scenarios
- 3) Genetic response with pollen contamination

4. Discussion & Conclusions

- 1) Decision-support framework for SO establishment strategies
- 2) Resources available in the repository

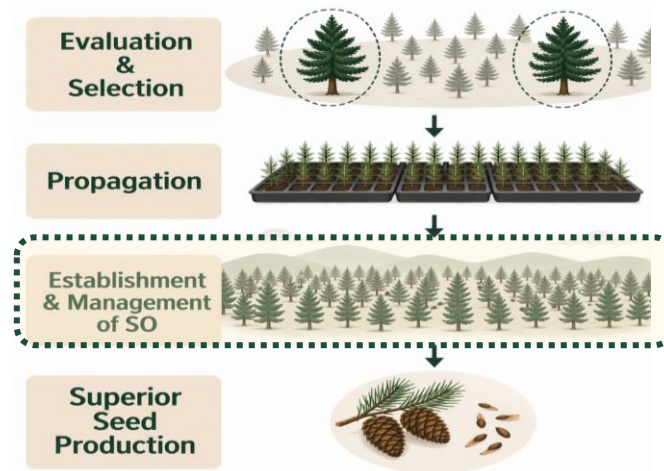
Introduction

Why seed orchard deployment matters

Seed orchards must deliver both superior genetic gain and sufficient diversity.

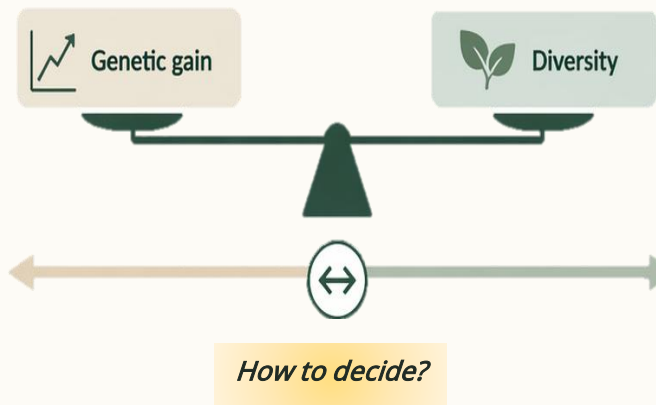
Operational role of seed orchards

- ❖ Seed orchards (SOs) use selected genotypes and distributed clones to promote panmixia and reduce inbreeding risk (Chaloupková et al., 2016; Funda and El-Kassaby, 2012; Lstibůrek and El-Kassaby, 2010).
- ❖ They produce genetically superior forest reproductive material.
- ❖ **Deployment decisions** determine which clones are used and in what proportions.



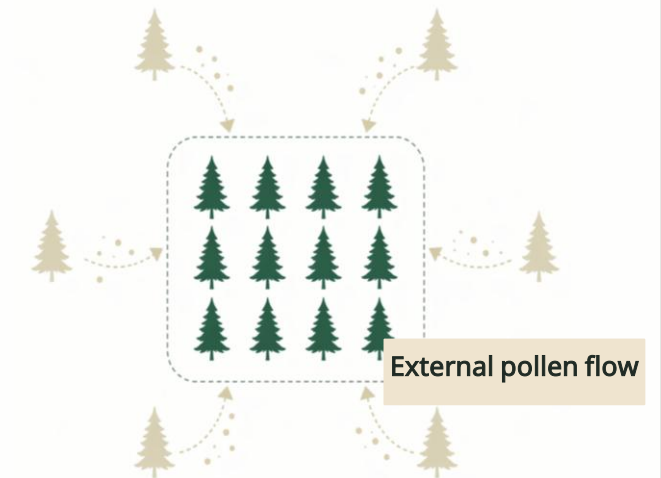
Why the decision is difficult

- ❖ Maintaining diversity in SO crop is essential for preserving adaptive potential under changing environmental conditions.
- ❖ Establishing SOs must **balance genetic gain and gene diversity**.
- ❖ Optimization determines **clone selection** and **relative clonal contribution** (number of grafts assigned to each clone).



Real-world constraints

- ❖ Practical deployment must account for:
 - Orchard size
 - Graft availability
 - Related candidates
 - Female-male specific contributions
 - External pollen flow



Research gap and motivation

OptOrch was designed as a transparent, integrated, and modifiable optimization framework.

Objectives of OptOrch

- Keep the mathematical model explicit and readable.
- Separate model specification, data input, and numerical solver.
- Support realistic biological and operational constraints.
- Provide optimized deployment decisions while accounting for genetic constraints among related clones.

Earlier analytical solutions

Useful for simplified settings, often assuming **unrelated clones** and **limited constraints** (Lindgren and Matheson, 1986). Hard to transfer directly to routine deployment with realistic breeding data.

Existing numerical approaches

Mixed-integer (Funda et al., 2009) and semidefinite (Ahlander et al., 2014) approaches exist, but many studies focused on **selected constraints separately** rather than one integrated toolkit.

Need in practice

Breeders need a framework that can be inspected, modified, and reused across scenarios without rebuilding solver-native data structures.

OptOrch overview

A transparent and solver-independent workflow.



1. AMPL model

- Defines the objective, variables, and constraints.
- Main layer for formulation and modification.



2. Numerical solver

- Solves the instantiated optimization problem.
- Gurobi is used in this study.
- The model is fully compatible with open-source solvers such as SCIP (Achterberg, 2009) and CBC (Forrest and Lougee-Heimer, 2005).



3. R workflow

- Generates inputs and automates repeated runs.
- Processes outputs for interpretation and comparison.

Inputs:

Breeding values

Genomic relationship matrix

Output:

Optimal clonal contributions

Core mathematical formulation – AMPL code

Declaration of parameters, sets, variables, objective function, and constraints in the baseline AMPL model.

AMPL CODE 1 — Baseline OptOrch.mod structure

```
# Parameters and sets
param nc > 0;
set S := 1..nc;
param ns > 0;
param b{i in S}; param l{i in S};
param u{i in S}; param c{i in S,j in S};

# Variables
var p{i in S} >= 0, <= 1;
var r{i in S} binary;

# Objective
maximize gain: sum{i in S} b[i]*p[i];

# Constraints
s.t. statusnumber: ...
s.t. totalcontribution: ...
s.t. lowerbound{i in S}: ...
s.t. upperbound{i in S}: ...
```

Core mathematical formulation

The baseline model maximizes genetic gain through continuous clonal contributions.

Parameters (input)

- $S := \{1, 2, \dots, n_c\}$: set of candidate clones
- n_c : number of candidate clones
- n_s : status number
- b_i : breeding value of clone i
- l_i : lower bound on the contribution of clone i
- u_i : upper bound on the contribution of clone i
- c_{ij} : genomic relationship between clones i and j

Variables (output)

- $p_i \in [0, 1]$: clonal contribution of clone i
- $r_i \in \{0, 1\}$: auxiliary binary variable for clone selection (0 = not selected, 1 = selected)

Objective function - maximizing genetic gain

$$\text{maximize } \Delta G = \sum_{i=1}^{n_c} b_i p_i$$

Constraints

- Limits group coancestry to maintain diversity

$$\sum_{i=1}^{n_c} \sum_{j=1}^{n_c} c_{ij} p_i p_j \leq \frac{1}{2n_s} \quad \forall i, j \in S$$

- Total clonal contribution sums to one

$$\sum_{i=1}^{n_c} p_i = 1$$

- Lower and upper bounds of selected clones

$$l_i r_i \leq p_i \leq u_i r_i \quad \forall i \in S$$

Model extensions - pairwise coancestry

Avoiding joint deployment of closely related clones, even when the overall status-number constraint is satisfied.

Logical formulation

- ❖ If two clones both receive non-zero contributions (selected), their pairwise coancestry must stay below a threshold c_u .
- ❖ This is useful when the solver can directly handle logical or non-convex expressions.

Additional Parameter

- c_u : threshold of pairwise coancestry

Additional Constraint

- Limits pairwise coancestry with a threshold c_u
$$\text{if } (p_i p_j) > 0 \text{ then } c_{ij} \leq c_u$$

Linearized formulation

- ❖ Introduce a binary variable y_{ij} , so that at most one clone from a high-coancestry pair can be selected.

Auxiliary Variable

- $y_{ij} \in \{0,1\}$: auxiliary binary variable for pairwise exclusion

Additional Constraint

- For clone pairs with $c_{ij} > c_u$, at most one clone can be selected

$$p_j \leq 1 - y_{ij} \quad p_i \leq y_{ij} \quad \forall i, j \in S, i < j, c_{ij} > c_u$$

Model extensions - female and male contributions

Split total contribution into sex-specific gametic components.

- ❖ Allows female and male gametic contributions to be modeled separately.
- ❖ Useful when reproductive contribution differs by sex.
- ❖ Particularly relevant for **pollen contamination** or **supplemental mass pollination** scenarios.
- ❖ Also useful for **dioecious** species.

Additional Variables

- $f_i \in [0,1]$: female contribution of clone i
- $m_i \in [0,1]$: male contribution of clone i

Linking Constraint

- Link to total clonal contribution

$$p_i = \frac{f_i + m_i}{2}$$

Model extensions - pollen contamination

Two complementary formulations represent external pollen flow: an explicit donor model or an aggregate adjustment.

Explicit female-male formulation

- ❖ External pollen donors are introduced as **male-only** contributors.
- ❖ SO parents contribute through **both female and male components**.
- ❖ This formulation is **biologically most direct**, but it requires additional decision variables.

Additional Parameter

- n_d : Number of external pollen donors

Additional Constraints

- SO parents' contribution

$$p_i = \frac{f_i + m_i}{2} \quad i = 1, \dots, (n_c - n_d)$$

- No female contribution from external population

$$f_i = 0 \quad i = (n_c - n_d + 1), \dots, n_c$$

- Male contribution from external pollen donors

$$p_i = m_i \quad i = (n_c - n_d + 1), \dots, n_c$$

Aggregate formulation

- ❖ External donors are not modeled explicitly.
- ❖ The formulation adjusts effective within-SO contribution and the coancestry bound.
- ❖ This reduces model complexity and is used in the manuscript analyses.

Additional Parameters

- $C \in \langle 0,1 \rangle$: Pollen contamination rate

Additional Constraints

- Reduced **effective within-SO** contribution

$$\sum_{i=1}^{n_c} p_i = 1 - \frac{C}{2}$$

- Adjusted **coancestry bound** under contamination

$$\sum_{i=1}^{n_c} \sum_{j=1}^{n_c} c_{ij} p_i p_j \leq \frac{1}{2n_s} - \frac{C^2}{8n_d}$$

Simulation scenarios: simulated population and genomic evaluation

To demonstrate the practical utility of OptOrch, optimization was applied to a simulated tree breeding dataset under operational conditions.

1) Population simulation (MoBPS; Pook et al., 2020)

- Founder population: 5,000 diploid individuals
- Genome: 4 chromosomes x 100 cM
- 20,300 SNPs total: 300 QTLs + 20,000 SNP markers
- Allele frequencies sampled from $Beta(1,1)$
- 100 generations of random mating to establish realistic LD

2) Phenotypes and selection

- True breeding value (TBV): sum of additive QTL effects
- Phenotype model: $y_i = \mu + TBV_i + e_i$
- Narrow-sense heritability: $h^2 = 0.2$
- 100 parental trees selected by phenotype
- 500 candidate individuals generated by random mating

3) Genomic relationship matrix

- Computed from SNPs using the method of VanRaden (VanRaden, 2008)

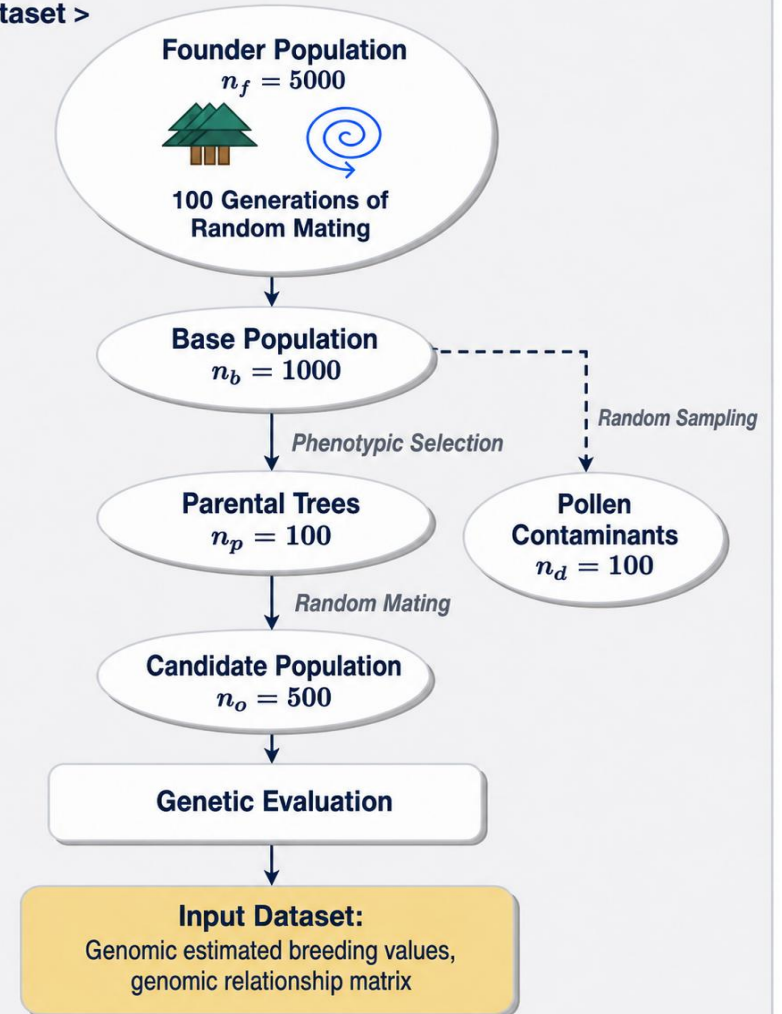
$$G = \frac{ZZ^T}{2 \sum_{k=1}^m p_k(1 - p_k)}$$

- Z , centered genotype matrix; p_k , allele frequency at marker k

4) Genetic evaluation

- ASReml-R (Butler et al., 2023) used for REML estimation and gBLUP prediction
- Model: $y = \mu + u + e$
 $u \sim N(0, G\sigma_a^2), \quad e \sim N(0, I\sigma_e^2)$
- Predicted random effects $\hat{u}$ were treated as GEBVs

< Creating Dataset >



Simulation scenarios: genetic constraints and pollen contamination

Two scenario analysis were conducted to examine how the model responds to different breeding scenarios, thereby providing an intuitive understanding of its practical applicability

Input data

- Genomic breeding values of candidate population
- Genomic relationship matrix

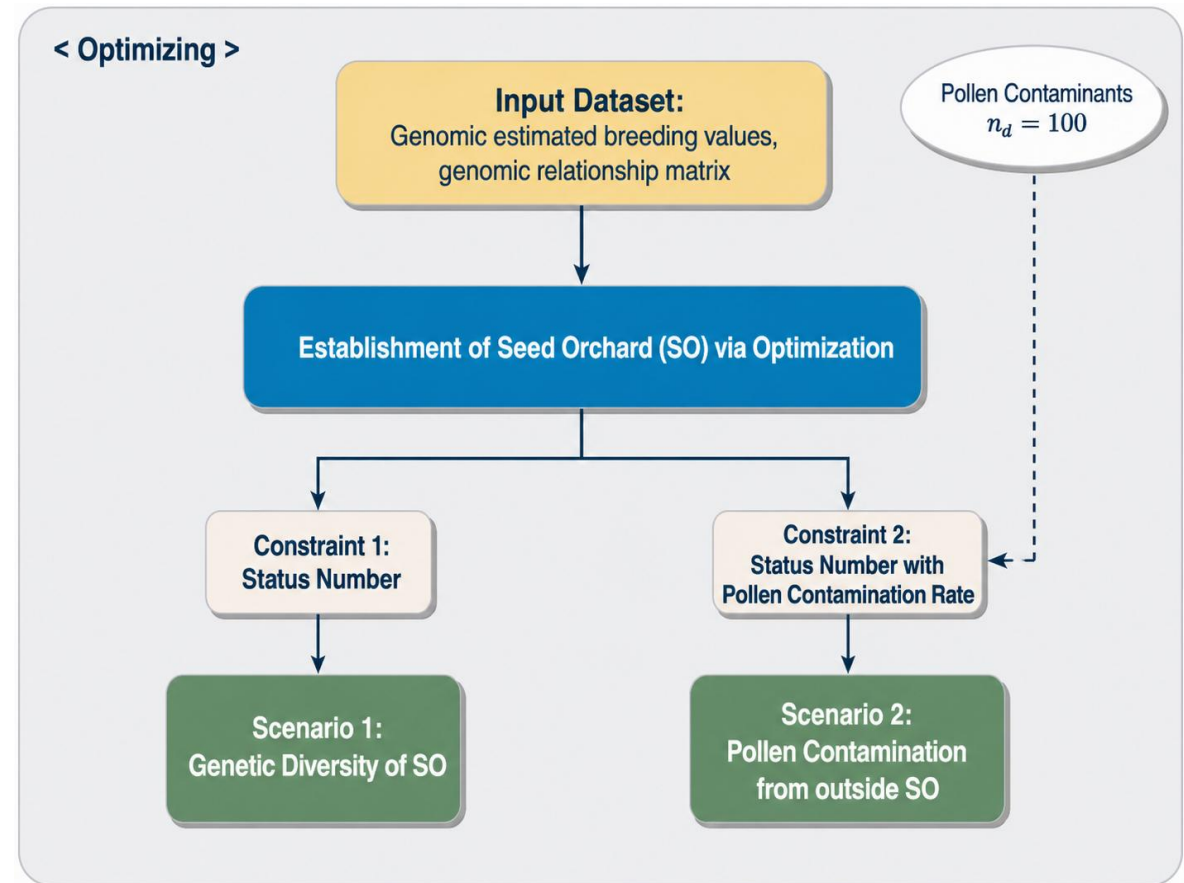
Scenario 1: constraints on status number

- Objective: maximize genetic gain while maintaining diversity
- Status number (n_s) levels: 10, 20, 30, 40
- Contribution bounds: lower bound = 0.01, upper bound = 0.15

Scenario 2: pollen contamination from outside SO

- External pollen donors: 100 individuals
- Pollen contamination rate: 30%
- External male contribution: 0.15
- Within-SO contribution: 0.85
- Compared with Scenario 1 at the same status number level

< Optimizing >



Computational performance and solver parameter tuning

Gurobi parameter tuning was used as a computational demonstration to show how runtime and convergence depended on the OptOrch instance

Key findings

- Tuning consistently reduced runtime at all status number levels.
- At $n_s = 30 - 40$, default settings struggled to close the optimality gap.
- Parameter sets differed by n_s , indicating structural changes in the search space.

Setup

- Tuning limit: 3,600 s per configuration; total tuning time 86,400 s
- Default setting: MIPGap = 0.005, NonConvex = 2
- At $n_s = 30$, one default run exceeded the 3,600 s limit.

Table 1. Optimization time before and after parameter tuning. Each configuration was evaluated three times.

Status number		10	20	30	40
Times (s)	Default	0.87 ± 0.02	4.87 ± 0.34	25.32, 24.51, > 3,600*	> 3,600 (all runs)
	Tuned	0.26 ± 0.00	0.50 ± 0.00	0.75 ± 0.02	929.11 ± 5.04
Tuning parameters		Aggregate 0 NormAdjust 0 NumericFocus 3 PreMIQCPForm 1 ScaleFlag 2	Heuristics 0.5 MIQCPMethod 0 PrePasses 1 Presolve 1 ScaleFlag 0	Aggregate 0 CutPasses 10 Cuts 1 PreMIQCPForm 1 Presolve 1 ScaleFlag 0 ZeroHalfCuts 0	Heuristics 0.001 NoRelHeurTime 900 ScaleFlag 2

Optimal contributions under simulated scenarios

Increasing status number reduces contribution concentration among top-ranking candidates.

Lower and Upper bounds

- The lower bound ($l = 0.01$) prevented trivial near-zero allocations.
- The upper bound ($u = 0.15$) avoided excessive concentration in a few clones.

Genetic diversity and optimal contributions

At $n_s = 10$

- Contributions were highly concentrated among top-ranking candidates.
- Most candidates received contributions close to zero, indicating strong selection intensity.

At $n_s = 40$

- More candidates received non-zero contributions.
- Contributions among selected clones became more evenly distributed.

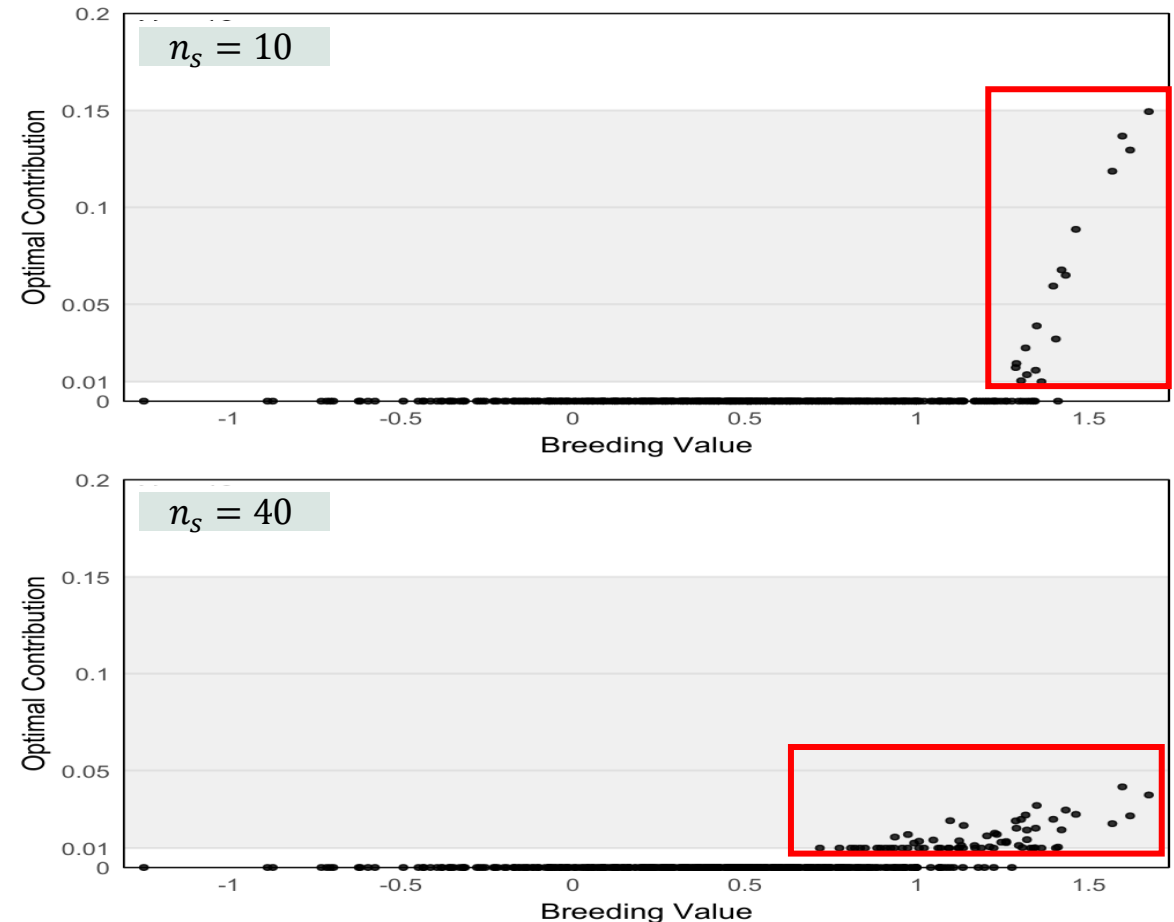


Figure 1. Optimal contributions of the candidate under different status number constraints.

Genetic response with pollen contamination

Genetic response decreased as status number increased, and pollen contamination reduced response across all status-number levels.

Genetic response

- Genetic response decreased as status number increased.
- At $n_s = 10$, genetic response was highest, reflecting strong concentration among top-ranking individuals.
- With contamination, genetic response remained about **86-88%** of the no-contamination scenario across all status number levels.
- External pollen diluted genetic gain, even though within seed orchard contribution was reduced to 0.85.

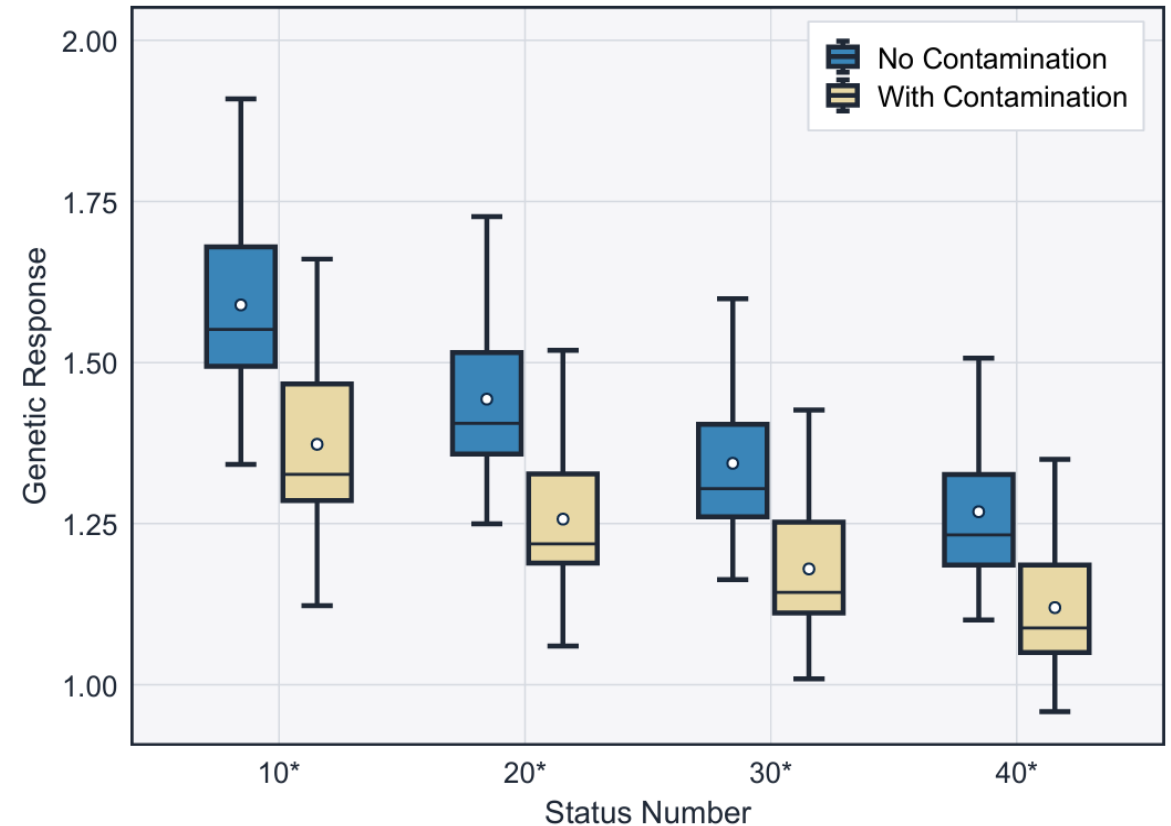


Figure 2. Effect of status number and pollen contamination on genetic response. Asterisks (*) denotes significant differences between scenarios (Mann-Whitney U test, $p < .0001$).

Discussion and Conclusions

Decision-support framework for SO establishment strategies

OptOrch is a transparent framework for comparing genetic gain, diversity, and operational constraints under realistic seed orchard conditions.

1) Flexible and transparent formulation

- AMPL keeps the objective function, variables, and constraints explicit (Fourer et al., 2023).
- Users can inspect and modify the model using algebraic notation.
- The framework can be adapted to different breeding objectives and operational requirements.

2) Quantitative solution assessment

- Mathematical programming provides a reference for solution quality within a predefined optimality tolerance.
- This is useful when multiple biological and operational constraints interact.

3) Realistic biological scenarios

- OptOrch can incorporate practical biological factors such as relatedness among clones, female-male specific contributions, and pollen contamination.
- This allows users to evaluate deployment strategies under conditions that more closely reflect real seed orchard management.

4) Practical use and caution

- Input parameters and constraint combinations strongly affect feasibility and outcomes.
- Overly restrictive bounds or incompatible biological requirements can cause infeasibility.
- Exploring multiple scenarios helps reveal feasible region and deployment trade-offs.

Acknowledgements and code availability

The paper provides open-source AMPL files, runnable examples, and companion R scripts.

Acknowledgements

- Funding: EEA/Norway Grants 2014–2021 and the Technology Agency of the Czech Republic.

Resources available in the repository:

- Core AMPL model
- Supplementary AMPL variants
- Runnable example data and run files
- R scripts for simulation and output processing

For any inquiries, please contact:

- Corresponding author: Prof. Milan Lstibůrek
E-mail: lstiburek@fd.czu.cz
- First author: Dr. Ye-Ji Kim
E-mail: kyeji1107@gmail.com

References

- Achterberg, T., 2009. SCIP: solving constraint integer programs. *Math Program Comput*, 1:1–41.
- Ahlinder, J., Mullin, T., and Yamashita, M., 2014. Using semidefinite programming to optimize unequal deployment of genotypes to a clonal seed orchard. *Tree Genetics & Genomes*, 10:27–34.
- Butler, D., Cullis, B., Gilmour, A., Gogel, B.G. and Thompson, R., 2023. ASReml-R Reference Manual Version 4.2.
- Chaloupková, K., Stejskal, J., El-Kassaby, Y.A., and Lstibůrek, M., 2016. Optimum neighborhood seed orchard design. *Tree Genetics & Genomes*, 12:105.
- Forrest, J., Lougee-Heimer, R., 2005. CBC user guide. *INFORMS TutORials in Operations Research*, pp. 257–277.
- Fourer, R., Gay, D., and Kernighan, B., 2003. *AMPL: A Modeling Language for Mathematical Programming*. 2nd ed. Thomson Brooks/Cole. Pacific Grove, CA.
- Funda, T. and El-Kassaby, Y.A., 2012. Seed orchard genetics. *CABI Reviews*, 7:1–23.
- Funda, T., Lstibůrek, M., Lachout, P., Klápště, J., and El-Kassaby, Y., 2009. Optimization of combined genetic gain and diversity for collection and deployment of seed orchard crops. *Tree Genetics & Genomes*, 5:583–593.
- Lindgren, D., Matheson, A., 1986. An algorithm for increasing the genetic quality of seed from seed orchards by using the better clones in higher proportions. *Silvae Genetica*, 35:173–177.
- Lstibůrek, M., El-Kassaby, Y.A., 2010. Minimum-inbreeding seed orchard design. *Forest Science*, 56:603–608.
- Pook, T., Schlather, M., Simianer, H., 2020. MoBPS-modular breeding program simulator. *G3: Genes, Genomes, Genetics*, 10:1915–1918.
- VanRaden, P., 2008. Efficient methods to compute genomic predictions. *Journal of Dairy Science*, 91:4414–4423.

Thank you for your attention
