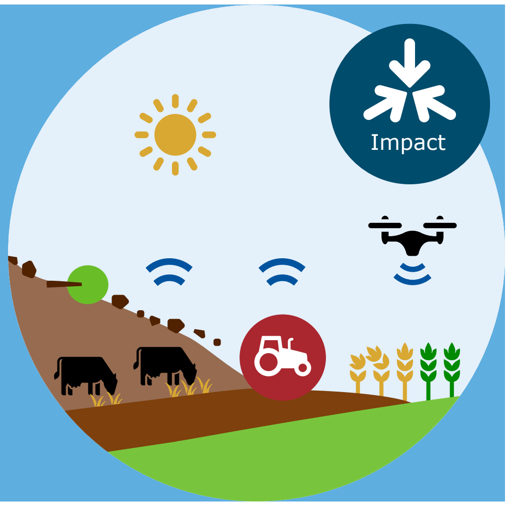
BreedGym

A reinforcement learning environment for plant breeding programs optimization

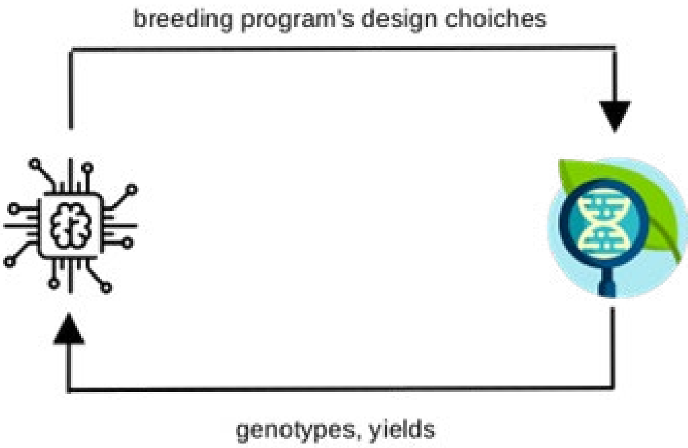
Emerging DS/AI methods

WAGENINGEN
UNIVERSITY & RESEARCH



Data Driven Discoveries in a changing climate (D3C2)

Objective: This project aims to show the potential of the reinforcement learning framework applied to optimize breeding programs' design choices.



Activities and results

Food demand is increasing due to world population growth. On the other hand, breeding programs allow for an increase in food production. They consist of a sequential process of selecting and mating individuals with desirable traits to produce offspring with improved characteristics. However, we need new techniques to double the annual improvement in order to satisfy the increasing demand.

Reinforcement Learning (RL) is a subfield of artificial intelligence that addresses sequential decision-making problems. While interacting multiple times with the problem, the RL agent learns the optimal action at every state. It succeeds in different problems like chess, robotics, and language model instruction (e.g., ChatGPT). However, RL needs many interactions with the problem, while a single breeding program can span multiple years or be potentially endless; we need a breeding simulator for training.

Activities
As a first step, we review the existing simulators. We decided to implement our custom simulator to exploit a typical assumption of RL training setups, i.e. the availability of one or multiple GPUs. Moreover, we left out advanced features for complex breeding programs that are not needed at this early stage and can hurt performances. We developed our simulator using JAX; this allows us to implement a differentiable cross function, where the parents of an individual are chosen as a weighted average instead of a discrete choice. This can foster research on discrete optimization via continuous relaxation for breeding programs.

Having a crossing simulator, we defined the problem using the classic RL interface. First, the agent observes the state of the current population, where "state" can mean genotype or phenotype, depending on the experiment. Depending on the state, the agent decides how to cross the individuals to maximize the yield. This happens at every timestep, i.e. every stage of the breeding program.

Achievement
We compared the performances of our simulator with AlphaSimR, one of the most used ones. When running on a single GPU, our simulator is ~20x faster. This is crucial for RL, which needs many interactions with the problem, but we hope it will benefit the breeding community as well.

We set up a simplified breeding schema in order to quickly test different strategies. We used maize data subsampling one thousand genome markers. The schema span 10 generations with a fixed population size of 200. At each generation, we select the 10 best individuals based on their estimated yield and we make 10 random crosses, each producing 20 offspring. We tried optimizing adaptively some of these design choices; we found out that the severe stochasticity of the outcome is a major obstacle to learning.

The most successful approach was training a neural network to learn a selection index, i.e. a value for each individual, and selecting based on this value. With this approach, we obtained ~7% improvement on the estimated yield of the last generation compared to selecting based on the estimated yield. This is the first AI-based selection index that outperforms existing ones by a sizeable margin in simple breeding schemas.

Outlook
We are preparing a publication for the breeding community to advertise this approach. To be more effective, we aim to apply this method to a more realistic breeding program. To improve our results further, we also want to experiment with new approaches, like learning a score value per possible cross. We hope to continue to strengthen collaboration across artificial intelligence and breeding communities.

Deliverables
The simulator and the RL's environment are in the following repository: https://gitlab.inf.ethz.ch/lucac/breedinggym

The code for running experiments is in this repository: https://github.com/younik/breeding-gym-train

The results of all the experiments are saved on Weights and Biases: https://wandb.ai/younis/BreedingGym

Lesson learned

This interdisciplinary project allows us to learn the basics of genetics and breeding programs, thanks to the meetings with people from the department of Environmental System Science at ETH Zürich and breeders at Wageningen Research. Due to the varying backgrounds of individuals, it was challenging to communicate effectively. To overcome this hurdle, we promptly ask and address any questions or uncertainties while consistently using straightforward language, refraining from technical jargon.

Contact

Prof. Ioannis Athanasiadis
Professor of Artificial Intelligence (project manager)
ioannis.athanasiadis@wur.nl

Wageningen University & Research
P.O. Box 47
6700 AB Wageningen
The Netherlands
T +31 317 48 07 00
www.wur.eu

Project members
Omar Younis, *ETHZ, Dept. computer science, Reinforcement learning*
Prof. Dick de Ridder, *WU, BIF, Bioinformatics*
Prof. Bruno Studer, *ETHZ , Dept. environmental system science, Molecular plant breeding*
Dr. Steven Yates, *ETHZ, Dept. environmental system science, Genomic selection for stress resilience traits*
Dr. Matteo Turchetta, *ETHZ, Dept. computer science, Reinforcement learning*
Dr. Luca Corinzia, *ETHZ, Dept. computer science, Theoretical machine learning, computer vision*