Simulation-based optimization and reinforcement learning methods to improve decision making in agriculture

by

Saba Moeinizade

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Industrial Engineering

Program of Study Committee:
Guiping Hu, Major Professor
Lizhi Wang
Sigurdur Olafsson
Patrick Schnable
Ulrike Genschel

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this dissertation. The Graduate College will ensure this dissertation is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University

Ames, Iowa

2021

DEDICATION

I dedicate my dissertation to victims of Ukraine flight PS752, including all students and professors who never returned home after winter break. We will never know what life-changing contributions they may have made in their areas of study and academic pursuits. Their loss is unfathomable and they shall not be forgotten.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

ACKNOWLEDGMENTS

Throughout the writing of this dissertation, I have received a great deal of support and assistance. I would first like to greatly acknowledge the support of my advisor Dr. Guiping Hu, who has been an invaluable resource over the past five years and always have believed in me. Dr. Hu has encouraged me in all the time of my academic research and daily life. I would also like to thank Dr. Lizhi Wang for his technical support and guidance during the course of my PhD degree. Additionally, I would like to express gratitude to Dr. Patrick Schnable for his treasured support, which was really influential in shaping my interdisciplinary skills. I would also like to thank Dr. Sigurdur Olafsson and Dr. Ulrike Genschel for serving in my committee and contributing to this work. I also appreciate all the support that I received from my mentors, Dr. Hieu Pham and Dr. Ye Han while completing my research internship at Syngenta. Furthermore, I would like to thank Dr. Aaron Kusmec who was always supportive and helped me with designing experiments. My appreciation also goes out to my family and friends for their encouragement and support all through my studies.

ABSTRACT

To address the global challenges of growing population, changing climates, and changing diets agriculture products should be more productive, sustainable, and resilient. To permit more informed decisions in agriculture systems, interdisciplinary efforts are needed. In this dissertation, we use operations research and data analytic techniques to enhance decision making in agriculture systems by tackling specific problems in plant breeding. We design algorithms for improving efficiency in genomic selection, which is a special type of nonlinear, non-convex, high-dimensional, and dynamic optimization problem constrained by resource availability and laws of reproductive biology. Furthermore, we develop an algorithm for selecting cultivars in trait introgression where the goal is to transfer the desirable traits from a donor to an elite line which lacks those desirable traits. The common attribute of the proposed algorithms in this dissertation is taking advantage of simulation to look-ahead and make informed decisions based on the estimated future outcome. We present a family of look-ahead algorithms for optimizing selection and mating decisions in breeding programs and use more advanced optimization techniques such as reinforcement learning to optimally allocate resources during a breeding program. What makes these problems more challenging is the uncertainty due to the recombination events. To capture the uncertainties and characterize the behavior of these complex systems, we develop a stochastic simulation framework. This framework enables testing the proposed algorithms and comparing them with conventional methods by designing case studies using realistic data sets.

More specifically, in chapter 2, we introduce the look-ahead selection algorithm to optimize selection and mating decisions by evaluating the probability of achieving high genetic gains within a specific time. In chapter 3, we propose multi-trait look-ahead selection algorithm, which maximizes certain traits while keeping others within desirable ranges. In chapter 4, a Monte Carlo simulation algorithm is proposed to optimize selection decisions in trait introgression where the objective is

to produce a new line that is highly close to the current available elite line, with the addition of desirable traits from the donor line. In chapter 5, the look-ahead selection algorithm is integrated in the framework of reinforcement learning to optimize resource allocation through different cycles of a breeding program.

CHAPTER 1. GENERAL INTRODUCTION

1.1 Background

In the 21st century, humanity is confronted with the grand challenge of increasing agricultural production to achieve food security and feed a population that is expected to grow to 10 billion people in the coming decades (Rosegrant and Cline, 2003; Bodirsky et al., 2015; Delgado et al., 2019). Moreover, changing climates is putting more pressure on agriculture systems and emphasizes the need for a more productive, sustainable, and resilient system (Suweis et al., 2015; Delgado et al., 2019). Advances in plant genetics have provided new knowledge and technologies needed to address these challenges (Ronald, 2011). Millions of lives depend upon the extent to which crop genetic improvement can keep pace with the growing global population, changing climate, depletion of water resources, and the potential for increased erosion and loss of productivity due to the occurrence of extreme weather events (Ronald, 2011; Delgado et al., 2019). Therefore, genetic improvement of crops is an essential component of agricultural systems.

Plant breeding techniques are widely used in agriculture to enhance the genetic improvement of crop varieties by using principals from a variety of sciences (Bhadouria et al., 2019). The process involves crossing parental plants to obtain the best characteristics for the future generation. Fehr (1991) defined plant breeding as the art and science of genetic improvement of plants and Poehlman and Sleper (1995) defined plant breeding as the art and science of changing the traits of plants to product desired characteristics.

To improve crop production and address the aforementioned global challenges, plant breeding should take advantage of analytics and data-driven solutions (Byrum et al., 2016). Recently, analytical decision making tools have been developed to support the plant breeding industry (Varshney et al., 2016; Pham and Stack, 2018; Moeinizade, 2018; Shahhosseini et al., 2020; Moeinizade et al., 2020b). Operations Research applies scientific and analytical methods to enable better decisions,

which result in reducing the time and cost required to develop higher-productivity crops. (Morse et al., 2003; Byrum et al., 2016). With the growing need for plant breeding in the agriculture industry and increasing complexity in decision making, the role of operations research has become more important. This dissertation focuses on filling the gaps between analytics and plant breeding by exploring operations research, simulation-based optimization and data data analytic methods to enhance efficiency in plant breeding.

1.2 Problem Statement

Phenotypic selection has been used successfully in plant breeding for thousands of years (Hallauer, 2011). In phenotypic selection, the trait values of a population such as grain yield, plant height, flowering time, and disease resistance are measured and then individuals are selected based on their trait values. This process can be time consuming, expensive, labor intensive and even destructive to plants (Akdemir et al., 2015; Ali et al., 2020).

Breeding methods have evolved from traditional phenotype-based selection to marker-assisted selection methods. Genomic selection (GS), which was initially proposed by Meuwissen et al. (2001), is a special form of marker assisted selection that estimates the effects of genome-wide markers in a training population consisting of genotyped and phenotyped individuals. Different statistical and machine learning models are proposed to develop prediction models based on the genotypic and phenotypic data of the training population (Neves et al., 2012). Then, the prediction model is used to derive the genomic estimated breeding values (GEBVs) for all individuals of the breeding population (BP) from their genomic profile by calculating the sum of the estimated marker effects. Given the genotype information and the estimated marker effects of individuals in a breeding population, there are different decisions that should be made within each breeding cycle. These decisions are as follows:

- Identification of individuals within the population that should be selected and crossed as breeding parents to produce the next generation of individuals
- The mating strategy of the selected parents

- Resource allocation decisions including budget allocation among generations, number of crosses and the number of progeny to be made from each cross

These decisions must be made in every generation with the objective of continuously improving varieties subject to deadline constraints. Having a large number of alleles across the genome with varying contributions and different recombination possibilities makes this problem more challenging. Furthermore, during a breeding program, it may be necessary to introduce new traits (e.g., disease resistance) to different individuals that lack the desirable traits. This process, known as Trait introgression (TI) involves introgressing desirable traits from a donor to a recipient. The goal of this procedure is to recover all attributes of the recipient cultivar, with the addition of the specified desirable traits from the donor (Hospital et al., 1992). Although, in principle, the intent of trait introgression is forthright, in practice, there exists many complications due to the stochastic nature and size of a commercial breeding program. Because of this uncertainty, multiple breeding generations may be required until the superior desired cultivar is achieved. An additional challenge of the TI process is deciding the parental crosses to perform out of a sizable, complex gene pool. Therefore, there is a need to investigate efficient selection strategies in TI programs.

To address the aforementioned challenges in genomic selection, we develop new GS strategies that rely on improving the selection and mating steps considering both single and multiple traits and introduce a new reinforcement learning algorithm to efficiently allocate resources given a specified breeding deadline. Additionally, to overcome the challenges in trait introgression, we develop a new selection strategy using Monte Carlo simulation to efficiently introgress desirable traits from donor to a recipient.

1.3 Summary of Contributions and Dissertation Organization

The objective of this study is to increase the rate of genetic gain in crop breeding programs by developing improved genomic selection and trait introgression strategies. This dissertation consists of four papers.

The first paper proposes a new selection strategy, look-ahead selection (LAS) to optimize selection and mating strategies in genomic selection considering one characteristic. The LAS method introduces a novel stochastic simulation method to estimate the performance of progeny in the targeted generation for a given selection and mating strategy and then optimizes these steps using a heuristic algorithm. Moreover, this study proposes a heuristic strategy to enhance genetic diversity in a breeding population. The main achievements of the LAS method include making a trade-off between short-term genetic gain and long-term growth potential by taking time into account and recognising the importance of mating decisions.

The second paper extends the LAS algorithm to multiple trait settings and proposes the multi-trait look-ahead selection (MT-LAS) method to make a trade-off between different characteristics. The MT-LAS method optimizes genetic gain with respect to a focal trait while controlling the variation in multiple secondary traits. It retains the advantages of single-trait LAS derived from considering the impacts of selection and mating decisions on the performance of individuals in the terminal generation of the breeding program.

The third paper investigates the selection strategies in trait introgression problem by developing a Monte Carlo simulation method. In this study, we have simulated the back-crossing process which is a well-known breeding approach that can be employed to introduce a specific trait, such as disease resistance, from one individual, often an unimproved one, to another individual that is typically an elite breeding individual. The proposed method, look-ahead Monte Carlo simulation method (LMC) has the potential to further improve the efficiency of trait introgression projects and was compared to other state-of-the-art approaches under different scenarios of resources.

Finally, the last paper develops a reinforcement learning based model for addressing resource allocation in genomic selection. At each selection cycle, breeders are facing the choice of budget allocation to make crosses and produce the next generation of breeding parents. We integrate the look-ahead simulation in a reinforcement learning framework to automatically learn to allocate limited resources across different generations of breeding. To do so, we first formulate the problem in the framework of Markov Decision Process (MDP) by defining the state and action spaces and

then use value function approximation techniques along with greedy policy improvement to optimize the allocation strategies.

The first paper, published in *G3: Genes, Genomes, Genetics* is presented in Chapter 2 (Moeinizade et al., 2019). The second paper, published in *Genetics* is presented in Chapter 3 (Moeinizade et al., 2020a). The third paper, published in *Nature Scientific Reports*, is presented in Chapter 4 (Moeinizade et al., 2021) and the last paper which is in preparation is presented in Chapter 5. Finally, conclusions and future research suggestions are outlined in Chapter 6.

1.4 References

Akdemir, D., Sanchez, J. I., and Jannink, J.-L. (2015). Optimization of genomic selection training populations with a genetic algorithm. *Genetics Selection Evolution*, 47(1):38.

Ali, M., Zhang, L., DeLacy, I., Arief, V., Dieters, M., Pfeiffer, W. H., Wang, J., and Li, H. (2020). Modeling and simulation of recurrent phenotypic and genomic selections in plant breeding under the presence of epistasis. *The Crop Journal*.

Bhadouria, R., Singh, R., Singh, V. K., Borthakur, A., Ahamad, A., Kumar, G., and Singh, P. (2019). Agriculture in the era of climate change: Consequences and effects. In *Climate Change and Agricultural Ecosystems*, pages 1–23. Elsevier.

Bodirsky, B. L., Rolinski, S., Biewald, A., Weindl, I., Popp, A., and Lotze-Campen, H. (2015). Global food demand scenarios for the 21 st century. *PloS One*, 10(11):e0139201.

Byrum, J., Davis, C., Doonan, G., Doubler, T., Foster, D., Luzzi, B., Mowers, R., Zinselmeier, C., Kloeber, J., Culhane, D., et al. (2016). Advanced analytics for agricultural product development. *Interfaces*, 46(1):5–17.

Delgado, J. A., Short, N. M., Roberts, D. P., and Vandenberg, B. (2019). Big data analysis for sustainable agriculture on a geospatial cloud framework. *Frontiers in Sustainable Food Systems*, 3:54.

Fehr, W. (1991). *Principles of cultivar development: theory and technique*. Macmillian Publishing Company.

Hallauer, A. R. (2011). Evolution of plant breeding. *Crop Breeding and Applied Biotechnology*, 11(3):197–206.

Hospital, F., Chevalet, C., and Mulsant, P. (1992). Using markers in gene introgression breeding programs. *Genetics*, 132(4):1199–1210.

Meuwissen, T. H. E., Hayes, B. J., and Goddard, M. E. (2001). Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157(4):1819–1829.

Moeinizade, S. (2018). A stochastic simulation approach for improving response in genomic selection. *Master's thesis,*, page Iowa State University.

Moeinizade, S., Han, Y., Pham, H., Hu, G., and Wang, L. (2021). A look-ahead monte carlo simulation method for improving parental selection in trait introgression. *Scientific reports*, 11(1):1–12.

Moeinizade, S., Hu, G., Wang, L., and Schnable, P. S. (2019). Optimizing selection and mating in genomic selection with a look-ahead approach: An operations research framework. *G3: Genes, Genomes, Genetics*, 9(7):2123–2133.

Moeinizade, S., Kusmec, A., Hu, G., Wang, L., and Schnable, P. S. (2020a). Multi-trait genomic selection methods for crop improvement. 215(4):931–945.

Moeinizade, S., Wellner, M., Hu, G., and Wang, L. (2020b). Complementarity-based selection strategy for genomic selection. *Crop Science*, 60(1):149–156.

Morse, P. M., Kimball, G. E., and Gass, S. I. (2003). *Methods of operations research*. Courier Corporation.

Neves, H. H., Carvalheiro, R., and Queiroz, S. A. (2012). A comparison of statistical methods for genomic selection in a mice population. *BMC Genetics*, 13(1):100.

Pham, X. and Stack, M. (2018). How data analytics is transforming agriculture. *Business Horizons*, 61(1):125–133.

Poehlman, J. and Sleper, D. (1995). Breeding field crops, (1995).

Ronald, P. (2011). Plant genetics, sustainable agriculture and global food security. *Genetics*, 188(1):11–20.

Rosegrant, M. W. and Cline, S. A. (2003). Global food security: challenges and policies. *Science*, 302(5652):1917–1919.

Shahhosseini, M., Hu, G., Archontoulis, S. V., and Huber, I. (2020). Coupling machine learning and crop modeling improves crop yield prediction in the us corn belt. *arXiv:2008.04060*.

Suweis, S., Carr, J. A., Maritan, A., Rinaldo, A., and D'Odorico, P. (2015). Resilience and reactivity of global food security. *Proceedings of the National Academy of Sciences*, 112(22):6902–6907.

Varshney, R. K., Singh, V. K., Hickey, J. M., Xun, X., Marshall, D. F., Wang, J., Edwards, D., and Ribaut, J.-M. (2016). Analytical and decision support tools for genomics-assisted breeding. *Trends in Plant Science*, 21(4):354–363.

CHAPTER 2. OPTIMIZING SELECTION AND MATING IN GENOMIC SELECTION WITH A LOOK-AHEAD APPROACH: AN OPERATIONS RESEARCH FRAMEWORK

Saba Moeinizade[1], Guiping Hu[1], Lizhi Wang[1], and Patrick S. Schnable[2]

[1]Department of Industrial and Manufacturing Systems Engineering, Iowa State University,

[2]Department of Agronomy, Iowa State University

Modified from a manuscript published in *G3: Genes, Genomes, Genetics*

2.1 Abstract

New genotyping technologies have made large amounts of genotypic data available for plant breeders to use in their efforts to accelerate the rate of genetic gain. Genomic selection (GS) techniques allow breeders to use genotypic data to identify and select, for example, plants predicted to exhibit drought tolerance, thereby saving expensive and limited field-testing resources relative to phenotyping all plants within a population. A major limitation of existing GS approaches is the trade-off between short-term genetic gain and long-term potential. Some approaches focus on achieving short-term genetic gain at the cost of reduced genetic diversity necessary for long-term gains. In contrast, others compromise short-term progress to preserve long-term potential without consideration of the time and resources required to achieve it. Our contribution is to define a new "look-ahead" metric for assessing selection decisions, which evaluates the probability of achieving high genetic gains by a specific time with limited resources. Moreover, we propose a heuristic algorithm to identify optimal selection decisions that maximize the look-ahead metric. Simulation results demonstrate that look-ahead selection outperforms other published selection methods.

2.2 Introduction

Feeding the world's growing population remains a significant challenge. Advances in plant breeding have been instrumental in improving agricultural output. Classical plant breeding programs rely on the phenotyping of progenies in field trials to identify superior individuals. The number of individuals that can be phenotyped is resource limited (Rincent et al., 2017), which limits genetic gain. Genomic selection (GS) refers to using a set of markers distributed across the genome to estimate the breeding value of selection candidates for quantitative traits (Goddard, 2009). GS makes it possible to predict the performance of unphenotyped individuals from readily available genotyping data (Rincent et al., 2017; Meuwissen et al., 2001). Genomic Estimated Breeding Value (GEBV) of individual plants (or animals) has been widely adopted as the selection criteria; it selects individuals based on the sum of their estimated marker effects (Meuwissen et al., 2001). This approach has been widely adopted in GS practice due to its effectiveness in achieving short-term genetic improvements. More recently, two methods have been proposed to improve conventional GS (CGS): the optimal haploid value (OHV) (Daetwyler et al., 2015) and the optimal population value (OPV) (Goiffon et al., 2017). Simulation experiments and some empirical studies have shown that CGS selection results in rapid genetic gains (Hayes et al., 2009; Lorenzana and Bernardo, 2009; VanRaden et al., 2009; Jannink, 2010). However, CGS focuses on one or two cycles of selection and does not guarantee long-term gain (Sonesson et al., 2012; Lin et al., 2017; Gorjanc et al., 2018; Akdemir et al., 2018). The OHV method, calculates the GEBV of the best possible doubled haploid (DH) derived from an individual (Daetwyler et al., 2015). This method focuses selection on haplotypes and optimizes the breeding program toward the end goal of generating an elite fixed line (Daetwyler et al., 2015). Simulation studies have shown that OHV selection results in more genetic gain and diversity as compared to CGS (Daetwyler et al., 2015). CGS and OHV are truncation selection approaches in that they rank *individuals* and select the top fraction of the population. In contrast, OPV is a group-based selection strategy. Specifically, OPV selects the best *group* of individuals based on their interactive effects and calculates the GEBV of the best possible progeny from this group produced after an unlimited number of generations, which may require

a large amount of time and resources to achieve (Goiffon et al., 2017). In this paper, we extend OPV by again selecting groups of individuals as a unit, but propose an innovative method for selecting groups, "look-ahead selection" (LAS). This new selection method can improve genetic gain by maximizing the expected GEBV of the best offspring in the terminal generation. It makes the optimal trade-off between short-term gain and long-term potential to achieve the highest genetic gain within a specified time.

2.3 Materials and Methods

2.3.1 Generic Formulation for GS Methods

In this section, we present a generic formulation for existing GS methods namely, CGS, OHV, OPV, and the new selection method, LAS. Equations (2.1), (2.2), and (2.3) show this genetic optimization formulation.

$$\max_x \quad f(x) \tag{2.1}$$

$$\text{s.t.} \quad \sum_{n=1}^{N} x_n = S \tag{2.2}$$

$$x_n \in \{0,1\}, n \in \{1,...,N\} \tag{2.3}$$

Here,

- N is the number of individuals in the population.
- x_n is a binary decision variable that shows whether individual n is selected ($x_n = 1$) or not ($x_n = 0$).
- S is the number of individuals that are to be selected out of the current population.

It should be observed that the only difference among the three previous methods is in their objective functions as they aim to maximize different objectives. The objective function of the optimization problem, $f(x)$ is formulated as $f(x)^{CGS}$, $f(x)^{OHV}$, and $f(x)^{OPV}$ in equations (2.4), (2.5), and (2.6) respectively.

2.3.1.1 Conventional genomic selection:

Meuwissen et al. (2001) proposed to evaluate an individual as a breeding parent by its genomic estimated breeding value (GEBV), which is the sum of all marker effects across the entire genome, as defined in equation (2.4). The CGS method selects individuals with the highest GEBVs.

$$f(x)^{\mathrm{CGS}} = \sum_{n=1}^{N} \sum_{l=1}^{L} \sum_{m=1}^{2} G_{l,m,n} \beta_l x_n. \tag{2.4}$$

Here, the notations are defined as follow:

- L: The number of marker loci.
- $G_{l,m,n} \in \{0,1\}, \forall l \in \{1,2,...,L\}, \forall m \in \{1,2\}$ and $\forall n \in \{1,2,...,N\}$: The genotypic information of locus l from chromosome m of individual n, with 1 and 0 representing the major and minor allele, respectively.
- β_l: The normalized effect of the major allele at locus l, with that for the minor allele being 0.
- M: The ploidy of the plants. We use diploid species (M=2) as an example in this paper.

To maximize long-term response, the weighted genomic selection (Goddard, 2009; Jannink, 2010) was proposed as a variation of the CGS method by emphasizing the preservation of rare favorable alleles. It replaced the allele effect β_l in equation (2.4) with $\frac{\beta_l}{\sqrt{\max(w_l, 1/N)}}$, where w_l is the frequency of favorable alleles at locus l among all individuals in the population. As such, this variation gives a higher weight to low-frequency favorable alleles. Notice that the denominator $\sqrt{\max(w_l, 1/N)}$ is equal to $\sqrt{w_l}$ except for $w_l = 0$ when $G_{l,m,n} = 0$ for all m and n.

2.3.1.2 Optimal haploid value:

More than a decade after the CGS metohd, OHV was proposed to combine the creation of doubled haploids with GS methods and evaluates the potential of producing elite doubled haploids (Daetwyler et al., 2015). Equation (2.5) shows the objective function for OHV selection. This method selects individuals with the highest OHVs.

$$f(x)^{\mathrm{OHV}} = 2\sum_{n=1}^{N}\sum_{b=1}^{B}\max_{m\in\{1,2\}}\sum_{l\in H(b)} G_{l,m,n}\beta_l x_n. \tag{2.5}$$

Here, segments of adjacent markers are clustered into haplotypes, which are defined as follows:

- B: The number of haplotype blocks per chromosome.
- $H(b), \forall b \in \{1, ..., B\}$: The set of marker loci that belong to haplotype block b.

The OHV of an individual is the GEBV of its best possible DH progeny. Recombination events are assumed to be possible between haplotypes but not within them. This assumption reduces the computational effort of the algorithm.

What also makes CGS and OHV computationally efficient is the fact that they are both truncation selection methods, which assumes that the contribution of breeding parents are separable and additive. Mathematically, the summation operator $\sum_{n=1}^{N}$ in equation (2.4) and (2.5) suggests that the maximization of the objective functions $f(x)^{\mathrm{CGS}}$ or $f(x)^{\mathrm{OHV}}$ can be easily achieved by evaluating each individual n separately and setting $x_n = 1$ for the ones with the highest GEBVs or OHVs.

Compared with CGS, OHV represents an important shift of the selection objective from maximizing genetic achievement of the parents to that of their progeny.

2.3.1.3 Optimal population value:

OPV selection is an extension to OHV which evaluates the breeding merit of a set of individuals instead of evaluating the breeding value of a single individual (Goiffon et al., 2017). The OPV of breeding population S is the GEBV of the best possible progeny produced after an unlimited number of generations. The objective function for the OPV method is defined as follows:

$$f(x)^{\mathrm{OPV}} = 2\sum_{b=1}^{B}\max_{n\in\{1,\ldots,N\}}\max_{m\in\{1,2\}}\sum_{l\in H(b)} G_{l,m,n}\beta_l x_n. \tag{2.6}$$

OPV represents another important shift of the selection objective from individual-based truncation selection to group-based selection. The contribution of a breeding parent is evaluated based

on not only the favorable alleles that it carries but also the favorable alleles that it carries but are missing in other selected breeding parents. A limitation of OPV is that the objective function $f(x)^{\text{OPV}}$ is a lot harder to optimize, since it is no longer separable with respect to x. As a result, heuristic algorithms were used to identify good but not necessarily optimal selections.

2.3.2 Potential Improvements

The success of CGS has been demonstrated in numerous simulation and field experiments, especially in achieving short-term genetic gains in both plant and animal breeding (Meuwissen, 1997; Rosvall, 1999; Hayes et al., 2002; Ullrich, 2007; Hayes et al., 2009; Lorenzana and Bernardo, 2009; VanRaden et al., 2009; Jannink et al., 2010; Mujibi et al., 2011; Nakaya and Isobe, 2012; Hallatschek and Geyrhofer, 2015). OHV and OPV were proposed as extensions of CGS to improve long-term genetic gains, which have been shown to be effective in simulation studies. Herein, we identify three areas in genomic selection that can be made more efficient and present a new genomic selection method that attempts to address each of these three areas.

First, time management. For a given population of individuals, the optimal selection decision should depend on whether the deadline of the breeding project is in the near future or far down the road. However, none of the aforementioned three methods take deadlines into consideration.

Second, mating strategy. All three methods focus on selecting breeding parents without explicitly indicating how they should be mated in pairs, but several studies have observed that different mating decisions may affect genetic gain (Toro and Varona, 2010; Kinghorn, 2011; Sun et al., 2013; Akdemir and Sánchez, 2016; Liu et al., 2017; Wang et al., 2018).

Third, resource allocation. Intuitively, making more crosses and producing more progenies leads to a higher chance of creating outstanding individuals from the progeny population, but this also requires more resources. Allocating a fixed total budget over a period of time to achieve the best final outcome is therefore a strategic decision that needs to be optimized (Lorenz, 2013).

2.3.3 Look-ahead Selection

The cornerstone of the LAS method is a new definition of the objective function, $f^{\text{LAS}}(x, y, r, T-t)$, that reflects what truly matters in genomic selection. The input of this function includes selected breeding parents (x), mating decisions (y), recombination frequencies (r), and remaining number of generations ($T-t$, the difference between the current generation number t and the deadline T). The former two input terms are decision variables that need to be optimized by the model, whereas the latter two are parameters that the model needs to take into account when searching for the optimal solution. We define f^{LAS} as the *expected GEBV of the best offspring in the terminal generation*. In comparison, f^{CGS} can be interpreted as the genetic achievement of the breeding parents measured in terms of GEBV; and f^{OHV} and f^{OPV} represent the best possible progeny that can be produced by, respectively, self pollination and cross pollination, both assuming unlimited time and resources. The models for these three methods only differ in the objective functions but share the same constraints (2.2) and (2.3), whereas the LAS model requires additional constraints. The LAS method can be formulated as follows.

$$\max_{x,y} \quad f^{\text{LAS}}(x, y, r, T-t) \tag{2.7}$$

$$\text{s.t.} \quad \text{Constraints (2.2) and (2.3)} \tag{2.8}$$

$$x_n = \textstyle\sum_{j=1}^{N} y_{n,j} \qquad \forall n \in \{1, ..., N\} \tag{2.9}$$

$$y_{i,j} \in \{0, 1\} \qquad \forall i, j \in \{1, ..., N\} \tag{2.10}$$

The new variables and parameters are defined as follows.

- $y_{i,j}$: A binary variable that shows whether individual i is mated with individual j ($y_{i,j} = 1$) or not ($y_{i,j} = 0$).
- $r \in [0, 0.5]^{L-1}$: The recombination frequency vector.

The remainder of this section will explain how to numerically evaluate the objective function $f^{\text{LAS}}(x, y, r, T-t)$ for any given solution (x, y), how to search for the optimal (or close to optimal)

solution (x^*, y^*) that achieves the maximal value in the objective function, and how to allocate resources to improve the rate of genetic gains.

2.3.3.1 Evaluation of the objective function $f^{\mathbf{LAS}}$:

The exact evaluation of the objective function f^{LAS} is challenging both computationally and analytically due to uncertain recombination events over $T-t$ generations as well as the selection, mating, and resource allocation decisions that will be made therein. To overcome this challenge, we designed a novel simulation method that provides a computationally tractable yet reasonable approximation of the true objective function. Figure 2.1 illustrates the look-ahead simulation that is based on two simplifying assumptions.

Assumption 1: The selected pairs of breeding parents will each produce one progeny in generation $t+1$.

Assumption 2: All progenies from generation $t+1$ to $T-1$ were crossed with each other (including selfing) in the same generation, each producing one progeny.

As such, the objective function f^{LAS} can be approximated by taking a random sample of the population in generation T of the look-ahead simulation and calculating the highest GEBV of all individuals.

The following theorem defines the distribution of the progenies in the final generation T, which allows efficient evaluation of the approximated objective function.

Theorem 1 *Let $G \in \{0,1\}^{L\times 2\times S}$ denote the genotype of a population in generation t with an even number, S, of individuals. Suppose all individuals with odd indices, $\{1,3,...,S-1\}$, are respectively mated with the next individuals, $\{2,4,...,S\}$. These individuals are mated according to Assumptions 1 and 2. Let $g \in \{0,1\}^L$ denote a random gamete produced by breeding parents in meiosis of the $(T-1)$st generation. The distribution of g can be described by the following equations (2.11) and (2.12):*

$$P(g_1 = G_{1,m,i}) = \frac{1}{2S}, \forall i \in \{1,2,...,S\}, \forall m \in \{1,2\}. \tag{2.11}$$

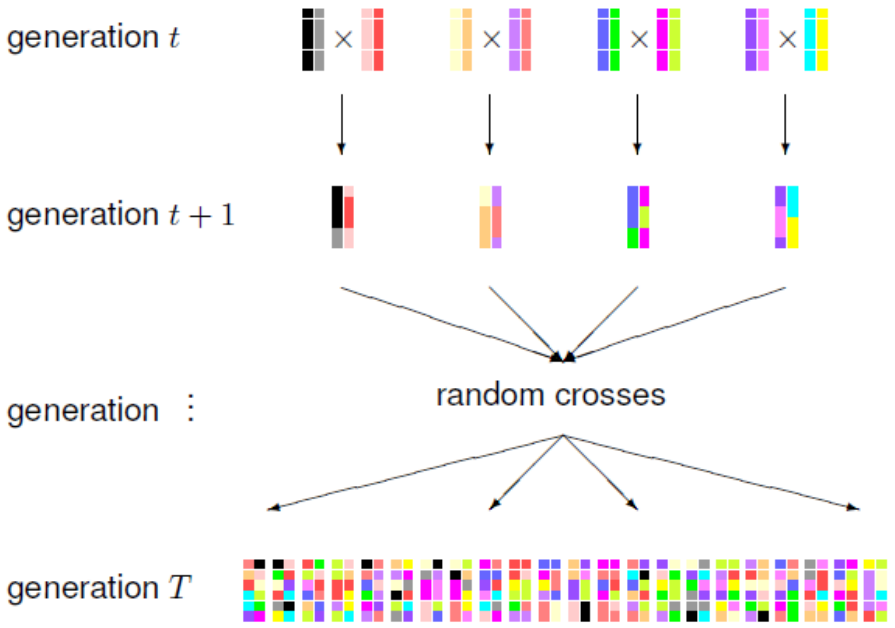

Figure 2.1 The look-ahead simulation.

$$P(g_{l+1} = G_{l+1,m^1,i^1} | g_l = G_{l,m^0,i^0}) = \begin{cases} (1-r_{l+1})^2(1-R_{l+1}), & \text{if } i^0 = i^1 \text{ and } m^0 = m^1 \\ r_{l+1}(1-r_{l+1})(1-R_{l+1}), & \text{if } i^0 = i^1 \text{ and } m^0 \neq m^1 \\ \frac{1}{2}\, r_{l+1}(1-R_{l+1}), & \text{if } \left\lceil \frac{i^0}{2} \right\rceil = \left\lceil \frac{i^1}{2} \right\rceil \\ \frac{R_{l+1}}{2(S-2)}, & \text{if } \left\lceil \frac{i^0}{2} \right\rceil \neq \left\lceil \frac{i^1}{2} \right\rceil \end{cases},$$

$$\forall l \in \{1, ..., L-1\},\ \forall i^0, i^1 \in \{1, 2, ..., S\},\ \forall m^0, m^1 \in \{1, 2\}.$$

Here, $r \in [0, 0.5]^{L-1}$ is the given vector of recombination frequencies and R_l is the recombination frequency between allele l and allele $l+1$ between generations $t+2$ and T for all $l \in \{1, ..., L-1\}$, which can be derived as:

$$R_l = \frac{(S-2)\left[1-(1-r_l)^{T-t}\right]}{S}. \tag{2.12}$$

The proof for equation (2.12) is provided in the appendix 2.7.

2.3.3.2 Optimization of the objective function $f^{\mathbf{LAS}}$:

Unlike truncation selection methods CGS and OHV, which are easy to optimize due to separable objective functions with respect to the selection decision x, the OPV and LAS methods require the optimization of the selected breeding parents' synergistic contribution. A heuristic algorithm was designed to optimize f^{OPV} in Goiffon et al. (2017), where a randomly selected set of breeding parents is iteratively updated to maximize the f^{OPV} function through pairwise swaps between a selected individual and every other unselected one. A similar heuristic can also be applied to optimize the f^{LAS} function with two minor points of caution. First, OPV only selects individuals, while in contrast, LAS also pairs them up, so the orders of the selected individuals in generation t must be preserved to reflect the mating strategy.

Second, constraint (2.2) ensures fair comparison between the four methods by specifying the number of selected individuals. This constraint helps CGS and OHV by maintaining genetic diversity. On the other hand, maintaining genetic diversity is a built-in feature in OPV and LAS methods. Hence, the decision maker can choose to relax constraint (2.2) on OPV or LAS methods in cases that selfing or polygamous crosses are beneficial.

2.3.3.3 Heuristic strategy for resource allocation:

There are two dimensions of resource allocation in genomic selection (beyond genomic prediction of allele effects): allocation of total budget across a number of generations and allocation of the given budget for a specific generation over multiple crosses. In this paper, we assumed equal temporal allocation of the total budget over the breeding duration and hence a fixed number of crosses and population size for each generation. The proposed heuristic strategy attempted to accelerate the rate of genetic gain by strategically varying the numbers of progenies produced from different crosses based on the genetic diversity of the breeding parents. Let n_1 and n_2 be the indices of the two breeding parents (that have been selected and paired according to the LAS method) in

the current generation with G representing its genotype, then the genetic diversity is defined as

$$\sum_{l} \left(\max_{\substack{n\in\{n_1,n_2\} \\ m\in\{1,2\}}} G_{l,m,n}\beta_l - \min_{\substack{n\in\{n_1,n_2\} \\ m\in\{1,2\}}} G_{l,m,n}\beta_l \right), \tag{2.13}$$

which is the aggregated range of GEBVs over all haplotype blocks. Given a fixed budget for the current generation, the numbers of progenies produced from multiple crosses are set to be proportional to the genetic diversity measures of the breeding parents. The rationale for this heuristic is to spend more resources on those crosses that have wider predicted phenotypic distributions and thus higher probabilities of producing outstanding progenies.

2.3.4 Data Availability

All data including phased single nucleotide polymorphisms (SNPs) for maize inbred lines from the Shoot Apical Meristem (SAM) Diversity Panel and genetic maps are available at Figshare: https://iastate.figshare.com/s/ 374176500b04fd6f3729.

2.4 Results

2.4.1 Simulation Setting

In this paper, the genotypic data ($G_{l,m,n}$), marker effects (β_l) and recombination rates (r_l) are based on Goiffon et al. (2017). The genotypic data contains genotypes of 369 maize inbred lines consisting of $L = 1,406,757$ SNPs distributed across ten maize chromosomes. Marker effects were estimated on the basis of 369 shoot apical meristem phenotypes (Leiboff et al., 2015) using the BayesB model (Meuwissen et al., 2001). Similar to Goiffon et al. (2017), we assumed that marker effects were known and that errors in marker effects have an equal effect on all selection methods. The genetic map developed from maize nested association mapping (NAM) population is used for estimating recombination rates (Yu et al., 2008). To facilitate comparisons, genetic data were scaled such that the maximum potential of the initial breeding population is 100.

The same simulation process (shown in Figure 2.2) as (Goiffon et al., 2017) was used to compare the four methods in our study. Each of the components in Figure 2.2 is explained as follows:

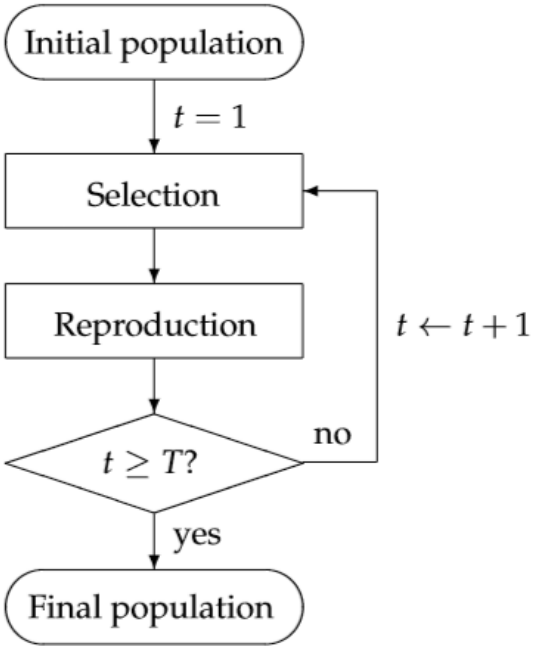

Figure 2.2 The simulation diagram, adopted from Goiffon et al. (2017).

- **The *initial population* start point:** In plant breeding, the genomic selection process starts with an initial population of individuals. The genotypes and marker effects are given at this point. In each simulation run, 200 individuals were selected randomly from the 369 maize inbred lines to make the initial population. Furthermore, the same set of 200 individuals were used as the initial population for all methods to make comparisons consistent.

- **The *selection* step:** All four methods were used to make selection decisions in this step, including mating strategies, number of crosses to make (nc) and number of progenies per cross (np).

  For **CGS**: $S = 20$ individuals with the highest GEBVs were selected and randomly mated to make $nc = 10$ crosses, each producing $np = 20$ progenies, maintaining a constant population size of 200.

  For **OHV**: $S = 20$ individuals with the highest OHVs were selected and randomly mated to make $nc = 10$ crosses, each producing $np = 20$ progenies, maintaining a constant population

size of 200. The same values of $B = 12$ and $F = 70\%$ as Goiffon et al. (2017) were used in our simulation where F is the percentage of individuals with the lowest GEBVs removed before optimizing the selected population.

For **OPV**: $S = 20$ individuals with the highest OPVs were selected and randomly mated to make $nc = 10$ crosses, each producing $np = 20$ progenies, maintaining a constant population size of 200. The same values of $B = 1$ and $F = 40\%$ as Goiffon et al. (2017) were used in our simulation.

For **LAS**: $S = 20$ individuals were selected and mated according to the look-ahead algorithm to make $nc = 10$ crosses. The number of progenies for each cross was determined by the heuristic strategy described in Section 2.3.3 with the constraint that the total number of progenies remains 200.

- **The *reproduction* step:** The selected individuals were crossed to make the breeding population for the next generation. A random progeny inherits the genetic information from its breeding parents according to *inheritance distribution* defined in Han et al. (2017). Let $P \in \{0,1\}^{L\times 2}$ denote the genotype of a random progeny produced from crossing individuals n_1 and n_2. Then P is determined as follow:

$$P_{i,j} = G_{i,J_i^j+1,n_j}, \forall i \in \{1,...,L\}, j \in \{1,2\},$$

where

$$J_1 = \begin{cases} 0, & \text{w.p. } 0.5 \\ 1, & \text{w.p. } 0.5 \end{cases}, \tag{2.14}$$

$$J_i = \begin{cases} J_{i-1} & \text{w.p. } 1-r_{i-1} \\ 1-J_{i-1} & \text{w.p. } r_{i-1} \end{cases}, \forall i \in \{2,...,L\}. \tag{2.15}$$

Here, "w.p." stands for "with probability".

- **The $t >= T$? condition:** The breeding cycle repeats itself until generation T, a predetermined deadline.
- **The *final population* end point:** After the terminal generation, the population will be assessed to determine its genetic improvement over the initial population.

2.4.2 Simulation Results

One thousand independent simulation repetitions were performed for each of the four selection approaches. Simulations were conducted on a computer with 256GB RAM and a processor with the following specifications: Intel(R) Xeon(R) CPU E5-4650 0 @2.70GHz 2.70GHz (2 processors). The computation time required for one simulation (including 4 methods) was 6248 seconds. Hence, it takes almost 1735 Hours (72 days) to conduct 1000 simulations. Ten different simulations have ran in parallel to reduce the CPU calender time to 7 days. The LAS method is modestly more computationally intensive. LAS requires approximately two times more computational time than the other three methods. Major results are summarized as follows.

2.4.2.1 Genetic gains over ten generations

Figure 2.3 shows the average cumulative genetic gains over ten generations. We define the cumulative genetic gain as the difference between the mean GEBV of the current population and that of the initial population. Because this figure shows genetic gains for each of the four methods averaged across 1,000 simulation repetitions, the comparison reflects their different performances in general. CGS achieved a high rate of genetic gain in the first three generations before gradually reaching a plateau. OHV maintained a relatively high rate of genetic gain throughout ten generations due to its emphasis on the progenies rather than the parents. OPV managed to achieve an even higher genetic gain by the terminal generation at the cost of lower rate of genetic gains in early generations, which is attribute to its group-based selection strategy that aims to achieve long-term genetic gains by combining desirable alleles from multiple breeding parents. LAS demonstrated a deadline-conscious strategy that patiently stays as an underdog in early generations while accumu-

lating desirable alleles but ultimately surpasses all other methods in the final generation. These results suggested that LAS is capable of making a trade-off between achieving short-term genetic gain and preserving long-term growth potential.

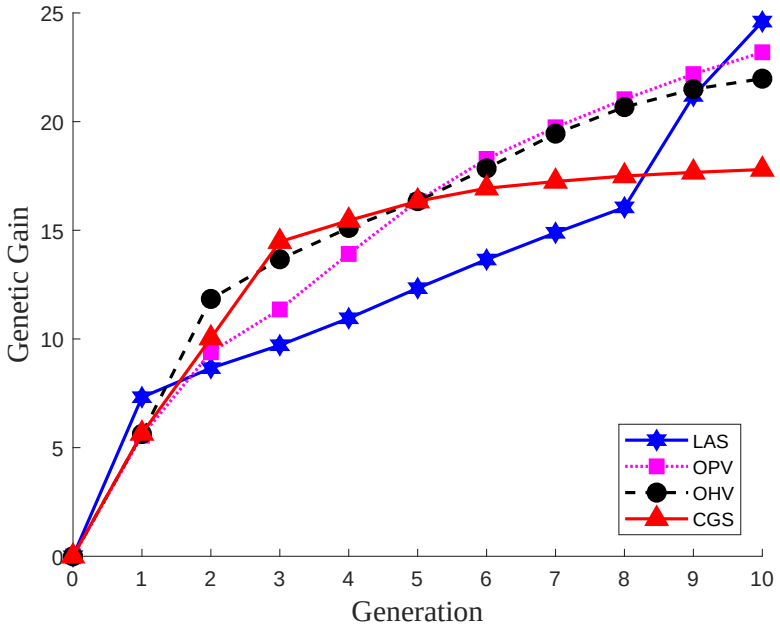

Figure 2.3 Cumulative genetic gains over 10 generations for four GS methods.

2.4.2.2 Genetic diversity over ten generations

Figure 2.4 displays the average genetic diversity (defined in equation (2.13)) over ten generations. The genetic diversity of the two truncation selection methods, CGS and OHV, dropped to about 35% of its initial value in the first two generations, which further deteriorated to about 15% in generation ten. In contrast, the two group-based selection methods, OPV and LAS, maintained genetic diversity at about 65% and 40% in generations two and ten, respectively. These results demonstrated the advantages of group-based selection methods over truncation-based methods in terms of preserving long-term genetic diversity.

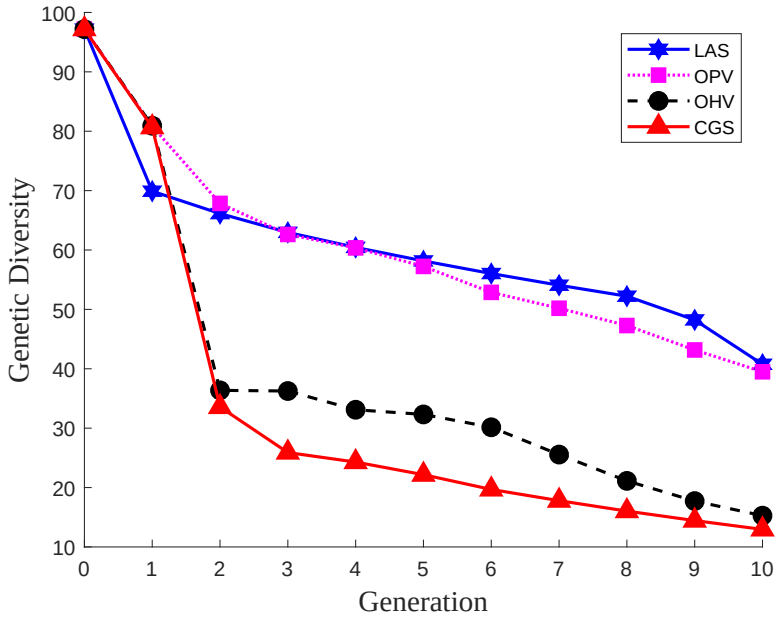

Figure 2.4 Genetic diversity over 10 generations for four GS methods.

2.4.2.3 Genetic gains with varying deadlines

LAS is the only method that adjusts selection decisions based on the user-defined deadline. Figure 2.5 shows the performance of LAS with varying deadlines from $T = 1$ to $T = 10$. In all ten cases, LAS used a similar strategy to patiently accumulate desirable alleles in early generations and make big leaps in the final two generations. As a result, LAS outperformed all other methods for all tested deadlines. The other three methods make the same selection decisions and thus result in the same performance under different deadlines.

2.4.2.4 Variable performance across different simulation repetitions

The average values and standard deviations (among the 1,000 simulation repetitions) for population minimum, mean, and maximum in the 10th generation are summarized in Table 2.1.

Figure 2.6 compares the cumulative distribution functions (CDFs) of the population maximum in generation 10. Here, the horizontal axis shows the GEBV of an individual (representing genetic gains) whereas the vertical axis is the percentile of the simulation repetitions. By definition,

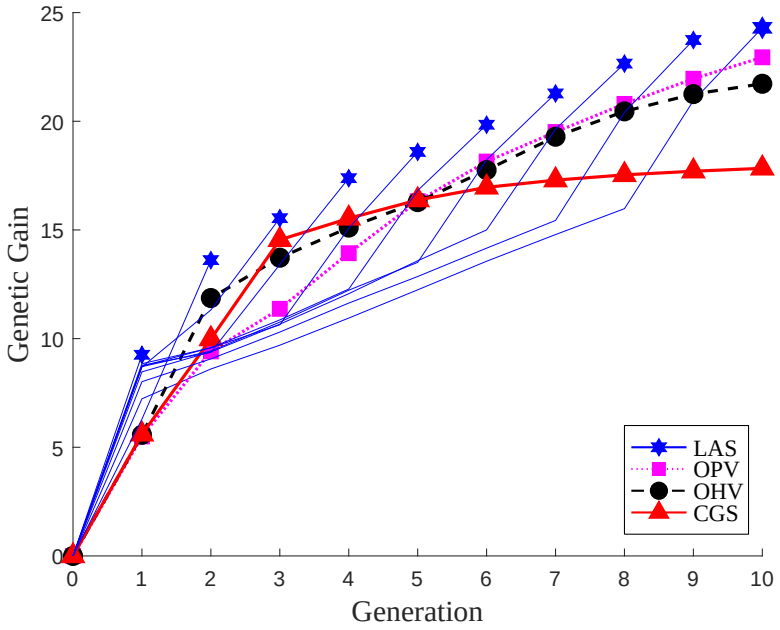

Figure 2.5 Genetic gains with varying deadlines from $T = 1$ to $T = 10$. LAS adjusts selection decisions based on the user-defined deadline whereas other three methods always make the same selection decisions.

the 1st percentile is one of the worst performances within the 1,000 simulation repetitions, the 99th percentile is one of the best, and the 50th percentile is the median value. As such, the further towards the right and bottom directions of the figure a CDF curves, the better performance a method has. The figure shows the improvements of different methods from CGS to LAS. In particular, LAS-X is a reduced version of LAS using the same resource allocation strategy with all previous methods (producing the same number of progenies from each cross), rather than using the

Table 2.1 Average values and standard deviations (among the 1,000 simulation repetitions) for population minimum, mean, and maximum in the 10th generation for four selection methods.

Method	Min	Mean	Max
CGS	54.88 ± 3.20	55.06 ± 3.23	55.24 ± 3.26
OHV	58.31 ± 4.27	58.95 ± 3.87	59.48 ± 3.84
OPV	57.56 ± 3.73	60.17 ± 3.97	62.16 ± 4.68
LAS	56.58 ± 3.97	61.53 ± 3.83	64.69 ± 4.25

heuristic strategy for resource allocation described in Section 2.3.3. These results demonstrated the effectiveness of LAS in making selection, mating, and resource allocation decisions.

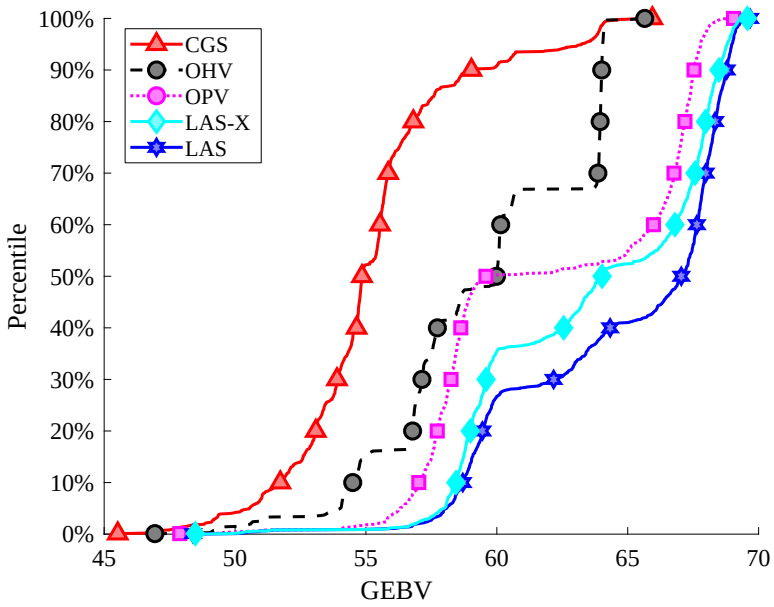

Figure 2.6 CDFs of population maximum, here LAS-X is the modified LAS method without resource allocation.

2.4.2.5 Behavior of LAS in the final two generations

LAS has an interesting behavior in the final two generations when it makes big leaps in genetic gain (Figures 2.3 and 2.5). This happens because LAS accumulates desirable alleles in the early generations to utilize in the final generations.

Figure 2.7 presents histograms of population GEBVs over time for one sample simulation using the LAS method. The yellow triangles show the GEBV of selected breeding parents from the population in each generation. This demonstrates how the breeding value rankings of the individuals selected by LAS change by generation. Note that in the last two generations LAS selects individuals with high GEBVs. This explains the behavior of LAS in the final two generations.

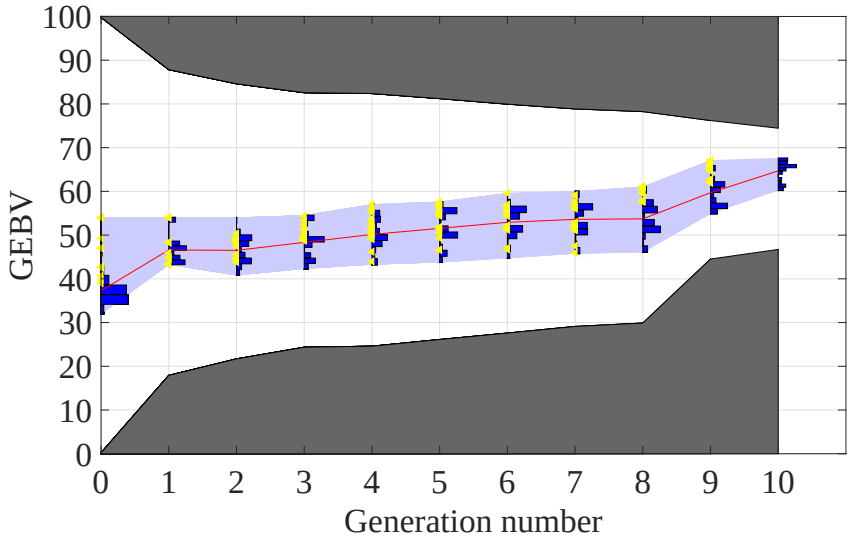

Figure 2.7 A sample simulation result using the LAS method presenting histograms of population GEBVs over time. Here, the red curve is the mean of population GEBVs and the boundaries of white and gray areas are the upper and lower selection limits. For a given generation, the upper selection limit shows the maximum potential of population in terms of GEBV values and similarly the lower selection limit shows the minimum potential of the population. The maximum, mean and minimum GEBVs are respectively 67.64, 64.69, and 60.18 in the final generation.

2.5 Conclusion

Genomic selection has been instrumental in improving the efficiency of plant breeding. In this study, we introduced a new selection method, LAS, which has the potential to further improve the efficiency of breeding given limited resources and specific user-defined project duration.

Unlike previous methods which try to maximize the genetic achievement of breeding parents or the best possible progeny without considering time and resource constraints, LAS is maximizing what exactly matters in a GS problem by aiming at the right objective. The objective of LAS is to maximize the expected GEBV of the best offspring in the terminal generation given a limited amount of resources. As such, this method is much more computationally challenging than previous ones, due to multiple complex factors such as recombination frequencies, mating strategy,

time management, and resource allocation that are explicitly accounted for. To deal with these challenges, we designed a simulation optimization algorithm that estimates and maximizes the LAS objective function by exploring the selection and mating solution space efficiently.

LAS makes three major contributions to the literature on genomic selection. First, LAS is deadline sensitive. Selection decisions adjust to the project duration to make a trade-off between achieving short-term genetic gains and maintaining genetic diversity long-term. Second, LAS optimizes both selection and mating strategies. It recognizes the importance of mating strategies and assigns selected individuals into pairs of breeding parents to achieve further genetic gains. Third, LAS involves resource allocation decisions. Rather than producing the same number of progenies from each cross, it allows breeding parents with higher genetic diversity to produce more progenies to increase the chance of producing high performers.

LAS was compared with previous genomic selection methods in a comprehensive simulation study using empirical data from a population of inbred maize lines. Computational results demonstrated the improvements of LAS over other methods in three perspectives: (1) LAS achieved the highest genetic gain by the deadline of the breeding project, which varied from one generation to ten generations. (2) LAS preserved the highest level of genetic diversity at the end of the breeding project. (3) LAS outperformed all other methods in almost all percentiles in the 1,000 simulation repetitions.

Future research is needed to address the limitations of the LAS method. The first assumption described in Section 2.3.1 is allowing only one progeny to be produced from the selected pairs of breeding parents in generation $t + 1$ and the second assumption is allowing the crosses to be made within the same generation each producing one progeny from generation $t + 1$ to $T - 1$. These two assumptions were made to simplify the computational requirement of estimating the objective function, which inevitably reduced its accuracy. Moreover, future studies can explore more comprehensive comparisons by performing simulations by: 1. using other methods for estimating marker effects such as GBLUP and ridge regression; 2. considering populations with different LD structures; and 3. applying different resource allocation strategies.

2.6 Acknowledgements

This work is supported by Agriculture and Food Research Initiative Grant no. 2017-67007-26175/Accession No. 1011702 from the USDA National Institute of Food and Agriculture. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the view of the U.S. Department of Agriculture. This work is also supported by the Plant Sciences Institute's Faculty Scholars program at Iowa State University.

2.7 Appendix: Proof for Theorem 1

This appendix proves theorem 1 through an example to provide a more insightful description for four different possibilities of recombination. Let's assume we start with three pairs of breeding parents (S=6). We represent the genotypic information of these individuals with the following matrices:

$$\text{pair one: } \begin{bmatrix} G_{1,1,1} & G_{1,2,1} \\ G_{2,1,1} & G_{2,2,1} \\ \vdots & \vdots \\ G_{L,1,1} & G_{L,2,1} \end{bmatrix} \times \begin{bmatrix} G_{1,1,2} & G_{1,2,2} \\ G_{2,1,2} & G_{2,2,2} \\ \vdots & \vdots \\ G_{L,1,2} & G_{L,2,2} \end{bmatrix}$$

$$\text{pair two: } \begin{bmatrix} G_{1,1,3} & G_{1,2,3} \\ G_{2,1,3} & G_{2,2,3} \\ \vdots & \vdots \\ G_{L,1,3} & G_{L,2,3} \end{bmatrix} \times \begin{bmatrix} G_{1,1,4} & G_{1,2,4} \\ G_{2,1,4} & G_{2,2,4} \\ \vdots & \vdots \\ G_{L,1,4} & G_{L,2,4} \end{bmatrix}$$

$$\text{pair three: } \begin{bmatrix} G_{1,1,5} & G_{1,2,5} \\ G_{2,1,5} & G_{2,2,5} \\ \vdots & \vdots \\ G_{L,1,5} & G_{L,2,5} \end{bmatrix} \times \begin{bmatrix} G_{1,1,6} & G_{1,2,6} \\ G_{2,1,6} & G_{2,2,6} \\ \vdots & \vdots \\ G_{L,1,6} & G_{L,2,6} \end{bmatrix}$$

The individuals in each pair are crossed to produce one progeny. The resulting progenies are then randomly mated for $T-t-1$ generations. $g \in \{0,1\}^L$ is the random gamete produced by breeding parents in meiosis of the $(T-1)$st generation. From equation (2.12) we see that four possibilities exist for recombination. Here, we illustrate those four cases with color coding. We divide the process into two phases: *Phase 1*: generation 0 until 2 and *Phase 2*: generation 2 until T. Let $h \in \{0,1\}^L$ denote the genotype of a specific gamete produced in meiosis by a progeny of a specific pair of breeding parents from the breeding population. This specific gamete contains the allele G_{l,m^0,i^0} that is passed on to the gamete g at locus l, i.e., $h_l = g_l = G_{l,m^0,i^0}$. We know that such a gamete uniquely exists because of the way the two phases are defined. The four cases are as follow:

Case 1: No recombination happens (g_2 comes from the same chromosome as g_1).

$$\begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_L \end{bmatrix} = \begin{bmatrix} G_{1,1,1} \\ G_{2,1,1} \\ \vdots \\ G_{L,m,s} \end{bmatrix}$$

According to equation (2.12), when $i^0 = i^1$ and $m^0 = m^1$, we have:

$$P(g_{l+1} = G_{l+1,m^1,i^1} | g_l = G_{l,m^0,i^0}) = (1-r_{l+1})^2(1-R_{l+1}) \tag{2.16}$$

$$\forall l \in \{1,...,L-1\},\ \forall i^0, i^1 \in \{1,2,...,S\},\ \forall m^0, m^1 \in \{1,2\}.$$

Using this definition equation (2.16) can be calculated as follow:

$$P(g_{l+1} = G_{l+1,m^1,i^1} | g_l = G_{l,m^0,i^0}) \tag{2.17}$$

$$= P(g_{l+1} = G_{l+1,m^0,i^0} | g_l = G_{l,m^0,i^0}) \tag{2.18}$$

$$= P(h_{l+1} = G_{l+1,m^0,i^0}, g_{l+1} = h_{l+1} | h_l = G_{l,m^0,i^0}, g_l = h_l) \tag{2.19}$$

$$= P(h_{l+1} = G_{l+1,m^0,i^0} | h_l = G_{l,m^0,i^0}, g_l = h_l) \tag{2.20}$$

$$\cdot P(g_{l+1} = h_{l+1} | h_l = G_{l,m^0,i^0}, g_l = h_l)$$

$$= P(h_{l+1} = G_{l+1,m^0,i^0} | h_l = G_{l,m^0,i^0}) P(g_{l+1} = h_{l+1} | g_l = h_l) \tag{2.21}$$

$$= (1-r_{l+1})^2(1-R_{l+1}) \tag{2.22}$$

Equation (2.18) comes from the fact that $i^0 = i^1$ and $m^0 = m^1$. Equation (3.8) is derived from equation (2.18) because of the way h is defined. To find equation (2.20) from (3.8) independency is used. Finally, equation (2.21) is derived from (2.20) due to the fact that $h_{l+1} = G_{l+1,m^0,i^0}$ is independent from $g_l = h_l$ and also $g_{l+1} = h_{l+1}$ is independent from $h_l = G_{l,m^0,i^0}$.

Here, R_l is the recombination frequency between allele l and allele $l+1$, $\forall l \in \{1, ..., L-1\}$ after $(T-t)-2$ number of generations and is calculated as:

$$R_l = 1 - P(g_{l+1} = h_{l+1} | g_l = h_l) \tag{2.23}$$

$$R_l^2 = 0$$
$$R_l^i = 1 - \left((1 - R_l^{i-1})(1 - r_l) + \frac{r_l}{S/2} \right) \quad \forall i \in \{3, 4, ..., \tau\}$$

Where r_l is the l^{th} recombination frequency for $l \in \{1, 2, ..., L-1\}$ and S is number of breeding parents. From the above equations we obtain:

$$R_l = \frac{(S-2)\left(1 - (1-r_l)^{T-t}\right)}{S} \tag{2.24}$$

This provides the proof for equation (2.12).

Case 2: Recombination happens within an individual (g_2 is coming from the other chromosome of the same individual where g_1 is coming from).

$$\begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_L \end{bmatrix} = \begin{bmatrix} G_{1,1,1} \\ G_{2,2,1} \\ \vdots \\ G_{L,m,s} \end{bmatrix}$$

According to equation (2.12), when $i^0 = i^1$ and $m^0 \neq m^1$, we have:

$$P(g_{l+1} = G_{l+1,m^1,i^1} | g_l = G_{l,m^0,i^0}) = r_{l+1}(1 - r_{l+1})(1 - R_{l+1}) \tag{2.25}$$

$$\forall l \in \{1, ..., L-1\},\ \forall i^0, i^1 \in \{1, 2, ..., S\},\ \forall m^0, m^1 \in \{1, 2\}.$$

Similarly, equation (2.25) can be calculated as follow:

$$P(g_{l+1} = G_{l+1,m^1,i^1} | g_l = G_{l,m^0,i^0}) \tag{2.26}$$

$$= P(h_{l+1} = G_{l+1,m^1,i^0}, g_{l+1} = h_{l+1} | h_l = G_{l,m^0,i^0}, g_l = h_l) \tag{2.27}$$

$$= P(h_{l+1} = G_{l+1,m^1,i^0} | h_l = G_{l,m^0,i^0}) P(g_{l+1} = h_{l+1} | g_l = h_l) \tag{2.28}$$

$$= r_{l+1}(1 - r_{l+1})(1 - R_{l+1}) \tag{2.29}$$

Case 3: Recombination happens within the paired individual.

$$\begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_L \end{bmatrix} = \begin{bmatrix} G_{1,1,1} \\ G_{2,1,2} \\ \vdots \\ G_{L,m,s} \end{bmatrix}, \text{ or } \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_L \end{bmatrix} = \begin{bmatrix} G_{1,1,1} \\ G_{2,2,2} \\ \vdots \\ G_{L,m,s} \end{bmatrix}$$

According to equation (2.12), when $\left\lceil \frac{i^0}{2} \right\rceil = \left\lceil \frac{i^1}{2} \right\rceil$, we have:

$$P(g_{l+1} = G_{l+1,m^1,i^1} | g_l = G_{l,m^0,i^0}) = \frac{1}{2} r_{l+1}(1 - R_{l+1}) \tag{2.30}$$

$$\forall l \in \{1, ..., L-1\},\ \forall i^0, i^1 \in \{1, 2, ..., S\},\ \forall m^0, m^1 \in \{1, 2\}.$$

Similarly, equation (2.30) can be calculated as follow:

$$P(g_{l+1} = G_{l+1,m^1,i^1} | g_l = G_{l,m^0,i^0}) \tag{2.31}$$

$$= P(h_{l+1} = G_{l+1,m^1,i^1}, g_{l+1} = h_{l+1} | h_l = G_{l,m^0,i^0}, g_l = h_l) \tag{2.32}$$

$$= P(h_{l+1} = G_{l+1,m^1,i^1} | h_l = G_{l,m^0,i^0}) P(g_{l+1} = h_{l+1} | g_l = h_l) \tag{2.33}$$

$$= \frac{1}{2} r_{l+1}(1 - R_{l+1}) \tag{2.34}$$

Case 4: This case considers all possible remaining recombination.

$$\begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_L \end{bmatrix} = \begin{bmatrix} G_{1,1,1} \\ G_{2,1,3} \\ \vdots \\ G_{L,m,s} \end{bmatrix}, \text{ or } \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_L \end{bmatrix} = \begin{bmatrix} G_{1,1,1} \\ G_{2,2,3} \\ \vdots \\ G_{L,m,s} \end{bmatrix}, \text{ or}$$

$$\begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_L \end{bmatrix} = \begin{bmatrix} G_{1,1,1} \\ G_{2,1,4} \\ \vdots \\ G_{L,m,s} \end{bmatrix}, \text{ or } \cdots, \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_L \end{bmatrix} = \begin{bmatrix} G_{1,1,1} \\ G_{2,2,6} \\ \vdots \\ G_{L,m,s} \end{bmatrix}$$

According to equation (2.12), when $\left\lceil \frac{i^0}{2} \right\rceil \neq \left\lceil \frac{i^1}{2} \right\rceil$, we have:

$$P(g_{l+1} = G_{l+1,m^1,i^1} | g_l = G_{l,m^0,i^0}) = \frac{R_{l+1}}{2(S-2)} \tag{2.35}$$

$$\forall l \in \{1, ..., L-1\},\ \forall i^0, i^1 \in \{1, 2, ..., S\},\ \forall m^0, m^1 \in \{1, 2\}.$$

Similarly, equation (2.35) can be calculated as follow:

$$P(g_{l+1} = G_{l+1,m^1,i^1} | g_l = G_{l,m^0,i^0}) \tag{2.36}$$

$$= P(h_{l+1} = G_{l+1,m^1,i^1}, g_{l+1} = h_{l+1} | h_l = G_{l,m^0,i^0}, g_l = h_l) \tag{2.37}$$

$$= P(h_{l+1} = G_{l+1,m^1,i^1} | h_l = G_{l,m^0,i^0}) P(g_{l+1} = h_{l+1} | g_l = h_l) \tag{2.38}$$

$$= \frac{1}{4} \times \frac{R_{l+1}}{\frac{S}{2} - 1} \tag{2.39}$$

$$= \frac{R_{l+1}}{2(S-2)} \tag{2.40}$$

2.8 References

Akdemir, D., Beavis, W., Fritsche-Neto, R., Singh, A. K., and Isidro-Sánchez, J. (2018). Multi-objective optimized genomic breeding strategies for sustainable food improvement. *Heredity*, page 1.

Akdemir, D. and Sánchez, J. I. (2016). Efficient breeding by genomic mating. *Frontiers in Genetics*, 7:210.

Daetwyler, H. D., Hayden, M. J., Spangenberg, G. C., and Hayes, B. J. (2015). Selection on optimal haploid value increases genetic gain and preserves more genetic diversity relative to genomic selection. *Genetics*, 200(4):1341–1348.

Goddard, M. (2009). Genomic selection: prediction of accuracy and maximisation of long term response. *Genetica*, 136(2):245–257.

Goiffon, M., Kusmec, A., Wang, L., Hu, G., and Schnable, P. (2017). Optimal population value selection: A population-based selection strategy for improving response in genomic selection. *Genetics*, pages genetics–116.

Gorjanc, G., Gaynor, R. C., and Hickey, J. M. (2018). Optimal cross selection for long-term genetic gain in two-part programs with rapid recurrent genomic selection. *Theoretical and Applied Genetics*, 131(9):1953–1966.

Hallatschek, O. and Geyrhofer, L. (2015). Collective fluctuations in models of adaptation. *arXiv preprint arXiv:1506.08683*.

Han, Y., Cameron, J. N., Wang, L., and Beavis, W. D. (2017). The predicted cross value for genetic introgression of multiple alleles. *Genetics*, 205(4):1409–1423.

Hayes, B., Shepherd, R., and Newman, S. (2002). Look ahead mate selection schemes for multi-breed beef populations. *Animal Science*, 74(1):13–23.

Hayes, B. J., Bowman, P. J., Chamberlain, A., and Goddard, M. (2009). Invited review: Genomic selection in dairy cattle: Progress and challenges. *Journal of Dairy Science*, 92(2):433–443.

Jannink, J.-L. (2010). Dynamics of long-term genomic selection. *Genetics Selection Evolution*, 42(1):35.

Jannink, J.-L., Lorenz, A. J., and Iwata, H. (2010). Genomic selection in plant breeding: from theory to practice. *Briefings in Functional Genomics*, 9(2):166–177.

Kinghorn, B. P. (2011). An algorithm for efficient constrained mate selection. *Genetics Selection Evolution*, 43(1):4.

Leiboff, S., Li, X., Hu, H.-C., Todt, N., Yang, J., Li, X., Yu, X., Muehlbauer, G. J., Timmermans, M. C., Yu, J., et al. (2015). Genetic control of morphometric diversity in the maize shoot apical meristem. *Nature Communications*, 6:8974.

Lin, Z., Shi, F., Hayes, B. J., and Daetwyler, H. D. (2017). Mitigation of inbreeding while preserving genetic gain in genomic breeding programs for outbred plants. *Theoretical and Applied Genetics*, 130(5):969–980.

Liu, H., Henryon, M., and Sørensen, A. (2017). Mating strategies with genomic information reduce rates of inbreeding in animal breeding schemes without compromising genetic gain. *Animal*, 11(4):547–555.

Lorenz, A. J. (2013). Resource allocation for maximizing prediction accuracy and genetic gain of genomic selection in plant breeding: a simulation experiment. *G3: Genes, Genomes, Genetics*, 3(3):481–491.

Lorenzana, R. E. and Bernardo, R. (2009). Accuracy of genotypic value predictions for marker-based selection in biparental plant populations. *Theoretical and Applied Genetics*, 120(1):151–161.

Meuwissen, T. (1997). Maximizing the response of selection with a predefined rate of inbreeding. *Journal of Animal Science*, 75(4):934–940.

Meuwissen, T. H. E., Hayes, B. J., and Goddard, M. E. (2001). Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157(4):1819–1829.

Mujibi, F., Nkrumah, J., Durunna, O., Stothard, P., Mah, J., Wang, Z., Basarab, J., Plastow, G., Crews Jr, D., and Moore, S. (2011). Accuracy of genomic breeding values for residual feed intake in crossbred beef cattle. *Journal of Animal Science*, 89(11):3353–3361.

Nakaya, A. and Isobe, S. N. (2012). Will genomic selection be a practical method for plant breeding? *Annals of botany*, 110(6):1303–1316.

Rincent, R., Charcosset, A., and Moreau, L. (2017). Predicting genomic selection efficiency to optimize calibration set and to assess prediction accuracy in highly structured populations. *Theoretical and Applied Genetics*, 130(11):2231–2247.

Rosvall, O. (1999). *Enhancing gain from long-term forest tree breeding while conserving genetic diversity*. Swedish University of Agricultural Sciences Umeå, Sweden, Sylvestria.

Sonesson, A. K., Woolliams, J. A., and Meuwissen, T. H. (2012). Genomic selection requires genomic control of inbreeding. *Genetics Selection Evolution*, 44(1):27.

Sun, C., VanRaden, P., O'Connell, J., Weigel, K., and Gianola, D. (2013). Mating programs including genomic relationships and dominance effects. *Journal of Dairy Science*, 96(12):8014–8023.

Toro, M. A. and Varona, L. (2010). A note on mate allocation for dominance handling in genomic selection. *Genetics Selection Evolution*, 42(1):33.

Ullrich, S. E. (2007). Breeding field crops. *Crop Science*, 47(2):900.

VanRaden, P., Van Tassell, C., Wiggans, G., Sonstegard, T., Schnabel, R., Taylor, J., and Schenkel, F. (2009). Invited review: Reliability of genomic predictions for north american holstein bulls. *Journal of Dairy Science*, 92(1):16–24.

Wang, L., Zhu, G., Johnson, W., and Kher, M. (2018). Three new approaches to genomic selection. *Plant Breeding*, 137(5):673–681.

Yu, J., Holland, J. B., McMullen, M. D., and Buckler, E. S. (2008). Genetic design and statistical power of nested association mapping in maize. *Genetics*, 178(1):539–551.

CHAPTER 3. MULTI-TRAIT GENOMIC SELECTION METHODS FOR CROP IMPROVEMENT

Saba Moeinizade[1], Aaron Kusmec[2], Guiping Hu[1], Lizhi Wang[1], and Patrick S. Schnable[2]

[1]Department of Industrial and Manufacturing Systems Engineering, Iowa State University,

[2]Department of Agronomy, Iowa State University

Modified from a manuscript published in *Genetics*

3.1 Abstract

Plant breeders make selection decisions based on multiple traits, such as yield, plant height, flowering time, and disease resistance. A commonly used approach in multi-trait genomic selection is index selection, which assigns weights to different traits relative to their economic importance. However, classical index selection only optimizes genetic gain in the next generation, requires some experimentation to find weights that lead to desired outcomes, and has difficulty optimizing non-linear breeding objectives. Multi-objective optimization has also been used to identify the Pareto frontier of selection decisions, which represents different trade-offs across multiple traits. We propose a new approach, which maximizes certain traits while keeping others within desirable ranges. Optimal selection decisions are made using a new version of the look-ahead selection algorithm, which was recently proposed for single trait genomic selection and achieved superior performance with respect to other state-of-the-art selection methods. To demonstrate the effectiveness of the new method a case study is developed using a realistic data set where our method is compared with conventional index selection. Results suggest that the multi-trait look-ahead selection is more effective at balancing multiple traits compared to index selection.

3.2 Introduction

Genomic selection (GS), which was initially proposed by Meuwissen et al. (2001), is a special form of marker assisted selection (MAS) that simultaneously estimates the effects of genome-wide markers in a training population consisting of genotyped and phenotyped individuals. Selection decisions are based on genomic estimated breeding values (GEBVs) in a breeding population, which are calculated as the sum of the estimated marker effects. The advantages of GS have been demonstrated by simulation and empirical studies (Meuwissen et al., 2001; Makowsky et al., 2011; Schaeffer, 2006; Wang et al., 2018; Goddard, 2009).

Previous studies have mainly focused on the development of models to improve the accuracy of GEBV prediction. Until recently, few studies have considered alternatives to truncation selection on GEBVs followed by random mating of the selected individuals. These studies have focused on selecting the parents of the next generation by defining new quantitative selection metrics (Goddard, 2009; Daetwyler et al., 2015; Goiffon et al., 2017; Moeinizade et al., 2020) or jointly considering selection and mating decisions (Moeinizade et al., 2019; Akdemir and Sánchez, 2016). The latter two methods are forms of mate selection (Kinghorn and R.K.Shepherd, 1999) that optimize the contributions of potential parents to the next generation based on maximizing a desired breeding objective. Typically, the optimization is performed with respect to the next generation (Kinghorn and Kinghorn, 2016; Akdemir and Sánchez, 2016). Look-ahead mate selection (LAMS) schemes that optimize parental contributions with respect to grand-progeny (i.e., two generations in the future) have also been proposed in the context of animal breeding (Hayes et al., 1998; Shepherd and Kinghorn, 1998; Hayes et al., 2002).

Moeinizade et al. (2019) implemented a LAMS scheme –look-ahead selection (LAS)– in a stochastic simulation framework that seeks to optimize the performance of the best possible progeny in an arbitrarily defined terminal generation. This strategy was shown to outperform conventional genomic selection (Meuwissen et al., 2016), optimal haploid value selection (Daetwyler et al., 2015), and optimal population value selection (Goiffon et al., 2017) using empirical data from a population

of maize inbred lines. LAS outperformed previous approaches by achieving more genetic gain and preserving more genetic diversity over the course of a simulated breeding program.

Although LAS presents a significant improvement over competing methods, it is confined to single trait genomic selection (ST-GS). Generally, the productivity of a crop variety is dependent on multiple characteristics such as yield, grain quality, and disease resistance. Hence, selection and mating decisions should be based on several different characteristics with potentially different breeding goals. Multi-trait selection poses difficulties for breeders because it often requires balancing competing breeding objectives. Four principle multi-trait genomic selection (MT-GS) strategies have been proposed in the literature: 1) tandem selection, whereby different traits are selected independently in different generations (Burgess and West, 1993); 2) independent culling, whereby truncation selection is performed on multiple traits simultaneously with independent thresholds (Hazel, 1943); 3) index selection, whereby multiple traits are selected at the same time by constructing an index that is a linear combination of multiple traits (Hazel and Lush, 1942; Hazel, 1943); and 4) mate selection, whereby multiple traits are selected at the same time by finding Pareto optimal solutions of a mate selection index (Kinghorn and Kinghorn, 2016).

Tandem selection, by definition, is not capable of selecting multiple traits simultaneously and is most useful when some traits should be selected in earlier generations than others. Independent culling does perform simultaneous selection but is very sensitive to the truncation points for the different traits. Index selection has become an important method that has been widely used for the development of superior varieties in both animal and plant breeding (Villanueva and Woolliams, 1997; Jannink et al., 2000; Ivkovich and Koshy, 2002; Sharma and Duveiller, 2003; Long et al., 2006; Yan and Frégeau-Reid, 2008). This often takes the form of truncation selection on an index constructed by integrating information on the economic values of the different traits and their phenotypic and additive genetic covariances. Brascamp (1984) provides a concise summary of different selection indices. Mate selection can consider different constraints and breeding goals for multiple traits and evaluates these criteria in the context of a proposed set of matings. Two recent studies in plants have evaluated the use of mate selection on long-term genetic gains (Cowling et al.,

2019; Suontama et al., 2018). Additionally, Akdemir and Sánchez (2016) and Akdemir et al. (2019) have developed new mate selection methods for single- and multi-trait scenarios, respectively, with an emphasis on application to plant breeding.

An additional challenge in multi-trait selection is the definition of breeding objectives for each trait. For example, a breeder wishing to maximize grain yield might also need to maintain minimum standards for standability and disease resistance and an acceptable range of plant heights. Kempthorne and Nordskog (1959) proposed maintaining a trait at an optimal level by weighting its squared deviations from the optimum. Wilton et al. (1968) generalized this approach to include both squares and cross products of multiple traits. Moav and Hill (1966) developed a graphical method to calculate explicitly non-linear indices on two traits. Later, iterative solutions were developed to identify the optimal weights for a non-linear index on an arbitrary number of traits (Itoh and Yamada, 1988; Pasternak and Weller, 1993). However, the general solution for the weights of a non-linear index is dependent on the population mean prior to selection and the intensity of selection (Weller et al., 1996). Therefore, the optimum selection index changes each generation and will be different from an index that maximizes gains over multiple generations.

In this paper, we propose an extension of the single-trait LAS method to multiple traits with different breeding objectives. The method maximizes a single, main trait while constraining other traits to fall within flexibly defined ranges. It retains the advantages of single-trait LAS derived from considering the impacts of selection, mating, and resource allocation decisions on the performance of individuals in the terminal generation of the breeding program.

3.3 Materials and Methods

3.3.1 Data-sets

A dataset of 5,022 maize recombinant inbred lines (RILs) from the US nested association mapping (US-NAM) (Yu et al., 2008) and intermated B73xMo17 (IBM) (Lee et al., 2002) populations was used in this study. Best linear unbiased predictors (BLUPs) for total kernel weight were taken from Yang et al. (2018). BLUPs for ear height were calculated from the phenotypic data in Kusmec

et al. (2017) using a mixed model with genotype and environment as random effects. The mixed model was implemented in the R package lme4 (Bates et al., 2015).

SNPs from Kusmec et al. (2017) were thinned using PLINK v1.90b (Chang et al., 2015) using the "indep-pairwise" function with a window size of 250 kb, a step size of 50 SNPs, and an LD threshold of 0.6. Thinned SNPs were imputed and phased with Beagle v4.0 (Browning and Browning 2008) using default parameters. This produced 359,826 imputed and phased SNPs. SNP effects for each phenotype were estimated using the BayesB algorithm (Meuwissen et al., 2001) implemented in GenSel4 (Fernando and Garrick, 2009). Recombination rates were estimated using the genetic map for the US-NAM population (Yu et al., 2008) following the procedure outlined in Goiffon et al. (2017).

3.3.2 Simulation Design

One hundred independent simulations of a ten-generation breeding program were performed using a maize data set. An initial population of 200 individuals was randomly selected from the full data set, and in each generation, 20 individuals were selected to make 10 crosses. More details on the simulation steps are available in Goiffon et al. (2017) and Moeinizade et al. (2019).

3.3.3 Single-trait Look-ahead Selection

In this section, we review the look-ahead selection method which was recently proposed for single-trait genomic selection (Moeinizade et al., 2019). To make this algorithm more robust, we present an equivalent reformulation of this method and then discuss how this algorithm can be extended for multiple trait settings in the next section.

The single-trait look-ahead selection (ST-LAS) method anticipates the consequences of selection and mating decisions over several generations via simulation by quantitatively taking into account recombination frequencies during meiosis. The ST-LAS method has three major contributions to the literature: 1) time management: ST-LAS is the only GS method that takes time constraints into account and is deadline sensitive; 2) mating strategy optimization: the ST-LAS method not

only makes the selection decisions but also specifies how to pair the selected individuals for mating; and 3) resource allocation: this method uses a heuristic strategy to allocate more progeny to crosses between more diverse parents to increase the probability of producing high performing individuals.

The cornerstone of this method is evaluating a given selection and mating strategy by estimating the distribution of progeny GEBVs in the final generation. By simulating the GEBVs of a random sample of individuals in the final generation, a breeder can make better selection and mating decisions. This method can be formulated as the following optimization model (Moeinizade et al., 2019):

$$\max_{x,y} \quad f^{\text{LAS}}(x, y, r, \tau) \tag{3.1}$$

$$\text{s.t.} \quad \textstyle\sum_{n=1}^{N} x_n = S \tag{3.2}$$

$$x_n \in \{0, 1\} \qquad \forall n \in \{1, ..., N\} \tag{3.3}$$

$$x_n = \textstyle\sum_{j=1}^{N} y_{n,j} \quad \forall n \in \{1, ..., N\} \tag{3.4}$$

$$y_{i,j} \in \{0, 1\} \qquad \forall i, j \in \{1, ..., N\} \tag{3.5}$$

This optimization model has two decision variables: x, which represents the selection strategy, and y, which represents the mating strategy. Below is a detailed description of the objective as well as all variables and parameters used in this model:

- f^{LAS}: The expected GEBV of the best offspring in the terminal generation.
- x_n: A binary decision variable that shows whether individual n is selected ($x_n = 1$) or not ($x_n = 0$).
- $y_{i,j}$: A binary variable that shows whether individual i is mated with individual j ($y_{i,j} = 1$) or not ($y_{i,j} = 0$).
- $r \in [0, 0.5]^{L-1}$: The recombination frequency vector.
- τ: The remaining number of generations ($\tau = T - t$ where t is the current generation and T is the deadline generation number).

- N: The number of individuals in the population.
- S: The number of individuals that are to be selected out of the current population.

As demonstrated in equation (3.1), the objective of the ST-LAS method is dependent on selection (x), mating (y), recombination frequencies (r), and remaining number of generations (τ). Constraint (3.2) states that S individuals are selected from total N individuals in the population to make $S/2$ crosses (assuming that S is an even number) and constraint (3.3) ensures that the decision variable x, is binary. Constraint (3.4) ensures that each selected individual is mated once. Finally, constraint (3.5) states that the decision variable y is binary.

In this model, evaluation of $f^{\text{LAS}}(x, y, r, T-t)$ is very challenging because of the uncertainty involved due to recombination frequencies (r) and also selection (x) and mating (y) decisions over $T-t$ generations. To deal with these challenges, a simulation optimization algorithm was designed that estimates and maximizes the LAS objective function by exploring the selection and mating solution space efficiently.

3.3.4 Equivalent Formulation of Single-trait Look-ahead Selection

According to Moeinizade et al. (2019), the objective of ST-LAS is to maximize the expected GEBV of the best offspring in the terminal generation (equation (3.1)). The *best* offspring can be the individual with maximum expected GEBV in the final generation; however, the maximum value does not necessarily represent the whole distribution. To make the prediction more robust and reduce the influence of outliers, we present an equivalent reformulation of the ST-LAS method (equations (3.6)-(3.8)) where the *best* offspring is defined as *the* $100\gamma^{th}$ *percentile* among predicted GEBVs of individuals in the terminal generation.

$$\max_{x,y} \quad \phi \tag{3.6}$$

$$\text{s.t.} \quad \text{Constraints (3.2), (3.3), (3.4), and (3.5)} \tag{3.7}$$

$$\Pr[g_1(x, y, r, \tau) \geq \phi] \geq 1 - \gamma \tag{3.8}$$

Here, ϕ is a threshold value, equivalent to the previous objective f^{LAS}, which represents the expected GEBV of the best offspring in the final generation where best is defined as the $100\gamma^{\text{th}}$ percentile of the simulated GEBV distribution. The new variables and parameters are defined as follow:

- ϕ: The expected GEBV of the best offspring in the terminal generation.
- $g_1(x, y, r, \tau)$: The expected GEBV of a random progeny in the terminal generation (for trait 1 which is the only trait in the case of ST-LAS).
- γ: A parameter that defines which percentile of the GEBV distribution is evaluated in the final generation.

In this model, constraint (3.8) states that for a random progeny in the final generation, the probability of having an expected GEBV at least equal to the threshold value is greater than or equal to $1-\gamma$. For example, for a random sample of 1000 progeny, if $\gamma = 0.98$, then ϕ will evaluate the GEBV of the top 2% of progeny.

3.3.5 Multi-trait Look-ahead Selection

In this section, we present a new approach for MT-GS problems to optimize the main goal of a breeding program while keeping other traits within desired ranges. This new approach, multi-trait look-ahead selection (MT-LAS), extends the ST-LAS method to multiple trait settings. It should be noted that the same resource allocation heuristic from Moeinizade et al. (2019) is applied to MT-LAS. This resource allocation strategy aims to preserve more genetic diversity by varying the number of progeny produced from each cross relative to their breeding parents genetic diversity.

Assume there exists J different traits of which one, $j = 1$ (e.g., yield), should be maximized while the other traits, $j \in \{2, 3, ..., J\}$ (e.g., plant height, ear height, etc.), should satisfy certain criteria. This problem can be formulated as an optimization model as follows:

$$\max_{x,y} \quad \phi \tag{3.9}$$

$$\text{s.t.} \quad \text{Constraints (3.2)-(3.5)} \tag{3.10}$$

$$\Pr[g_1(x,y,r,\tau) \geq \phi | l_j \leq g_j(x,y,r,\tau) \leq u_j, \tag{3.11}$$

$$\forall j \in \{2,3,...,J\}]$$

$$\geq 1-\gamma$$

This model shares the same objective and constraints (3.2), (3.3), (3.4), and (3.5) with the equivalent reformulation of ST-LAS. However, constraint (3.11) is a modification of constraint (3.8) which focuses on making sure that traits $j \in \{2,3,...,J\}$ fall into desired ranges by defining a conditional probability. Below is a detailed description of all new variables and parameters:

- $g_j(x,y,r,\tau)$: The expected GEBV of a random progeny in the terminal generation for trait j where $j \in \{2,3,...,J\}$.
- l_j: The lower value for trait j.
- u_j: The upper value for trait j.

This model aims to maximize the expected GEBV of the top $100(1-\gamma)\%$ of offspring in the terminal generation for the trait of interest (e.g., yield) among offspring that also meet thresholds with respect to other traits (e.g., plant height, grain quality, etc.). Without loss of generality, $l_j = -\infty$ or $u_j = \infty$ capture the cases when only a lower bound or upper bound should be considered. Note that when only one trait ($j = 1$) is considered, this formulation is equivalent to ST-LAS.

The ST-LAS optimization model was already challenging to solve and after adding a nonlinear and non-convex constraint (3.11), the computational complexity increases significantly. To overcome this challenge, we redefine constraint (3.11) by converting the conditional probability on l and u to a penalty that dynamically adjusts the objective function in response to violations of the

boundaries. The penalty allows violations of the boundaries that are offset by improvements in the objective function. Take, for example, the case that the decision maker wants to maximize yield while making sure that plant height does not exceed a certain value. What if we could improve yield by slightly violating the height constraint? We want the height constraint to be true, but not at the expense of losing the main objective.

The following mathematical model formulates the problem:

$$\max_{x,y} \quad \phi \tag{3.12}$$

$$\text{s.t.} \quad \text{Constraints (3.2)-(3.5)} \tag{3.13}$$

$$\theta_j = \Pr[l_j \leq g_j(x,y,r,\tau) \leq u_j], \forall j \in \{2,...,J\} \tag{3.14}$$

$$\Delta = \max(g_j(x,y,r,\tau) - u_j, l_j - g_j(x,y,r,\tau), 0) \tag{3.15}$$

$$\Pr[h(x,y,r,\tau) \geq \phi] \geq 1-\gamma \tag{3.16}$$

$$h(x,y,r,\tau) = \theta_1 g_1(x,y,r,\tau) - \textstyle\sum_{j=2}^{J} \frac{(1-\theta_j)}{J-1}\Delta \tag{3.17}$$

Here, θ_j is the probability that a random progeny is acceptable in the final generation with respect to trait j for $j \in \{2,3,...,J\}$ and $\theta_1 = \frac{\sum_{j=2}^{J}\theta_j}{J-1}$. The new function $h(x,y,r,\tau)$ is a linear combination of the expected GEBV of a random progeny for trait $j=1$ and the penalty of violating the desired range for traits $j \in \{2,3,...,J\}$.

Here are some properties of constraints (3.14)-(3.17) :

- The term Δ in equation (3.15) represents the penalty for violating the upper or lower bounds for a random progeny in the terminal generation. As the magnitude of the violation increases, the penalty term increases. In the case of no violation, the penalty becomes 0.

- From equation (3.17), the term $\sum_{j=2}^{J} \frac{(1-\theta_j)}{J-1}\Delta$ is the weighted sum of penalties for all traits of $j \in \{2,3,...,J\}$. The weight $(1-\theta_j)$ is the probability that a random progeny violates the desired range.

Algorithm 1: Heuristic for optimizing the MT-LAS model

Select S random individuals from the population;
Randomly mate selected individuals;
Calculate ϕ ($Vmax \leftarrow \phi$);
Set $f \in \{1, 2, ..., S\}$ as list of positions to check;
Set $nf \in \{1, 2, ..., S\}$ as number of positions to check;
while $nf \geq 0$ **do**
    Generate $z \in [1, nf]$ as a random integer;
    $i \leftarrow$ the z^{th} value in f;
    $j \leftarrow$ index of the i^{th} individual;
    Swap j with every unselected individual from population;
    Calculate ϕ_w for every possible swap w;
    $VmaxN \leftarrow max(\phi_w)$
end
if $VmaxN \leq Vmax$ **then**
    Reject the swap and keep j;
    Remove the z^{th}position from f;
else
    Accept the swap;
    $Vmax = VmaxN$;
    $f \in \{1, 2, ..., S\} \setminus i$;
    $nf = S - 1$
end

- When all the individuals with respect to traits $j \in \{2, 3, ..., J\}$ (e.g., height) are acceptable, $\theta_1 = 1$ and the focus will be only on the trait of interest (e.g., yield).
- The sum of all weights in equation (3.17) equals 1 ($\theta_1 + \sum_{j=2}^{J} \frac{(1-\theta_j)}{J-1} = \frac{\sum_{j=2}^{J} \theta_j}{J-1} + \sum_{j=2}^{J} \frac{(1-\theta_j)}{J-1} = 1$).
- The larger that θ_1 is, more weight is placed on the trait of interest (e.g., yield) in selection and mating decisions.

After evaluation of the MT-LAS objective function, the next step is to optimize the model. A similar heuristic algorithm from Moeinizade et al. (2019) is used to optimize the MT-LAS model. This algorithm is defined as follows.

3.3.6 Example with Illustration

In this section, we illustrate the MT-LAS method with an example to provide a more intuitive description. Assume that the goal is maximizing yield (trait 1) while ensuring that plant height (trait 2) falls within a desired range. For a given selection and mating strategy at the current generation (t), the look-ahead stochastic simulation predicts the GEBV of individuals in the final generation (T) with respect to both traits as illustrated in Figure 3.1.

In this example, 20 random progeny are produced in the final generation. The GEBVs of these progeny for both traits are approximated with the look-ahead algorithm. In Figure 3.1, the green and blue bars represent the GEBVs for each progeny with respect to traits 1 and 2 (i.e., $g_1(x, y, r, \tau)$ and $g_2(x, y, r, \tau)$, respectively). GEBVs for plant height are constrained to fall between 15 and 35. Hence, among all progeny, lines 1, 6, 7, 8, 9, 12, 13, 15, 19, and 20 are not acceptable for plant height. These progeny are distinguished from progeny that meet the height requirements with a cross mark. Because 10 out of 20 individuals meet the height criterion, θ_1 and θ_2 are both 0.5. Finally, the penalty (Δ) and penalized GEBVs for each progeny are calculated using equations (3.15) and (3.17), respectively. Penalized GEBVs are plotted as the purple bars in Figure 3.1.

After sorting the progeny with respect to penalized GEBVs, we can calculate the objective ϕ. Let's assume $\gamma = 0.90$. The 90^{th} percentile among 20 individuals is the third best individual and according to Figure 3.1, line 14 is the third best individual. Hence we have $\phi = 21$ which is the value of $h(x, y, r, \tau)$ for line 14.

3.3.7 Data Availability

Data are available at Figshare (DOI: 10.25380/iastate.12145752).

3.4 Results

In this section, we present a case study with real data. Without loss of generality, two traits—total kernel weight (TKW) and ear height (EHT)—are used in this case study. Our objective is to maximize TKW given a constraint on EHT.

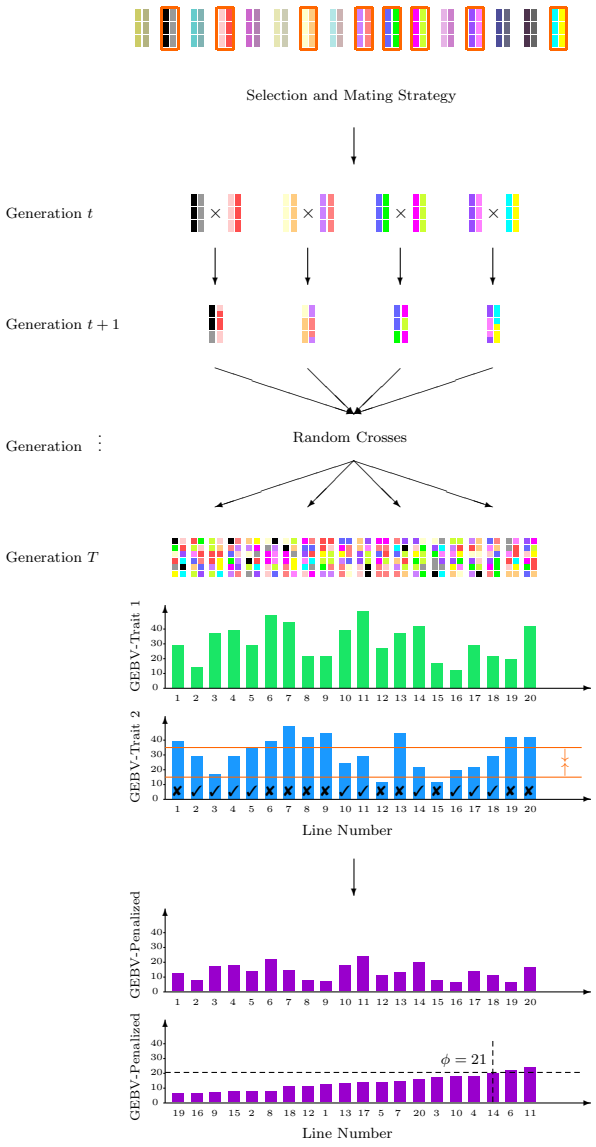

Figure 3.1 The look-ahead simulation illustration for MT-LAS method. In this example, the population consists of 16 individuals. In generation t, 8 individuals are selected from the population and mated accordingly to make 4 crosses. Each breeding parent produces one progeny in generation $t+1$ and from generation $t+1$ to $T-1$ all progeny are crossed with each other in the same generation, each producing one progeny. Then, the look-ahead objective can be approximated by taking a random sample of progeny in generation T. In this example, 20 lines are produced and the GEBV of each individual with respect to traits 1 and 2 are measured and visualized with green and blue bars respectively. Our goal is to maximize trait 1 after $T-t$ generations while making sure that trait 2 does not exceed a certain value of $u = 35$ and is not less than $l = 15$. We observe that 10 individuals among 20 are not acceptable. The progeny with acceptable values for bounded trait are distinguished with check marks. The penalized GEBVs are calculated and represented as purple bars and calculation of the objective ϕ is demonstrated for a given γ.

We first present the performance of ST-LAS where the goal is to maximize TKW only. In this way, we can observe the behavior of EHT versus TKW in the absence of any constraints on EHT. Then, we investigate truncation selection on a selection index for TKW and EHT with different choices of weights. However, this does not allow keeping a trait within a specified range. Hence, we define a penalized index by assigning a negative weight on the absolute deviations from the specified range. The penalized index is used as a benchmark against the performance of MT-LAS. Finally, we present the MT-LAS results and compare the effectiveness of MT-LAS to that of the penalized index.

3.4.1 Single-trait Look-ahead Selection - Maximizing TKW

In this section, we investigate the behavior of EHT over ten generations when the objective is to maximize TKW and there is no constraint on EHT. This will help provide reasonable bounds for EHT when testing the MT-LAS algorithm.

Figure 3.2 (A) presents the total kernal weight and ear height GEBVs over ten generations for a single simulation when selection is only on TKW. Over ten generations the mean GEBV of TKW increases from 2.78 to 40.25 with a maximum of 47.09. For EHT, the range of GEBVs changes from [−21.87,25.42] to [−1.68,14.77]. Figure 3.2 (B) presents the minimum, mean, and maximum GEBVs of both traits over ten generations averaged over 100 simulation replicates. On average, the GEBVs of EHT fall in a range of [−2.73, 16.54] in the final generation.

3.4.2 Index Selection - Maximizing TKW and EHT with Assigned Weights

A selection index is a linear combination of traits according to some weighting scheme. Typically, truncation selection is applied to the index. Here, we construct an index for TKW and EHT where the index is the weighted sum of the GEBVs for each trait ($W_{\text{TKW}}\text{GEBV}_{\text{TKW}} + W_{\text{EHT}}\text{GEBV}_{\text{EHT}}$) and truncation selection is applied to the index. Let W_{TKW} and W_{EHT} be the weights placed on the GEBVs for total kernel weight and ear height, respectively. Weights are chosen from the real numbers between -1 and 1. It should be noted that, in this scheme, we are selecting for larger values

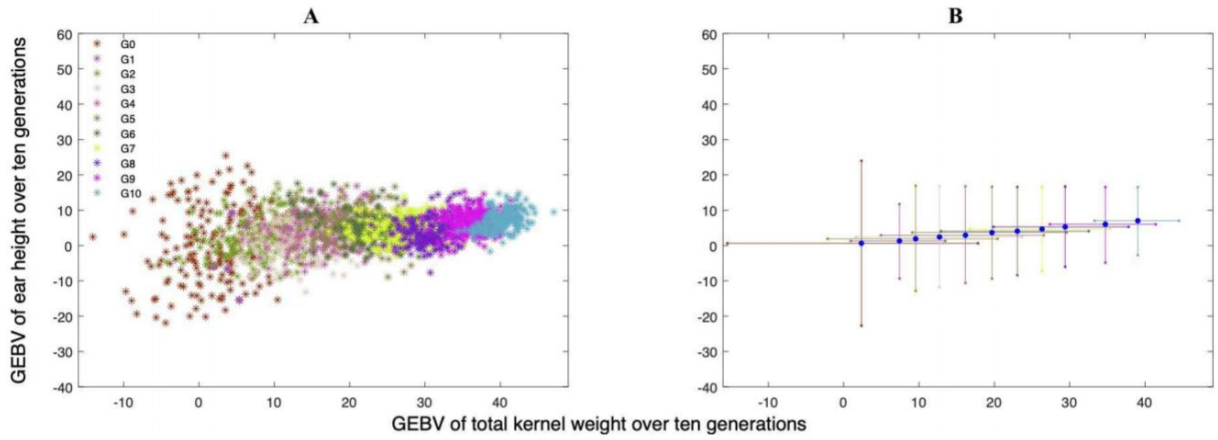

Figure 3.2 **(A)** Population GEBVs of EHT versus TKW for one simulation replicate over ten generations when selection and mating decisions are optimized using ST-LAS algorithm with an objective of maximizing TKW. Each generation includes 200 individuals represented by stars and different colors are distinguishing between generations. The final generation has a minimum, mean, and maximum of 34.36, 40.25, 47.09 for TKW and −1.68, 7.17, 14.77 for EHT respectively. **(B)** Minimum, mean and maximum GEBVs of TKW and EHT over ten generations averaged over 100 simulation replicates. Selection and mating decisions are optimized using ST-LAS algorithm with an objective of maximizing TKW. The final generation has a minimum, mean, and maximum of 33.30, 39.04, 44.51 for TKW and −2.73, 7.00, 16.54 for EHT respectively.

of both TKW and EHT. Placing a negative weight on a trait selects for smaller values and produces progress in the opposite direction to that under strictly positive weights. Figure 3.3 presents the average GEBVs over ten generations averaged over 100 replicate simulations under index selection with varying weights, including the case where the weight on EHT is negative.

As expected, increased weight for EHT (positive or negative) negatively impacts the efficiency of selection for TKW. The mean GEBVs for both traits change in the direction of their assigned weights over time, indicating the lack of strong genetic correlations between TKW and EHT. The

highest mean GEBV for TKW (34.67) is achieved by selection solely on TKW ($W_{\text{TKW}} = 1, W_{\text{EHT}} = 0$). Table 3.1 provides the minimum, mean, and maximum GEBVs for TKW in the final generation under the different choices for weights. It should be noted that the maximum GEBV for TKW achieved after 10 generations of selection is less than that achieved using ST-LAS (36.05 vs. 44.51). This considerable impact on response is due to the fact that the look-ahead selection focuses on maximizing the expected GEBV of the best offspring in the terminal generation, considering uncertainty in recombination in each generation whereas truncation selection on GEBVs focuses on maximizing the genetic gain in the next generation. Additionally, look-ahead selection selects pairs of individuals as a group and recognizes the importance of mating.

Table 3.1 Summary statistics of population GEBV values in generation 10 averaged over 100 replicate simulations for TKW using conventional genomic selection with different weights (index selection). These results are based on simulations in Figure 3.3.

W_{TKW}	W_{EHT}	Min	Mean	Max
0	± 1	0.22	2.81	5.41
0.1	± 0.9	1.86	4.63	7.53
0.2	± 0.8	4.23	7.08	9.86
0.3	± 0.7	7.21	10.28	13.3
0.4	± 0.6	12.07	15.30	18.43
0.5	± 0.5	18.53	21.68	24.69
0.6	± 0.4	25.86	28.77	31.6
0.7	± 0.3	30.21	32.57	34.85
0.8	± 0.2	32.00	33.81	35.49
0.9	± 0.1	32.69	34.11	35.48
1	0	33.29	34.67	36.05

3.4.3 Penalized Index Selection - Maximizing TKW with a Constraint on EHT

In this section, we reformulate the index selection to be able to specify a desired range for the secondary trait. This enables a direct comparison to the MT-LAS method. After applying ST-LAS to TKW, the GEBVs for EHT in the final generation fell between -2.73 and 16.54. We subsequently

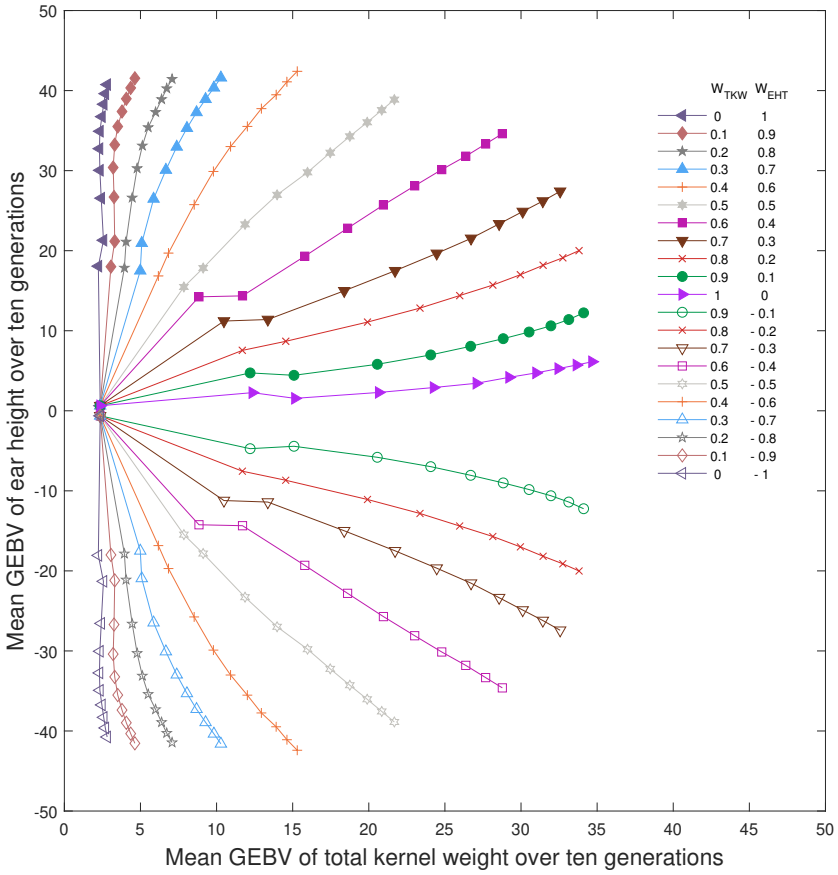

Figure 3.3 Index selection considering different weights for TKW and EHT averaged over 100 simulation replicates. The mean GEBV of individuals over ten generation are calculated given a pair of weights for two traits. The absolute values of the weights add up to 1. Each curve demonstrates the mean GEBV of individuals (represented by markers) over ten generations for assigned weights.

investigated three cases where EHT is constrained to fall outside this range of variation. The three cases are as follows:

- Case 1: $l = 20, u = 30$
- Case 2: $l = -15, u = -5$
- Case 3: $l = 45, u = +\infty$

Similar to the use of a quadratic index to approach an optimum phenotype (Kempthorne and Nordskog, 1959; Wilton et al., 1968), we define an index that penalizes the absolute deviations from a desired range. The constructed index is formulated as $W_{\text{TKW}}\text{GEBV}_{\text{TKW}} - W_{\text{EHT}} \max(l - \text{GEBV}_{\text{EHT}}, 0, \text{GEBV}_{\text{EHT}} - u)$. Weights are chosen from the real numbers between 0 and 1, constrained to sum to unity. Figure 3.4 presents the average GEBVs over ten generations averaged over 100 replicate simulations under penalized index selection for three different cases. These results are compared against the index selection without penalization from Figure 3.3. We observe that the non-penalized index selection cannot satisfy the ear height criterion. As expected, over multiple generations of selection the GEBV of TKW increases and the penalty term accommodates keeping EHT within the specified range. For case 1 and case 2, the EHT criterion is satisfied when $W_{\text{EHT}} \geq 0.3$. However, for case 3, the criterion cannot be satisfied even with the penalized index selection because the bound represents an extreme case. The behavior of case 3 is very similar to the index selection without penalization.

3.4.4 Multi-trait Look-ahead Selection - Maximizing TKW with a Constraint on EHT

The MT-LAS method aims to maximize genetic gain in a target trait while ensuring that one or more secondary traits fall within specified boundaries. Here, we maximize TKW subject to constraints on EHT for three different cases.

Population GEBVs over ten generations for one simulation replicate are presented in Figure 3.5 (A) and the average of 100 simulation replicates are presented in Figure 3.5 (B). For cases 1 and 2, the GEBVs for EHT of $\sim 90\%$ of the individuals in the final generation fall within the specified

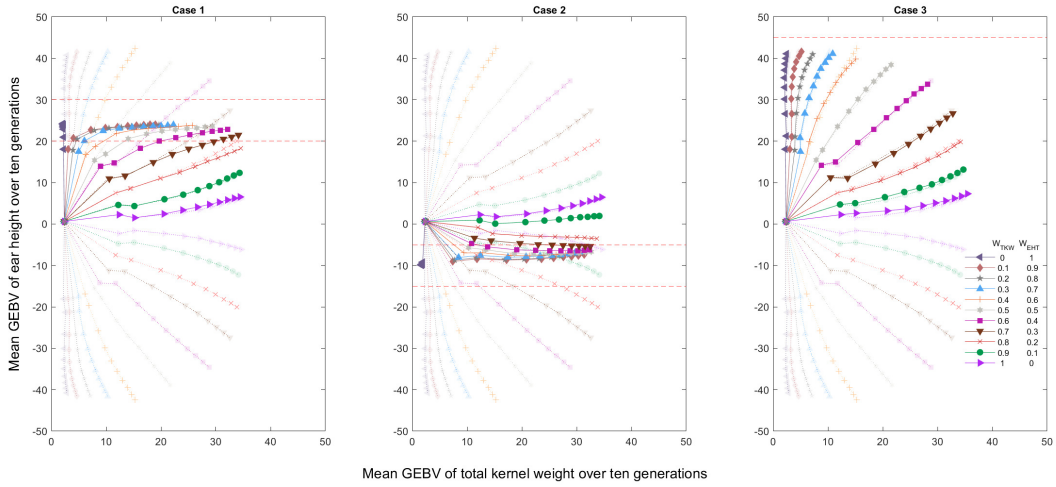

Figure 3.4 Penalized index selection considering different weights for TKW and EHT averaged over 100 simulation replicates for three different cases. Each curve demonstrates the mean GEBV of individuals (represented by markers) over ten generations for assigned weights. The transparent curves in the background present the index selection results without penalization and the red dashed lines are the decision boundaries.

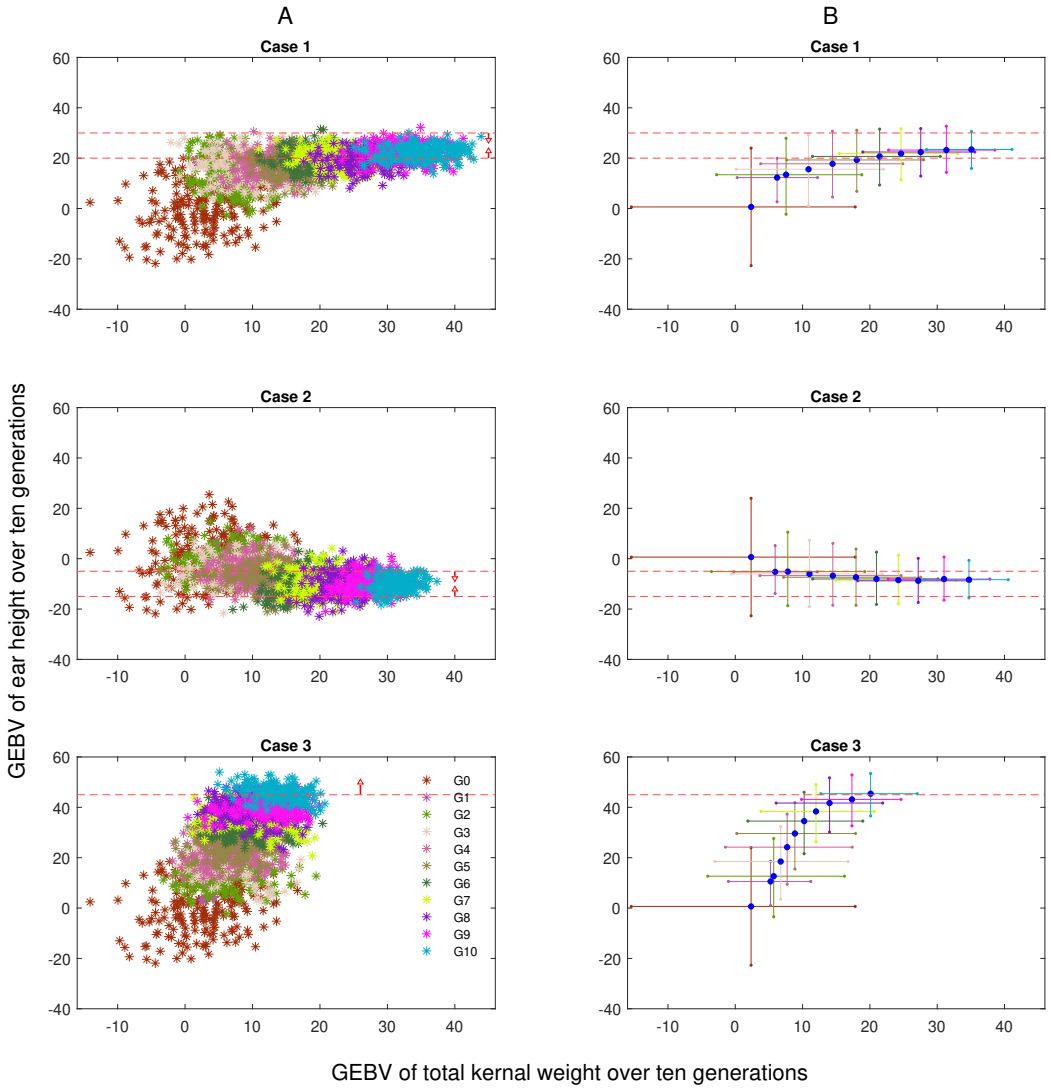

Figure 3.5 (**A**) GEBVs of individuals over ten generations for one simulation replicate. Optimal selection and mating decisions were made using the MT-LAS method in all three cases. Generations are distinguished with different colors. Over multiple generations of selection, the GEBV of TKW increases and the GEBV of EHT falls within the desired range. The red dashed lines are the decision boundaries and the arrows demonstrate the direction for which the condition is satisfied. (**B**) Minimum, mean and maximum GEBVs over ten generations averaged over 100 simulation replicates. The blue markers in the middle of cross marks are the mean GEBVs and the end of the cross marks represent minimum and maximum GEBVs.

boundaries. For case 3, only a lower bound on EHT GEBV was specified. This bound represents an extreme case where index selection is unable to achieve the lower bound even when selecting solely on EHT. However, MT-LAS is able to exceed the bound with $\sim 50\%$ of the population falling into the acceptable range.

3.4.5 Comparison - Performance of MT-LAS Against Penalized Index Selection

Figure 3.6 compares the performance of MT-LAS with the results of non-penalized and penalized index selection using weights that produced results satisfying the desired ranges for 3 cases. We also show that ST-LAS for TKW exceeds the performance of truncation selection on TKW alone ($W_{\text{TKW}} = 1, W_{\text{EHT}} = 0$). For both cases 1 and 2, MT-LAS is able to produce populations that surpass the performance of the comparable index selection scenarios with respect to TKW and also keep almost all individuals within the specified boundaries for EHT. For case 3, the highest EHT achieved by index selection with or without penalization cannot satisfy the desired range criterion. However, MT-LAS not only achieves the expected EHT, but also improves the TKW considerably.

Overall, using MT-LAS with optimization of selection and mating decisions and a soft penalty on ear height improves the response. It should be noted that the distributions of look-ahead methods are quite different from index selection. As shown in Figure 3.6 the look-ahead methods achieve wider distributions in the terminal generation.

Figures 3.7 and 3.8 display the standard deviation of population GEBVs over ten generations for 100 simulation replicates and compare the performance of MT-LAS/ST-LAS with index selection. As expected, look-ahead methods maintain more genetic variance than index selection indicating that there is greater room for population improvement after 10 generations. Furthermore, the genetic correlations between two traits are presented over time for one simulation replicate which indicate the lack of strong correlation between TKW and EHT (see Figure 3.9 in Appendix).

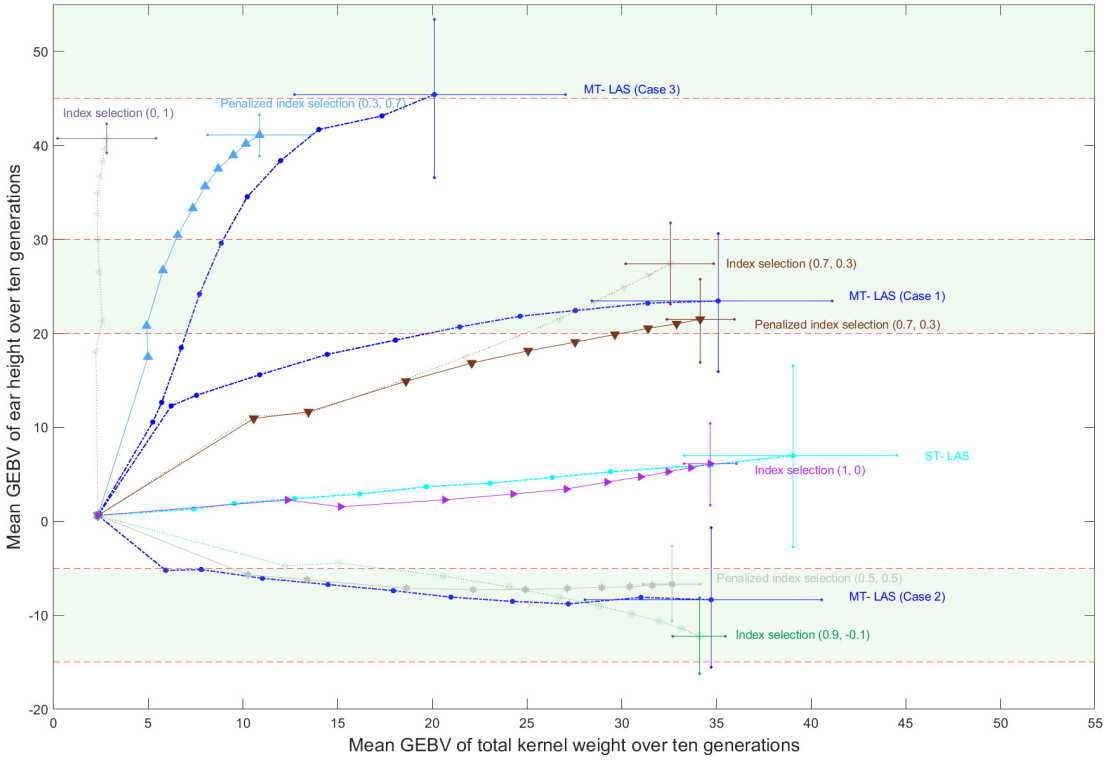

Figure 3.6 Comparison of MT-LAS, ST-LAS and Index selection methods. The mean GEBVs of population over ten generations are averaged over 100 simulation replicates and represented for two traits. Furthermore, the minimum and maximum GEBVs in the final generation are demonstrated using the cross marks. The green bar specifies the boundaries.

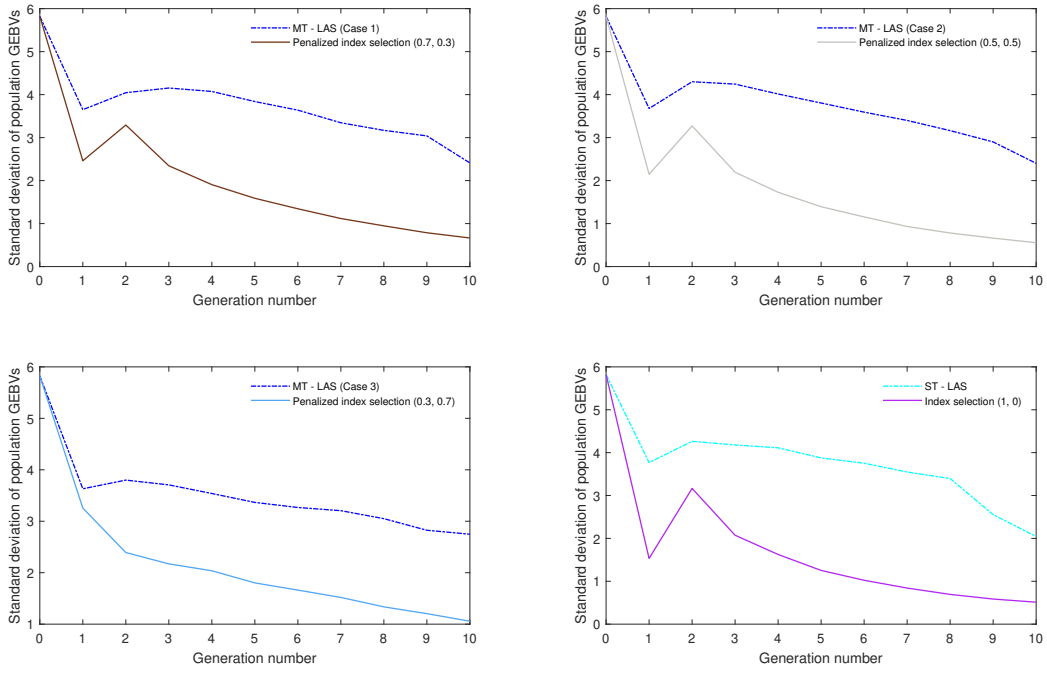

Figure 3.7 Standard deviations of total kernel weight GEBVs over time averaged for 100 simulation replicates.

3.5 Discussion

The production of a crop variety depends on multiple characteristics such as grain quality, yield, and drought resistance which are subject to different breeding objectives. In this study, we proposed a new multi-trait selection approach using genomic information that maximizes genetic gain with respect to a focal trait while controlling the variation in multiple secondary traits.

To demonstrate the effectiveness of the proposed method, we conducted a case study using real data where MT-LAS is compared with index selection with varying weights. In this case study, the goal was to maximize total kernel weight while constraining ear height. Three different cases with varying bounds were investigated, and the results suggested that MT-LAS was more effective at balancing multiple traits than index selection.

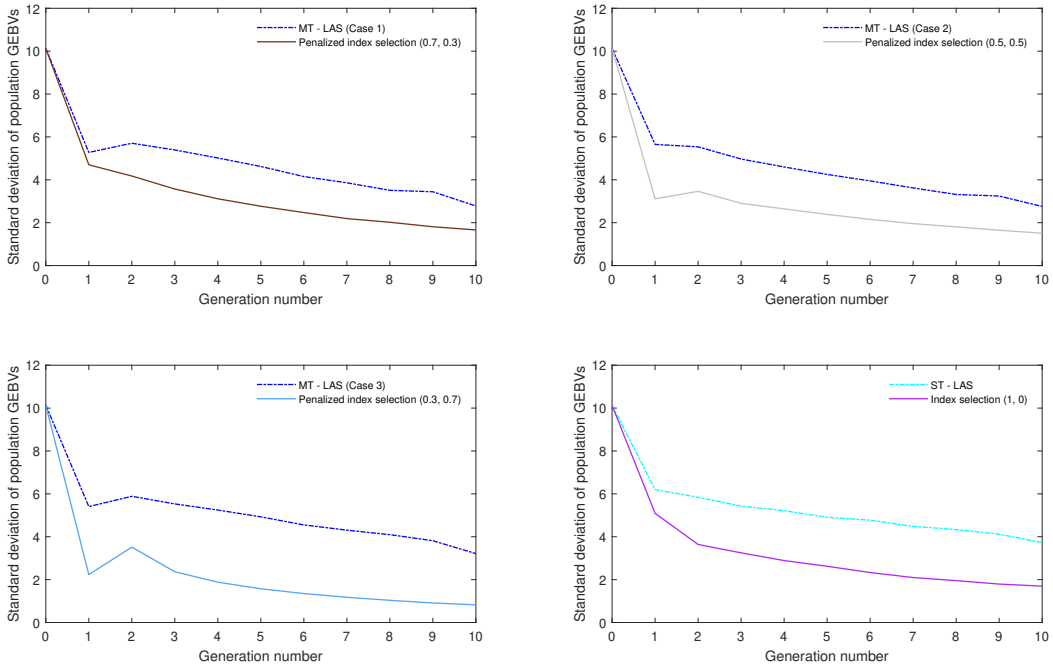

Figure 3.8 Standard deviations of ear height GEBVs over time averaged for 100 simulation replicates.

Fundamentally, the MT-LAS algorithm surpassed conventional index selection because of four reasons. The first reason is the **satisfiability** of this method. MT-LAS automatically and dynamically balances multiple traits and is able to optimize selection and mating decisions in a way that satisfies the constraints for bounded traits while simultaneously maximizing the main trait of interest in the terminal generation. For two of our three scenarios, the penalized index was able to satisfy the constraints on the bounded trait but at the cost of reduced performance in the maximized trait. Moreover, with index selection, it may not be possible to achieve some values for the bounded traits without mate selection. For example, in case 3, we investigate the performance of MT-LAS with a lower bound of 45 which is not reached with either non-penalized or penalized index selection (Figures 3.3 and 3.4).

The second advantage of MT-LAS is its **dynamic adjustability**. The MT-LAS method places more emphasis on feasibility requirements (having individuals that meet the thresholds for the bounded traits) when most of the individuals are not predicted to fall within the bounds for the bounded traits in the terminal generation. On the other hand, this algorithm focuses on the main trait when most of the individuals become acceptable for the bounded trait. Overall, selection and mating decisions are dynamically adjusted in every generation by making a trade off between optimizing the main goal and reaching the desired range for the bounded traits.

A third benefit of MT-LAS is its **interpretability**. By defining the weights in terms of bounds on the desired values of the trait, MT-LAS provides an intuitive description of the breeding objective on the original measurement scale.

A fourth benefit of MT-LAS is its **time-awareness**. As opposed to classical index selection, which maximizes genetic merit in the next generation, MT-LAS maximizes genetic merit in an arbitrary terminal generation. This is similar to work on look-ahead mate selection in animal breeding (Hayes et al., 1998; Shepherd and Kinghorn, 1998; Hayes et al., 2002) where the quantity to be maximized is the genetic merit of grand-progeny. Additionally, this shift alleviates the difficulties posed by the dependence of classical non-linear indices on the current generation mean and intensity of selection which can cause such an index to be non-optimal over multiple generations (Weller et al., 1996).

The main contribution of MT-LAS is constraints (3.14), (3.16), and (3.17), which allow the algorithm to dynamically adjust the objective function according to the progress of the current population. Future research is needed to more fully characterize the MT-LAS algorithm and address the limitations of this study. First, the current paper only considers two traits, although the model is formulated for J traits. Further simulations to explore the behavior of the algorithm when constraining ¿1 trait are desirable. Second, the hyper-parameter γ plays a crucial role in identifying the optimal selection and mating decisions. In this study we selected γ after experimenting with several values. Future work is needed to design systematic methods for optimizing this parameter. Third, the objective of the look-ahead selection relates to the final generation and future research

can focus on designing new selection methods that also consider intermediate generations in the objective. Finally, we based our simulations on a single data set from a single crop organism. Further simulations considering more diverse populations are necessary to demonstrate the general applicability of MT-LAS.

3.6 Acknowledgements

This work is supported by Agriculture and Food Research Initiative Grant no. 2017-67007-26175/Accession No. 1011702 from the USDA National Institute of Food and Agriculture. Any opinions, findings, conclusions, or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the view of the U.S. Department of Agriculture. This work is also supported by the Plant Sciences Institute's Faculty Scholars program at Iowa State University.

3.7 Appendix

3.7.1 Genetic Correlations

Figure 3.9 demonstrates population GEBVs for one simulation replicate over ten generations for different methods. The standard deviation of population GEBVs for TKW and EHT are presented over time. It is observed that almost in every generation the population has higher genetic variation for both traits when selection and mating decisions are optimized using look-ahead methods. Furthermore, the genetic correlations between two traits are presented over time which shows these two traits are correlated with a low degree.

3.7.2 Repeatability of the Results

The simulations are stochastic because they model stochastic recombination events. Figure 3.10 (Right) depicts the distribution of breeding values in the final generation for 100 simulations using the same starting population but different random seeds. The left panels provide a closer look at the first 10 simulations. As expected, there is variation around the average performance across all

simulations. The average of the first 10 simulations is similar to the average of all 100 simulations, suggesting that the results are repeatable.

Figure 3.9 Comparison of the population performance for MT-LAS, ST-LAS and index selection methods over ten generations for one simulation replicate. The gray bars specify boundaries. Each box has three numbers including standard deviation of population GEBVs for trait 1 and trait 2 as well as the correlation between two traits from top to bottom respectively.

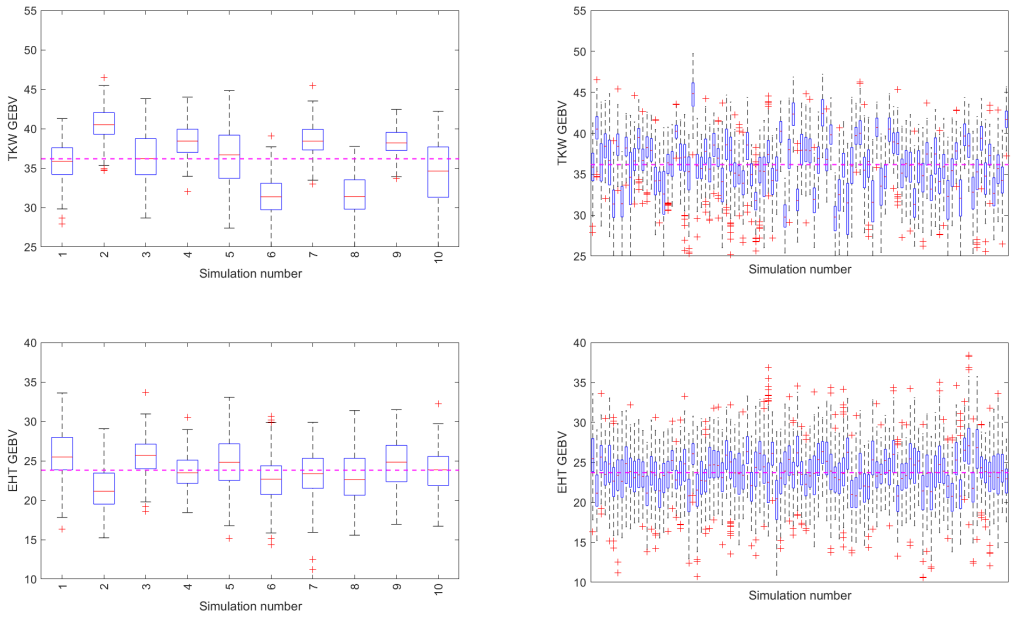

Figure 3.10 Population GEBV box-plots for 10 and 100 independent simulations ((Left) and (Right) respectively). Selection and mating decisions are optimized using MT-LAS method (with an objective of maximizing TKW and having a constraint on EHT (lower-bound 20 and upper-bound 30, similar to case 1)). The purple dashed line demonstrates the average of GEBVs across all simulations.

3.8 References

Akdemir, D., Beavis, W., Fritsche-Neto, R., Singh, A. K., and Isidro-Sánchez, J. (2019). Multi-objective optimized genomic breeding strategies for sustainable food improvement. *Heredity*, 122(5):672.

Akdemir, D. and Sánchez, J. I. (2016). Efficient breeding by genomic mating. *Frontiers in Genetics*, 7:210.

Bates, D., Mächler, M., Bolker, B., and Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software, 67(1): 1-48. doi: 10.18637/jss.v067.i01.*

Brascamp, E. (1984). Selection indices with constraints. *Anim. Breed. Abst.*, 52:645–654.

Burgess, J. C. and West, D. (1993). Selection for grain yield following selection for ear height in maize. *Crop Science*, 33(4):679–682.

Chang, C. C., Chow, C. C., Tellier, L. C., Vattikuti, S., Purcell, S. M., and Lee, J. J. (2015). Second-generation plink: rising to the challenge of larger and richer datasets. *Gigascience*, 4(1):7.

Cowling, W. A., Li, L., Siddique, K. H., Banks, R. G., and Kinghorn, B. P. (2019). Modeling crop breeding for global food security during climate change. *Food and Energy Security*, 8(2):e00157.

Daetwyler, H. D., Hayden, M. J., Spangenberg, G. C., and Hayes, B. J. (2015). Selection on optimal haploid value increases genetic gain and preserves more genetic diversity relative to genomic selection. *Genetics*, 200(4):1341–1348.

Fernando, R. and Garrick, D. (2009). Gensel—user manual for a portfolio of genomic selection related analyses. *Technical report. Available at: http://bigs.ansci.iastate.edu/bigsgui. Accessed: June 13, 2017.*

Goddard, M. (2009). Genomic selection: prediction of accuracy and maximisation of long term response. *Genetica*, 136(2):245–257.

Goiffon, M., Kusmec, A., Wang, L., Hu, G., and Schnable, P. S. (2017). Improving response in genomic selection with a population-based selection strategy: Optimal population value selection. *Genetics*, 206(3):1675–1682.

Hayes, B., Shepherd, R., and Newman, S. (2002). Look ahead mate selection schemes for multi-breed beef populations. *Animal Science*, 74(1):13–23.

Hayes, B., Shepherd, R., Newman, S., and Kinghorn, B. (1998). A tactical approach to improving long term response in across breed mating plans.

Hazel, L. and Lush, J. L. (1942). The efficiency of three methods of selection. *Journal of Heredity*, 33(11):393–399.

Hazel, L. N. (1943). The genetic basis for constructing selection indexes. *Genetics*, 28(6):476–490.

Itoh, Y. and Yamada, Y. (1988). Linear selection indices for non-linear profit functions. *Theoretical and Applied Genetics*, 75(4):553–560.

Ivkovich, M. and Koshy, M. (2002). Optimization of multiple trait selection in western hemlock (tsuga heterophylla (raf.) sarg.) including pulp and paper properties. *Annals of Forest Science*, 59(5-6):577–582.

Jannink, J.-L., Orf, J., Jordan, N. R., and Shaw, R. G. (2000). Index selection for weed suppressive ability in soybean. *Crop Science*, 40(4):1087–1094.

Kempthorne, O. and Nordskog, A. W. (1959). Restricted selection indices. *Biometrics*, 15(1):10–19.

Kinghorn, B. and Kinghorn, A. (2016). Instructions for matesel.

Kinghorn, B. and R.K.Shepherd (1999). Mate selection for the tactical implementation of breeding programs. *Proceedings of the Advancement of Animal Breeding and Genetics*, 13:130–133.

Kusmec, A., Srinivasan, S., Nettleton, D., and Schnable, P. S. (2017). Distinct genetic architectures for phenotype means and plasticities in zea mays. *Nature Plants*, 3(9):715.

Lee, M., Sharopova, N., Beavis, W. D., Grant, D., Katt, M., Blair, D., and Hallauer, A. (2002). Expanding the genetic map of maize with the intermated b73× mo17 (ibm) population. *Plant Molecular Biology*, 48(5-6):453–461.

Long, J., Holland, J. B., Munkvold, G. P., and Jannink, J.-L. (2006). Responses to selection for partial resistance to crown rust in oat.

Makowsky, R., Pajewski, N. M., Klimentidis, Y. C., Vazquez, A. I., Duarte, C. W., Allison, D. B., and de Los Campos, G. (2011). Beyond missing heritability: prediction of complex traits. *PLoS Genetics*, 7(4):e1002051.

Meuwissen, T., Hayes, B., and Goddard, M. (2016). Genomic selection: A paradigm shift in animal breeding. *Animal Frontiers*, 6(1):6–14.

Meuwissen, T. H. E., Hayes, B. J., and Goddard, M. E. (2001). Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157(4):1819–1829.

Moav, R. and Hill, W. (1966). Specialised sire and dam lines. iv. selection within lines. *Animal Science*, 8(3):375–390.

Moeinizade, S., Hu, G., Wang, L., and Schnable, P. S. (2019). Optimizing selection and mating in genomic selection with a look-ahead approach: An operations research framework. *G3: Genes, Genomes, Genetics*, pages g3–200842.

Moeinizade, S., Wellner, M., Hu, G., and Wang, L. (2020). Complementarity-based selection strategy for genomic selection. *Crop Science*, 60(1):149–156.

Pasternak, H. and Weller, J. (1993). Optimum linear indices for non-linear profit functions. *Animal Science*, 56(1):43–50.

Schaeffer, L. (2006). Strategy for applying genome-wide selection in dairy cattle. *Journal of Animal Breeding and Genetics*, 123(4):218–223.

Sharma, R. and Duveiller, E. (2003). Selection index for improving helminthosporium leaf blight resistance, maturity, and kernel weight in spring wheat. *Crop Science*, 43(6):2031–2036.

Shepherd, R. and Kinghorn, B. (1998). A tactical approach to the design of crossbreeding programs. In *Proceedings of the sixth world congress on genetics applied to livestock production*, volume 25, pages 431–438.

Suontama, M., Kinghorn, B., Cowling, W., and Dungey, H. (2018). Tactical desired gains for control of red needle cast in radiata pine under optimal contributions selection.

Villanueva, B. and Woolliams, J. (1997). Optimization of breeding programmes under index selection and constrained inbreeding. *Genetics Research*, 69(2):145–158.

Wang, L., Zhu, G., Johnson, W., and Kher, M. (2018). Three new approaches to genomic selection. *Plant Breeding*, 137(5):673–681.

Weller, J., Pastermak, H., and Groen, A. (1996). Selection indices for non-linear breeding objectives, selection for optima. *Interbull Bulletin*, (12).

Wilton, J., Evans, D. A., and Van Vleck, L. (1968). Selection indices for quadratic models of total merit. *Biometrics*, pages 937–949.

Yan, W. and Frégeau-Reid, J. (2008). Breeding line selection based on multiple traits. *Crop Science*, 48(2):417–423.

Yang, J., Ramamurthy, R. K., Qi, X., Fernando, R. L., Dekkers, J. C., Garrick, D. J., Nettleton, D., Schnable, P. S., et al. (2018). Empirical comparisons of different statistical models to identify and validate kernel row number-associated variants from structured multi-parent mapping populations of maize. *G3: Genes, Genomes, Genetics*, 8(11):3567–3575.

Yu, J., Holland, J. B., McMullen, M. D., and Buckler, E. S. (2008). Genetic design and statistical power of nested association mapping in maize. *Genetics*, 178(1):539–551.

CHAPTER 4. A LOOK-AHEAD MONTE CARLO SIMULATION METHOD FOR IMPROVING PARENTAL SELECTION IN TRAIT INTROGRESSION

Saba Moeinizade[1], Ye Han[2], Hieu Pham[2], Guiping Hu[1], and Lizhi Wang[1]

[1]Department of Industrial and Manufacturing Systems Engineering, Iowa State University,

[2]Syngenta

Modified from a manuscript published in *Scientific Reports*

4.1 Abstract

Multiple trait introgression is the process by which multiple desirable traits are converted from a donor to a recipient cultivar through backcrossing and selfing. The goal of this procedure is to recover all the attributes of the recipient cultivar, with the addition of the specified desirable traits. A crucial step in this process is the selection of parents to form new crosses. In this study, we propose a new selection approach that estimates the genetic distribution of the progeny of backcrosses after multiple generations using information of recombination events. Our objective is to select the most promising individuals for further backcrossing or selfing. To demonstrate the effectiveness of the proposed method, a case study has been conducted using maize data where our method is compared with state-of-the-art approaches. Simulation results suggest that the proposed method, look-ahead Monte Carlo, achieves higher probability of success than existing approaches. Our proposed selection method can assist breeders to efficiently design trait introgression projects.

4.2 Introduction

From a commercial breeding perspective, trait introgression (TI) is a necessary process to produce the elite cultivar with the most desirable traits. This technique is used to incorporate desired traits from a donor into an existing elite cultivar, preserving the performance of the elite

cultivar and adding the benefits of the introduced traits. The result is essentially the same elite cultivar with the added desired traits that will bring benefits to growers (Ødegård et al., 2009).

As an illustration, imagine two maize populations: one population (recipient) characterized by high yielding potential and low resistance to drought stress, whereas the other population (donor) characterized by low yield potential and high resistance to drought stress. In this scenario, one would hope to recover all the attributes of the recipient while obtaining the drought resistant alleles of the donor by some mechanized breeding process to create a new elite cultivar.

Marker-assisted backcrossing strategies provide important time and quality advantages over classical procedures for introgression of desirable alleles from a donor to an elite cultivar (Ragot et al., 1995; Visscher et al., 1996; Frisch and Melchinger, 2005; Khan et al., 2018; Sun and Mumm, 2015). Backcrossing is a well-known breeding approach for the introgression of a target gene from a donor cultivar into the genomic background of a recipient cultivar (Visscher et al., 1996; Frisch et al., 1999; Frisch and Melchinger, 2005; Peng et al., 2014; Khan et al., 2018). The donor parent (DR) provides the desired trait and may not perform as well as an elite variety in other areas. The elite cultivar, called the recurrent parent (RP), usually performs well in the background. The objective is to increase the recipient genome content of the progenies, by repeated backcrosses to the recipient cultivar to recover all the attributes of the recipient cultivar, with the addition of the specified desirable traits (Bouchez et al., 2002).

Although, in principle, the intent of trait introgression is forthright, in practice, there exists many complications due to the stochastic nature and size of a commercial breeding program. Because of this uncertainty, multiple breeding generations may be required until the superior, desired cultivar is achieved (Wang et al., 2011). An additional challenge of the TI process is selecting the most promising backcross individuals for further backcrossing or selfing (Frisch and Melchinger, 2005; Frisch et al., 1999). At each backcross generation cycle, plant breeders are faced with the difficult decision of identifying crosses to perform to produce the next generation of, hopefully, superior cultivar. In the prefect scenario, plant breeders would be able to cross every possible combination of parents until the desired cultivar is achieved. However, in reality, due to

the limited amount of available resources (time, money, land, technology, etc.), breeders may only consider a small fraction of an existing gene pool, possibly leading to sub-optimal decision making (Allier et al., 2019; Twyford and Ennos, 2012; Dempewolf et al., 2017).

Recent advances in simulation and optimization techniques have been applied to variety of disciplines including plant breeding (Li et al., 2012; Shahhosseini et al., 2019; Ansarifar et al., 2020; Shahhosseini et al., 2020; Hosseini et al., 2020; Günay et al., 2020; Haghiri et al., 2018). Computer simulation approaches help identify optimal breeding strategies by adopting assumptions of the breeding system and running multiple scenarios, whereas, optimization approaches aim to produce the best framework to maximize the probability of achieving the desired cultivar while minimizing input resources. It should be noted that the combination of analytical techniques and plant breeding has mainly been applied to genomic selection and not trait introgression (Moeinizade et al., 2019, 2020a,b; Muleta et al., 2019; Yao et al., 2018; Berro et al., 2019; Moeinizade, 2018).

Although there does not currently exist much literature to integrate operations research techniques and trait introgression, there are still a few impactful studies. Cameron et al. utilized an operations research framework with a stochastic optimization model to identify the best breeding strategies for a given population under resource constraints (Cameron et al., 2017). This work illustrates the potential optimization modeling can have on resource allocation in plant breeding. Probabilistic simulation techniques have also been performed by Sun et al. to assess *in silico* various crossing schemes and breeding approaches (Sun and Mumm, 2016). Moreover, Han et al. has framed trait introgression as an algorithmic process and introduced a novel selection metric, predicted cross value (PCV), which predicts specific combining ability by estimating the probability that a pair of parents will produce a perfect gamete with all desirable alleles (Han et al., 2017).

Due to the importance of optimizing the breeding pipeline and the need to consider resource limitations for large scale breeding programs, this paper aims to design a platform that integrates operations research methods to trait introgression. Specifically, the authors develop a novel Monte Carlo simulation approach for the TI pipeline to consider the parental selection aspect under different scenarios of resources present within a commercial scale TI program. The originality in the

proposed method, look-ahead Monte Carlo (LMC), is to look ahead and estimate the performance of progeny in the target generation and then optimize the selection decisions based on the estimated performance. In this study, we use computer simulations to compare selection strategies with respect to the recurrent parent background gene recovery percentage of individuals in the final generation.

4.3 Methods

In this section, we first define the problem by describing the backcrossing breeding pipeline and introduce two existing selection methods. Then, we propose the novel look-ahead Monte Carlo selection method.

4.3.1 Problem Definition

The following abbreviations will be used in subsequent sections in this paper:

DR: donor

RP: recurrent parent

BCt: backcross population at generation t

BCTF2: self-fertilized population after final backcross

GEBV: genomic estimated breeding value

PCV: predicted cross value

LMC: look-ahead Monte Carlo

The general objective of trait introgression projects is to produce a new line that is highly close to the recurrent parent and contains the desired alleles or traits from the donor parent. First, an initial cross is made between the donor and the recurrent parent to produce F1 progeny. Since, the donor and recurrent parents are both homozygous, this step is deterministic which means the F1 progeny has 50% of the genetic material from each parent. Next, the F1 individual is crossed to the recurrent parent to develop a backcross one (BC1) population. Figure 4.1 represents a schematic overview of the backcross project where the ultimate goal is to produce drought resistant individual

plants with good agronomics. In Figure 4.1, we see n individuals in the backcross one population denoted with $\mathrm{BC1}_1, \mathrm{BC1}_2, ..., \text{and } \mathrm{BC1}_n$. Best individuals from the BC1 population were selected based on a selection strategy and then again backcrossed to the recurrent parent.

In successive generations, progeny are first selected for the trait of interest and then backcrossed to the recurrent parent. This process is repeated for T backcross generations. We refer to an individual as positive if it contains the desirable alleles from the donor. For positive individuals in BCT population, the percentage of background recovery was calculated by dividing the number of desirable alleles in the background by the total number of background alleles. Furthermore, we monitor and evaluate the BCTF2 individuals.

To simulate the recombination process during meiosis, we used the same inheritance distribution defined in Han et al. (2017). In subsequent sections, the recombination frequency vector is denoted by $r \in [0, 0.5]^{L-1}$, where L is the total number of markers in the genome. To represent the genotype of an individual plant, we use an $L \times m$ binary matrix, say $G \in \mathbf{B}^{L \times m}$, where $G_{l,m} = 1$ indicates whether the l^{th} allele from chromosome m is desirable or not ($G_{l,m} = 0$). For each individual plant represented with a binary matrix, each row is a locus in the genome. The number of columns in the binary matrix represents the ploidy of the plant. We use diploid species in this paper ($m = 2$). Here, we review two existing approaches for parental selection.

4.3.1.1 Background Selection

The background selection approach first selects the individuals with desired marker genotypes and then among these positive individuals selects for the desired background genotype (Hospital et al., 1992; Visscher et al., 1996; Frisch et al., 1999). Background selection has been shown to be efficient by previous theoretical work (Hillel et al., 1990; Hospital et al., 1992; Groen and Smith, 1995; Visscher et al., 1996) and experimental work (Ragot et al., 1995).

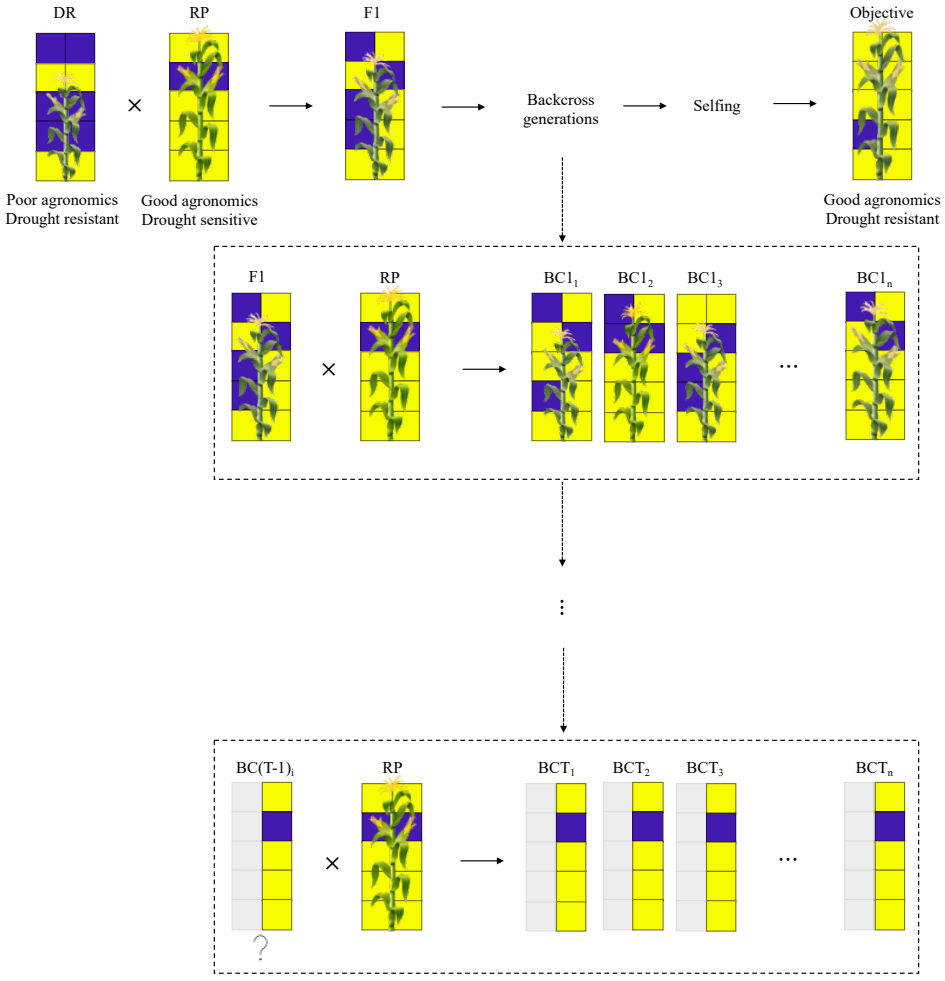

Figure 4.1 A schematic overview of the backcross project. The donor parent (poor agronomics, drought resistant) is crossed with the recurrent parent (good agronomics, drought sensitive) to produce F1. F1 progeny have 50% of their genetic material from each parent (yellow square: favorable allele, purple square: unfavorable allele). Then, F1 is backcrossed with the recurrent parent to develop the BC1 population. Best individuals from BC1 are selected based on a predefined metric and backcrossed to the recurrent parent. This process is repeated for T generations. The ultimate goal is to achieve an individual which is drought resistant and has good agronomics.

The breeding value of a background genotype can be estimated using genomic estimated breeding value (GEBV) (Meuwissen et al., 2001; Bernardo, 2009). GEBV of individual plants (or animals) is defined as the sum of their estimated marker effects (Meuwissen et al., 2001). We assume $\mathbf{D} = \{d_1, d_2, ..., d_z\}$ is the location of the positive markers from the donor and there are total Z markers that should be introgressed. If we assume a uniform weight for all desirable alleles, then the background GEBV of an individual is equivalent to the number of desirable alleles in its background:

$$GEBV = \sum_{\substack{l=1 \\ l \notin D}}^{L} \sum_{m=1}^{2} G_{l,m} \tag{4.1}$$

According to this approach, the positive individuals with highest GEBVs will be selected as parents.

4.3.1.2 Predicted Cross Value Selection

The predicted cross value (PCV) calculates the probability that a pair of breeding parents will produce a gamete with desirable alleles at all specified loci by taking into account the recombination frequencies (Han et al., 2017). This approach selects individuals based on their likelihood to produce an elite gamete by combining all desirable alleles. Since in a backcrossing scheme, individuals are always crossed with the recurrent parent, the PCV can be defined as the probability that each individual will produce an elite gamete.

Let $g \in \mathbf{B}^{L \times 1}$ denote a random gamete produced by a breeding parent. The PCV of an individual is calculated as follows:

$$PCV(G, r) = Pr(g_i = 1, \forall i \in \{1, 2, ..., L\}) \tag{4.2}$$

To calculate this probability, the same water-pipe algorithm described in Yao et al. (2018) is used. The rationale for the PCV definition is to calculate the probability that none of the undesirable alleles survives two generations of meiosis (Han et al., 2017). According to this approach, the positive individuals with the highest PCVs will be selected as parents.

4.3.2 Proposed Look-ahead Monte Carlo Algorithm

In this section, we propose a novel probabilistic and heuristic driven search algorithm, look-ahead Monte Carlo (LMC) for parental selection. The underlying concept is to use Monte Carlo simulation for modeling uncertainty involved due to recombination events. Monte Carlo simulation is a technique that relies on repeated random sampling to obtain numerical results (Caflisch, 1998). This technique is often used in physical and mathematical problems and is most suited to be applied when it is impossible to obtain a closed-form expression or infeasible to apply a deterministic algorithm (Bihani, 2014).

The look-ahead Monte Carlo algorithm for parental selection evaluates different selection decisions periodically during the learning phase by predicting the genetic distribution of the progeny of backcrosses after multiple generations using information of recombination events. This algorithm makes a trade-off between exploration and exploitation. It exploits the selection strategies that is found to be best until the current generation and explores the alternative decisions to find out if they could replace the current best. The essence of this algorithm is to strategically search the space to find optimal crosses that can result in best performance in the targeted generation.

Figure 4.2 presents an overview of the LMC algorithm. For every individual in BCt population (e.g., BCt_i), multiple random gametes are simulated according to the recombination frequencies. These gametes are narrowed down to the ones which have the desirable markers from the donor in the introgressed loci. Then, one of these positive gametes is selected randomly to form the next BC progeny (e.g., BC(t+1)_i). This process is repeated until the target generation (BCT). Finally, individuals are evaluated based on their performance after selfing (BCTF2).

In BCTF2, success can be defined as achieving certain amount of recovery percentage (e.g., 95%) among positive individuals. Suppose the population size of the BCTF2 generation is K and n individuals with desirable markers have achieved the desired recovery percentage. Then $\frac{n}{K}$ is the probability of getting a positive individual that has met the background recovery requirements. Since through backcross generations the gametes are selected randomly, this probability is estimating only one of the possible outcomes for individual i in BCt population. To have a reasonable

approximation for the performance of progeny in BCTF2, the same process should be repeated multiple times. The objective of the LMC algorithm can then be calculated as:

$$Q = \frac{\sum_{j=1}^{P} \frac{n^j}{K} v^j}{P} \tag{4.3}$$

Where v^j represents the maximum recovery percentage achieved in BCTF2 for the j^{th} round, and P is the total rounds of repetition. According to LMC approach, individuals with highest Q values will be selected as the breeding parents.

4.4 Results

In this section, we first describe the data sets used in this case study, and then compare the proposed method with two existing selection methods in different scenarios of resources using computer simulation.

4.4.1 Data

Data contains donor and recipient's genetic information and recombination frequencies. To explore the effect of having different initial genetic similarities between the donor and recurrent parent, we considered three cases as demonstrated in Table 4.1. The genetic similarity is calculated based on the NEIs metric (Nei and Li, 1979). Cases 1, 2, and 3 have low, moderate, and high initial genetic similarities, respectively. Our goal is to compare the performance of selection strategies using these 3 different cases given that the low initial genetic similarity (case 1) is expected to be more difficult relatively.

Case	NEIs	Number of markers
1	0.58%	195
2	0.72%	173
3	0.89%	172

Table 4.1 The description of data sets

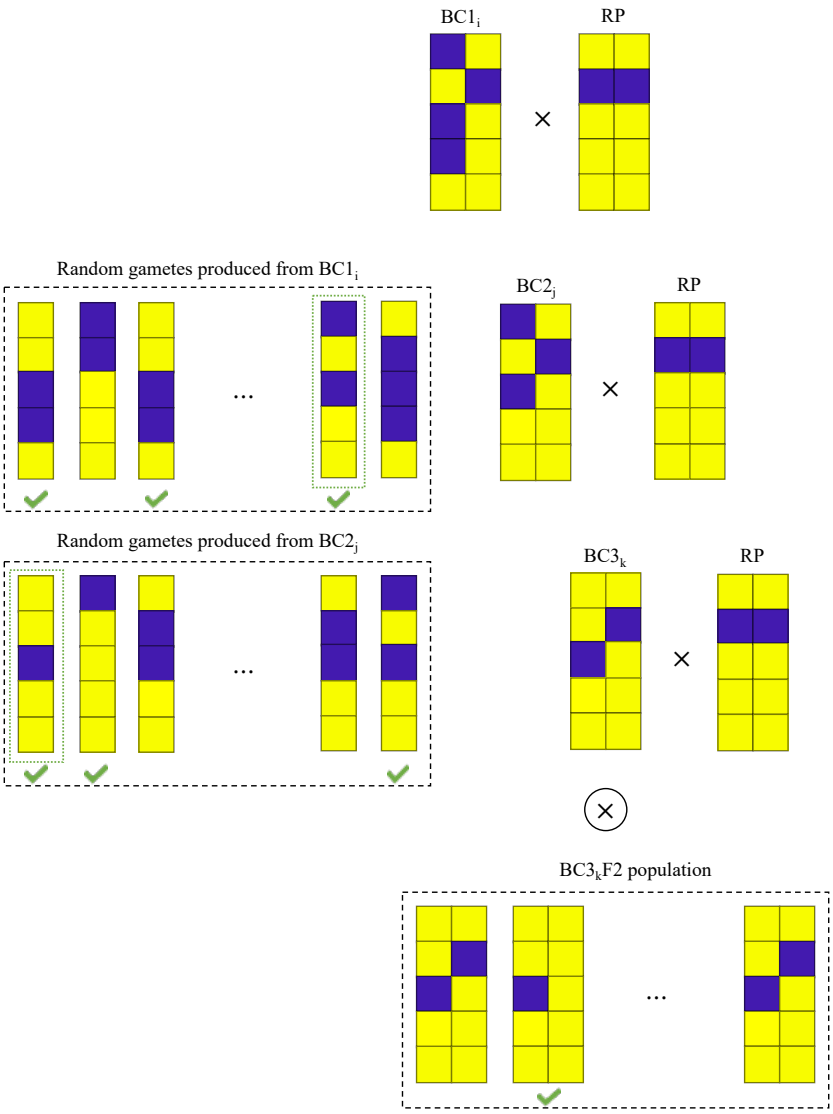

Figure 4.2 The Monte Carlo search for parental selection in trait introgression. For a deadline of T generations, we estimate the performance of BCTF2 individuals for a given selection strategy by searching across all possible paths. This is a schematic overview of monitoring the estimated performance in BC3F2 for a single path. The same process is repeated multiple times and then the average value is assigned to $BC1_i$.

The genetic information of the donor and recurrent parent for these three cases are illustrated in Figure 4.3. For all cases, three markers should be integrated from the donor to the recurrent parent. Furthermore, the recombination events are presented in the supplementary information.

4.4.2 Simulation Settings

Multiple trait introgression was studied using realistic maize data with three different selection methods, including GEBV, PCV, and LMC. We considered three cases with different genetic similarities and two different scenarios for resources. These scenarios are designed considering the practical aspects of a breeding program. In scenario 1, we are allocating limited resources by making 2 crosses in each generation, whereas in scenario 2, we are allocating moderate resources by making 6 crosses in each generation (see Table 4.2). Scenario 1 more closely resembles what occurs in a commercial breeding program, namely, decision making with limited resources.

Generation	Scenario 1 (limited resources)	Scenario 2 (moderate resources)
BC1	2	6
BC2	2	6
BC3	2	6

Table 4.2 Numbers of crosses in each generation for two different scenarios.

One hundred independent simulation replicates were performed for each of the selection methods using MATLAB (R2019-a). Simulation has been performed for three generations of backcrosses followed by selfing. The evaluation is based on the recovery percentage of individuals in BC3F2 generation.

It is assumed that each cross makes 200 progeny and for each scenario the number of crosses remains the same through all generations (i.e., resources are distributed evenly among different generations).

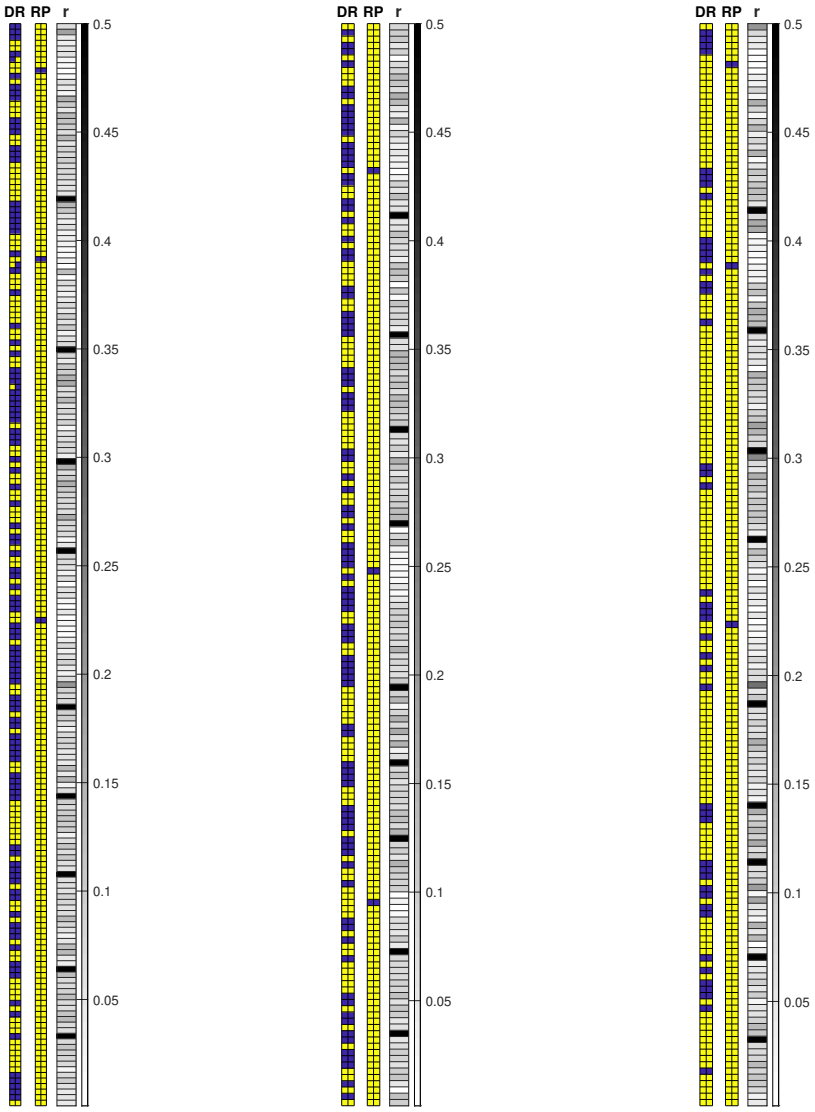

Figure 4.3 Donors and recurrent parents' genetic information and recombination frequencies for three cases (DR: donor, RP: recurrent parent, r: recombination frequency). A yellow square is used to denote a favorable allele ("1") and a purple square is used to denote an unfavorable allele ("0"). The gray charts are heat maps for recombination frequencies.

4.4.3 Simulation Results

Comparison of selection methods for one sample simulation: Figure 4.4 presents the performance of three selection methods for one sample simulation. The histograms of background recovery percentage for positive individuals are demonstrated over BC1, BC2, BC3, and BC3F2 generations.

All three methods start with the same BC1 population and then produce the next population based on different selection decisions. As expected, the background recovery improves from BC1 to BC3 for all selection methods. For this sample simulation, the (maximum, mean, minimum) recovery percentage in BC3 is (94, 90.61, 85), (94, 89.71, 84), (97, 94.07, 92) for GEBV, BPV, and LMC methods respectively which demonstrates improvement in recovery percentage when selection decisions are made using the LMC method.

It should be noted that the BC3F2 individuals should have all 6 alleles desirable in the three markers that are to be integrated from the donor (i.e. BC3F2 individuals are homozygous). However, the BC individuals are expected to have 3 desirable alleles total since their second chromosome is being inherited from the recurrent parent. This can explain why recovery percentage drops from BC3 to BC3F2. As demonstrated in Figure 4.4, for this sample simulation, the LMC method achieves 95% recovery in BC3F2, however the other two selection methods achieve maximum 91% recovery.

Background recovery percentage of the top individual in BC3 across all simulation replicates: Figure 4.5 compares the cumulative distribution functions (CDFs) of maximum recovery percentage achieved in BC3 for three selection methods among 100 simulation replicates. The further toward the right direction CDF curves, the better performance a method has. Take for example, point (97, 75) means that 75% of the simulations have achieved recovery percentage less than or equal to 97. In all cases and scenarios, the LMC method achieves higher recovery percentage.

For case 3, which has the highest genetic similarity between donor and recurrent parent, there was one simulation that resulted in having one individual in BC3 with all desirable traits (100%

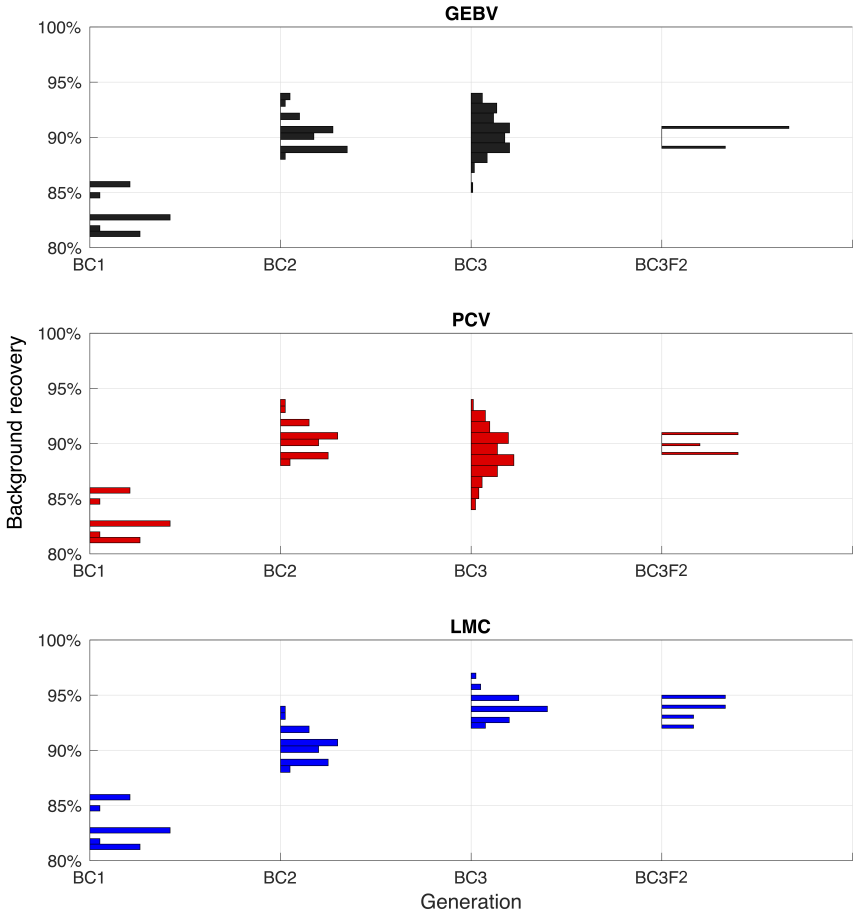

Figure 4.4 A sample simulation result for three different selection methods presenting histograms of population background recovery percentage over different generations. This simulation is performed for case 2, scenario 1.

recovery percentage). Note that since this is a backcross generation, for this individual the second chromosome still lacks the desirable alleles from the donor. As expected, for each case, scenario 2 has better performance compared to scenario 1 since there are more resources available.

Average background recovery percentage of top 10 individuals in BC3 across all simulation replicates: Figure 4.6 presents the box-plots of average recovery percentage of the top 10 individuals in BC3 generation. For all cases and scenarios, the median value is higher when selection decisions are optimized using LMC method. Furthermore, PCV has generally higher median values than GEBV. The overall range of values is greater for LMC method (as shown by the distances between the ends of the two whiskers for each box-plot). The interquartile ranges are reasonably similar (as shown by the lengths of the boxes), except for case 2, scenario 1, where LMC has considerably higher range.

Background recovery percentage of the top individual in BC3F2 across all simulation replicates: Figure 4.7 compares the probability of success for three selection methods by evaluating the recovery percentage of best individual in BC3F2. For example, point (0.8, 95) means that 80% of the simulations have achieved recovery percentage of 95 in the terminal generation. The curves with better performance are expected to be closer to the upper right corner of the plot.

As expected, scenario 2 has generally higher probability of success compared to scenario 1 as more resources are used. Take for example, for case 2, the probability of achieving 95 percentage recovery with LMC method increases from 0.74 to 0.83 when having more resources. This probability also increases from 0.59 to 0.71 for PCV and from 0.54 to 0.69 for GEBV method. Furthermore, the probability of success increases from case 1 to 3 where there is more genetic similarity between the donor and parent. Take scenario 2, for example, the probabilities of having 95 percentage recovery when selection decisions are optimized using LMC method are 0.81, 0.83, 0.89 for cases 1, 2, and, 3, respectively.

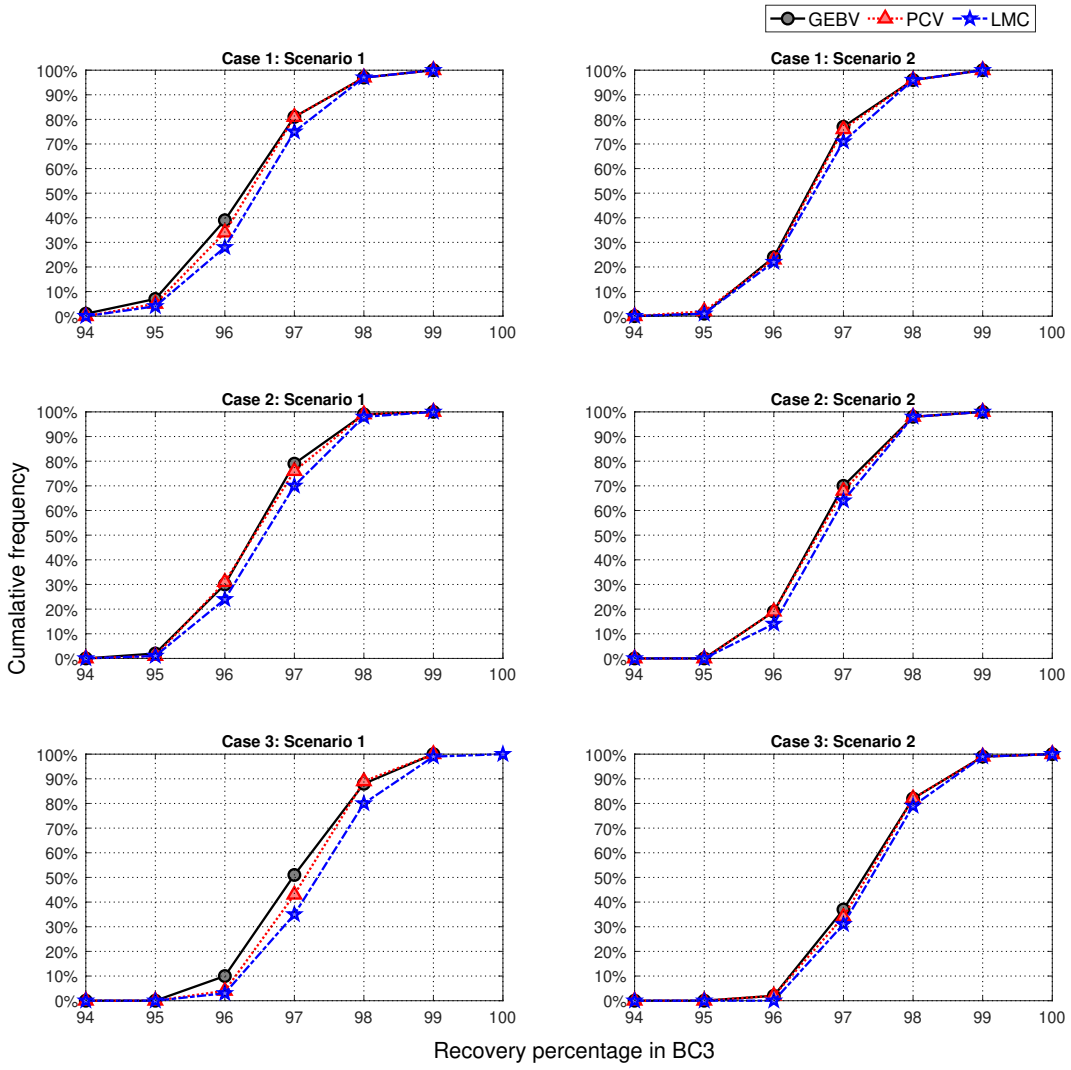

Figure 4.5 Cumulative distribution functions of population maximum in the BC3 for three cases and two different scenarios. Results are based on 100 simulation replicates.

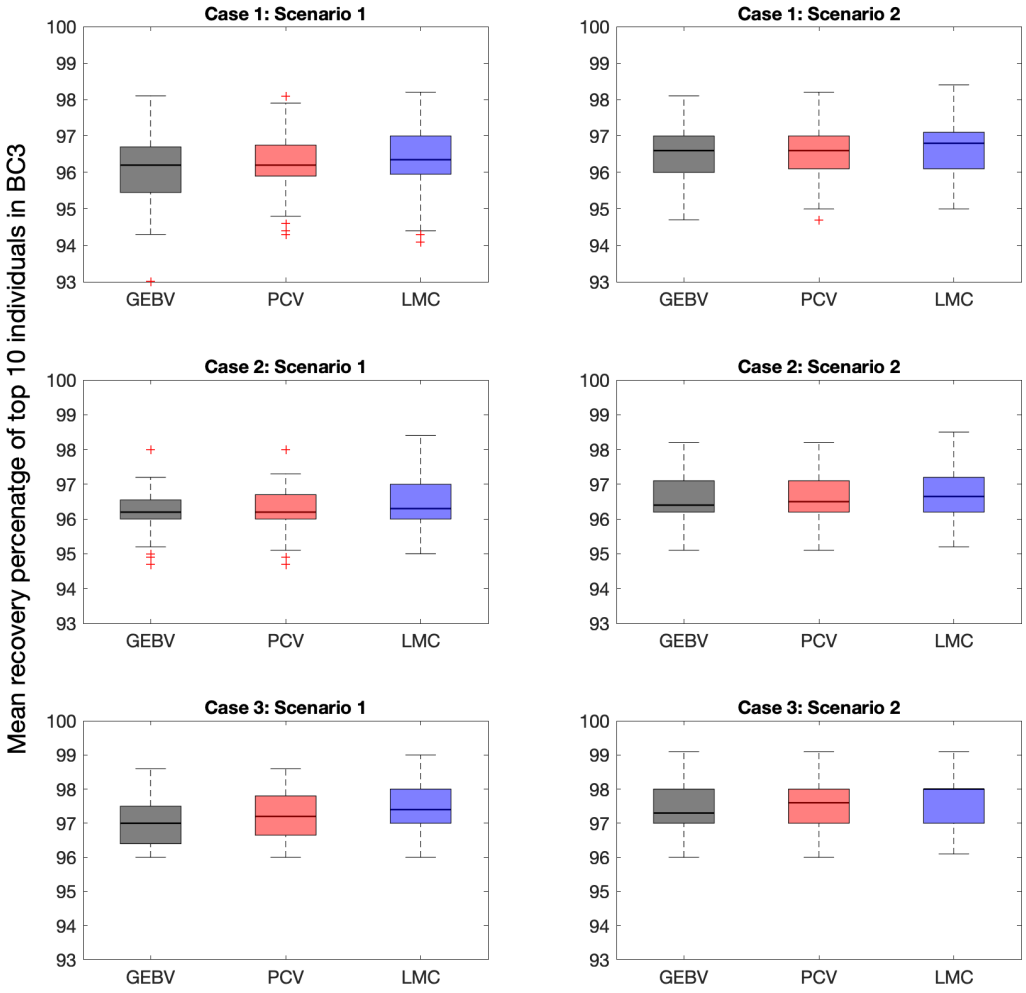

Figure 4.6 Box-plots of mean recovery percentage of top 10 individuals in BC3 for three selection methods. For each case and scenario, 100 simulation replicates are performed. The median values are demonstrated with a bold line.

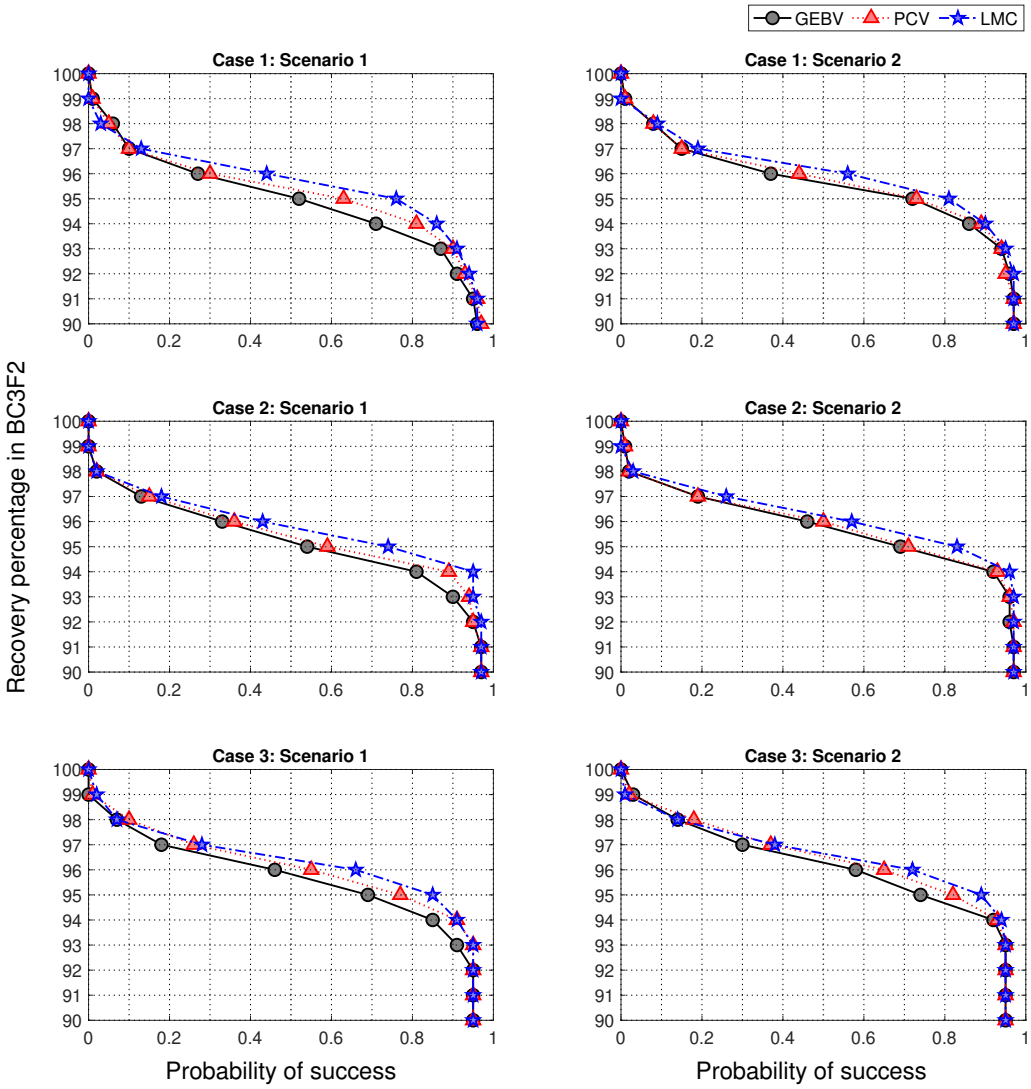

Figure 4.7 Probability of success in BC3F2 for three different selection approaches considering 3 cases of initial genetic similarity and 2 scenarios of resource allocation for 100 simulation replicates. The maximum recovery percentage of positive individuals in BC3F2 is first identified and then probability of success has been defined as the proportion of simulations that have achieved a certain recovery.

4.5 Discussion

Selection methods based on marker information make trait introgression more efficient and effective. When introgressing the desired traits form a donor to a recipient, background selection is the conventional selection approach that aims to recover the desired background genome. Recent advances in optimization and simulation techniques can help enhance the efficiency of parental selection in breeding programs.

In this study, we introduced a new selection method, LMC, which has the potential to further improve the efficiency of breeding given limited time and resources by integrating operations research techniques and trait introgression. The proposed method was compared with existing selection methods in a simulation study using empirical maize data. Computational results demonstrate the improvements of the LMC method over two existing selection approaches, GEBV and PCV.

One of the advantages of the LMC method is being sensitive to the deadline. Unlike other selection methods that evaluate the performance based on only next generation, the LMC method relates the objective to the performance of individuals in the targeted generation. Another advantage of the LMC method is the trade-off between exploration and exploitation. When the look ahead process finds exploitation to contribute more to the final objective, the algorithm behaves in a greedy way to maximize performance. However, when the exploration is found to be more beneficial, the algorithm explores new possible outcomes.

The simulations in this study were designed based on practical considerations. The trait introgression pipeline included three backcross generations followed by a selfing so that selected individuals will be homozygous for the target trait. There is no absolute number for the number of backcrosses needed to be performed but generally between two to five backcrosses are performed in maize. The number of required generations can be determined based on the breeding objective and the resources invested at each generation (Ribaut et al., 2002). Intuitively, making more crosses and producing more progeny leads to a higher chance of creating desirable individuals, however the resources are limited and the breeding strategy should be customized based on the available re-

sources. Here, we considered two scenarios to represent both limited and moderate cases of resource availability. Scenario 1 limits the number of crosses to two in each generation where as scenario 2 allows six crosses. According to the reproductive biology of maize, it is possible to obtain ≈ 200 seeds from a cross. Thus, we assumed each cross makes 200 progeny, which means for scenarios 1 and 2, the population size of each generation becomes 400 and 1200, respectively. As expected, the simulation results demonstrated that the probability of success increases when having more resources.

Future work should investigate optimizing the resource allocation strategies by spreading out the budget systematically among different generations. Moreover, this study investigated introgressing desirable alleles from a single donor, however desirable alleles can be carried by multiple donors. Hence, another direction that deserves investigation is to extend the LMC method for cases with multiple donors. Moreover, we based our simulations on a single crop organism. Further simulations considering more diverse populations are necessary to demonstrate the general applicability of the proposed selection method.

4.6 Acknowledgements

This work was partially supported by Agriculture and Food Research Initiative Grant no.2017-67007-26175/Accession No. 1011702 from the USDA National Institute of Food and Agriculture. Additionally, this work was partially supported by the National Science Foundation under the LEAP HI and GOALI programs (grant number 1830478). This work was also supported by the Plant Sciences Institute's Faculty Scholars program at Iowa State University and Syngenta.

4.7 Appendix

Recombination frequencies for 3 data sets that were used in this study.

	Ch1	Ch2	Ch3	Ch4	Ch5	Ch6	Ch7	Ch8	Ch9	Ch10
Marker1	0.0662	0.1441	0.0865	0.1588	0.0790	0.0915	0.0955	0.0662	0.1266	0.0722
Marker2	0.1812	0.1122	0.0412	0.0705	0.0739	0.0955	0.0782	0.0636	0.0748	0.1176
Marker3	0.0458	0.0574	0.1035	0.1067	0.0592	0.0627	0.1003	0.0610	0.0765	0.0915
Marker4	0.0357	0.0394	0.0979	0.1615	0.0282	0.0253	0.1152	0.0874	0.0906	0.0857
Marker5	0.0512	0.0671	0.1533	0.1051	0.0679	0.0739	0.0955	0.0857	0.1325	0.0731
Marker6	0.0627	0.0310	0.1682	0.0849	0.0060	0.0857	0.0987	0.0592	0.0773	0.0234
Marker7	0.0167	0.0347	0.0697	0.0748	0.0610	0.0857	0.0263	0.0636	0.0636	0.0739
Marker8	**0.0089**	0.0282	0.0697	0.0748	0.0557	0.0503	0.0688	0.1398	0.1145	0.1003
Marker9	0.0050	0.0158	0.1413	0.0636	0.0215	0.0565	0.1176	0.0840	0.1347	0.0592
Marker10	0.0089	**0.0089**	0.0565	0.1688	0.0070	0.0412	0.0653	0.0557	0.0748	0.0412
Marker11	0.0662	0.0253	0.0857	0.0815	0.0530	0.1377	0.1090	0.0282	0.0653	0.0494
Marker12	0.0476	0.0010	0.0583	0.0756	**0.0020**	0.0412	0.0679	0.0512	0.0476	
Marker13	0.0244	0.1333	0.0939	0.0739	0.0196	0.1484	0.0627	0.1281	—	—
Marker14	0.1581	0.0040	0.0244	0.0244	0.0119	0.0301	—	0.1377	—	—
Marker15	0.1051	0.0722	0.0618	0.0512	0.0010	0.0645	—	0.0440	—	—
Marker16	0.0494	0.0430	0.0636	—	0.0375	—	—	0.0394	—	—
Marker17	0.1391	0.0282	0.0832	—	0.0731	—	—	—	—	—
Marker18	0.1191	0.0476	0.0748	—	0.0301	—	—	—	—	—
Marker19	0.0995	0.0782	0.0366	—	0.0583	—	—	—	—	—
Marker20	0.0662	0.0476	—	—	0.0931	—	—	—	—	—
Marker21	0.1470	0.0947	—	—	0.0421	—	—	—	—	—
Marker22	0.0539	0.0618	—	—	0.0319	—	—	—	—	—
Marker23	0.0931	0.1075	—	—	0.0653	—	—	—	—	—
Marker24	0.0765	0.0253	—	—	0.2045	—	—	—	—	—
Marker25	0.1011	0.0583	—	—	0.0874	—	—	—	—	—
Marker26	0.0748	0.0722	—	—	0.0782	—	—	—	—	—
Marker27	0.0714	—	—	—	0.0739	—	—	—	—	—
Marker28	0.0782	—	—	—	—	—	—	—	—	—
Marker29	0.0565	—	—	—	—	—	—	—	—	—
Marker30	0.0592	—	—	—	—	—	—	—	—	—
Marker31	0.0548	—	—	—	—	—	—	—	—	—

Table 4.3 Recombination frequencies for case 1. Each row refers to a marker and each column refers to a chromosome. The recombination frequencies for the three markers that should be integrated from the donor to the recipient are distinguished with an underline.

	Ch1	Ch2	Ch3	Ch4	Ch5	Ch6	Ch7	Ch8	Ch9	Ch10
Marker1	0.0799	0.0485	0.0674	0.0733	0.0028	0.0910	0.0757	0.0855	0.0770	0.0524
Marker2	0.0953	0.0626	0.0618	0.0614	0.0801	0.1472	0.1118	0.0966	0.0724	0.0671
Marker3	0.1057	0.1189	0.1467	0.0990	0.1353	0.0246	0.0587	0.0498	0.0793	0.0959
Marker4	0.0484	0.0868	0.1368	0.0406	0.0279	0.0814	0.0870	0.1510	0.0721	0.0914
Marker5	0.0911	0.0659	0.1306	0.1580	0.0236	0.1506	0.0928	0.1049	0.0894	0.0567
Marker6	0.0040	0.0693	0.0752	0.1174	0.0130	0.0737	0.0861	0.0888	0.0820	0.1009
Marker7	0.0792	0.0392	0.1079	0.0551	0.0113	0.1709	0.0640	0.0915	0.1219	0.1118
Marker8	0.0815	0.1110	0.1199	0.1113	**0.0150**	0.0663	0.1220	0.0966	0.0598	0.0510
Marker9	0.1409	0.1297	0.0646	0.0625	0.0129	0.1496	0.1310	0.0016	0.1178	0.0025
Marker10	0.1164	0.0997	0.1395	0.0790	0.0484	0.0285	0.0739	**0.0258**	0.0979	0.1216
Marker11	0.0670	0.0086	0.1265	0.0975	0.0115	0.0650	0.1451	0.0033	0.0545	0.0966
Marker12	0.1578	0.1076	0.1031	0.0975	0.0481	—	—	0.0146	0.0595	—
Marker13	0.1389	0.0115	0.0823	0.1265	0.0775	—	—	0.0645	—	—
Marker14	0.0542	0.1247	0.1043	0.1357	0.0922	—	—	0.0653	—	—
Marker15	0.0356	0.0729	—	—	0.0899	—	—	0.0541	—	—
Marker16	0.1474	0.0843	—	—	0.1107	—	—	0.1075	—	—
Marker17	0.0213	0.0833	—	—	0.0716	—	—	0.0450	—	—
Marker18	0.0938	0.0365	—	—	0.0923	—	—	—	—	—
Marker19	0.0527	—	—	—	0.0657	—	—	—	—	—
Marker20	0.0311	—	—	—	0.1572	—	—	—	—	—
Marker21	0.0490	—	—	—	0.0714	—	—	—	—	—
Marker22	0.0269	—	—	—	0.0853	—	—	—	—	—
Marker23	**0.0279**	—	—	—	0.0670	—	—	—	—	—
Marker24	0.0080	—	—	—	0.0653	—	—	—	—	—
Marker25	0.0257	—	—	—	0.1101	—	—	—	—	—
Marker26	0.0890	—	—	—	—	—	—	—	—	—
Marker27	0.0490	—	—	—	—	—	—	—	—	—
Marker28	0.1303	—	—	—	—	—	—	—	—	—
Marker29	0.0515	—	—	—	—	—	—	—	—	—
Marker30	0.0636	—	—	—	—	—	—	—	—	—

Table 4.4 Recombination frequencies for case 2.

	Ch1	Ch2	Ch3	Ch4	Ch5	Ch6	Ch7	Ch8	Ch9	Ch10
Marker1	0.2098	0.0722	0.0653	0.2332	0.0421	0.0697	0.1635	0.0697	0.0338	0.0756
Marker2	0.0739	0.1708	0.0384	0.0756	0.1333	0.0592	0.0756	0.0722	0.0557	0.0748
Marker3	0.0476	0.1695	0.0790	0.0503	0.0890	0.0906	0.1035	0.1027	0.0653	0.0874
Marker4	0.0440	0.0128	0.0530	0.1615	0.0592	0.0583	0.1377	0.1888	0.1019	0.1355
Marker5	0.0030	0.0225	0.0512	0.1003	0.0119	0.0739	0.0882	0.0234	0.0539	0.0865
Marker6	**0.0060**	0.0282	0.0060	0.1214	0.0574	0.1648	0.1662	0.1760	0.0987	0.1122
Marker7	0.0050	0.0158	0.1512	0.0731	0.0030	0.1251	0.0679	0.0565	0.0244	0.1051
Marker8	0.0089	**0.0089**	0.1152	0.1129	0.0592	0.0583	0.1027	0.0467	0.0557	0.0756
Marker9	0.0347	0.0263	0.1075	0.0955	0.0375	0.0898	—	0.0177	0.0601	0.0310
Marker10	0.0329	0.0291	0.0756	0.1183	0.0449	0.0688	—	0.1533	0.0898	0.0476
Marker11	0.0874	0.0206	0.0688	0.1043	0.0070	0.0548	—	0.0548	0.1405	—
Marker12	0.0196	0.0963	0.0291	0.0679	0.0530	0.0799	—	0.1413	0.1137	—
Marker13	0.1601	0.1244	0.0923	0.0329	**0.0020**	0.0301	—	0.0020	—	—
Marker14	0.0857	0.0099	0.1051	—	0.0196	0.0601	—	0.0485	—	—
Marker15	0.0244	0.1145	0.1831	—	0.0128	0.0244	—	—	—	—
Marker16	0.0824	0.1628	0.1427	—	0.0384	—	—	—	—	—
Marker17	0.1498	0.1296	0.0832	—	0.0440	—	—	—	—	—
Marker18	0.0375	0.1289	0.0748	—	0.0310	—	—	—	—	—
Marker19	0.0815	—	—	—	0.0824	—	—	—	—	—
Marker20	0.0618	—	—	—	0.0263	—	—	—	—	—
Marker21	0.1574	—	—	—	0.0739	—	—	—	—	—
Marker22	0.0119	—	—	—	0.0705	—	—	—	—	—
Marker23	0.0530	—	—	—	0.2767	—	—	—	—	—
Marker24	0.1114	—	—	—	0.0512	—	—	—	—	—
Marker25	0.0773	—	—	—	0.1051	—	—	—	—	—
Marker26	0.1168	—	—	—	—	—	—	—	—	—
Marker27	0.1281	—	—	—	—	—	—	—	—	—
Marker28	0.1137	—	—	—	—	—	—	—	—	—
Marker29	0.0539	—	—	—	—	—	—	—	—	—

Table 4.5 Recombination frequencies for case 3.

4.8 References

Allier, A., Moreau, L., Charcosset, A., Teyssèdre, S., and Lehermeier, C. (2019). Usefulness criterion and post-selection parental contributions in multi-parental crosses: application to polygenic trait introgression. *G3: Genes, Genomes, Genetics*, 9(5):1469–1479.

Ansarifar, J., Akhavizadegan, F., and Wang, L. (2020). Performance prediction of crosses in plant breeding through genotype by environment interactions. *Scientific Reports*, 10(1):1–11.

Bernardo, R. (2009). Genomewide selection for rapid introgression of exotic germplasm in maize. *Crop Science*, 49(2):419–425.

Berro, I., Lado, B., Nalin, R. S., Quincke, M., and Gutiérrez, L. (2019). Training population optimization for genomic selection. *The Plant Genome*, 12(3).

Bihani, A. (2014). A new approach to monte carlo simulation of operations. *Laser*, 20:0–20.

Bouchez, A., Causse, M., Gallais, A., Charcosset, A., et al. (2002). Marker-assisted introgression of favorable alleles at quantitative trait loci between maize elite lines. *Genetics*, 162(4):1945–1959.

Caflisch, R. E. (1998). Monte carlo and quasi-monte carlo methods. *Acta Numerica*, 7:1–49.

Cameron, J. N., Han, Y., Wang, L., and Beavis, W. D. (2017). Systematic design for trait introgression projects. *Theoretical and Applied Genetics*, 130(10):1993–2004.

Dempewolf, H., Baute, G., Anderson, J., Kilian, B., Smith, C., and Guarino, L. (2017). Past and future use of wild relatives in crop breeding. *Crop Science*, 57(3):1070–1082.

Frisch, M., Bohn, M., and Melchinger, A. E. (1999). Comparison of selection strategies for marker-assisted backcrossing of a gene. *Crop Science*, 39(5):1295–1301.

Frisch, M. and Melchinger, A. E. (2005). Selection theory for marker-assisted backcrossing. *Genetics*, 170(2):909–917.

Groen, A. and Smith, C. (1995). A stochastic simulation study of the efficiency of marker-assisted introgression in livestock. *Journal of Animal Breeding and Genetics*, 112(1-6):161–170.

Günay, E. E., Kremer, G. E. O., and Zarindast, A. (2020). A multi-objective robust possibilistic programming approach to sustainable public transportation network design. *Fuzzy Sets and Systems*.

Haghiri, S., Daghighi, A., and Moharramzadeh, S. (2018). Optimum coagulant forecasting by modeling jar test experiments using anns. *Drinking Water Engineering & Science*, 11(1).

Han, Y., Cameron, J. N., Wang, L., and Beavis, W. D. (2017). The predicted cross value for genetic introgression of multiple alleles. *Genetics*, 205(4):1409–1423.

Hillel, J., Schaap, T., Haberfeld, A., Jeffreys, A., Plotzky, Y., Cahaner, A., and Lavi, U. (1990). Dna fingerprints applied to gene introgression in breeding programs. *Genetics*, 124(3):783–789.

Hospital, F., Chevalet, C., and Mulsant, P. (1992). Using markers in gene introgression breeding programs. *Genetics*, 132(4):1199–1210.

Hosseini, S. A., Alhasan, A., and Smadi, O. (2020). Use of deep learning to study modeling deterioration of pavements a case study in iowa. *Infrastructures*, 5(11):95.

Khan, G. H., Shikari, A. B., Vaishnavi, R., Najeeb, S., Padder, B. A., Bhat, Z. A., Parray, G. A., Bhat, M. A., Kumar, R., and Singh, N. K. (2018). Marker-assisted introgression of three dominant blast resistance genes into an aromatic rice cultivar mushk budji. *Scientific Reports*, 8(1):1–13.

Li, X., Zhu, C., Wang, J., and Yu, J. (2012). Computer simulation in plant breeding. In *Advances in agronomy*, volume 116, pages 219–264. Elsevier.

Meuwissen, T. H. E., Hayes, B. J., and Goddard, M. E. (2001). Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157(4):1819–1829.

Moeinizade, S. (2018). A stochastic simulation approach for improving response in genomic selection. *Master's thesis,*, page Iowa State University.

Moeinizade, S., Hu, G., Wang, L., and Schnable, P. S. (2019). Optimizing selection and mating in genomic selection with a look-ahead approach: An operations research framework. *G3: Genes, Genomes, Genetics*, 9(7):2123–2133.

Moeinizade, S., Kusmec, A., Hu, G., Wang, L., and Schnable, P. S. (2020a). Multi-trait genomic selection methods for crop improvement. 215(4):931–945.

Moeinizade, S., Wellner, M., Hu, G., and Wang, L. (2020b). Complementarity-based selection strategy for genomic selection. *Crop Science*, 60(1):149–156.

Muleta, K. T., Pressoir, G., and Morris, G. P. (2019). Optimizing genomic selection for a sorghum breeding program in haiti: a simulation study. *G3: Genes, Genomes, Genetics*, 9(2):391–401.

Nei, M. and Li, W.-H. (1979). Mathematical model for studying genetic variation in terms of restriction endonucleases. *Proceedings of the National Academy of Sciences*, 76(10):5269–5273.

Ødegård, J., Yazdi, M., Sonesson, A., et al. (2009). Incorporating desirable genetic characteristics from an inferior into a superior population using genomic selection. *Genetics*, 181(2):737–745.

Peng, T., Sun, X., and Mumm, R. H. (2014). Optimized breeding strategies for multiple trait integration: I. minimizing linkage drag in single event introgression. *Molecular Breeding*, 33(1):89–104.

Ragot, M., Biasiolli, M., Delbut, M., Dell'Orco, A., Malgarini, L., Thevenin, P., Vernoy, J., Vivant, J., Zimmermann, R., and Gay, G. (1995). Marker-assisted backcrossing: a practical example. *COLLOQUES-INRA*, pages 45–45.

Ribaut, J.-M., Jiang, C., and Hoisington, D. (2002). Simulation experiments on efficiencies of gene introgression by backcrossing. *Crop Science*, 42(2):557–565.

Shahhosseini, M., Hu, G., Archontoulis, S. V., and Huber, I. (2020). Coupling machine learning and crop modeling improves crop yield prediction in the us corn belt. *arXiv:2008.04060*.

Shahhosseini, M., Martinez-Feria, R. A., Hu, G., and Archontoulis, S. V. (2019). Maize yield and nitrate loss prediction with machine learning algorithms. *Environmental Research Letters*, 14(12):124026.

Sun, X. and Mumm, R. H. (2015). Optimized breeding strategies for multiple trait integration: Iii. parameters for success in version testing. *Molecular Breeding*, 35(10):201.

Sun, X. and Mumm, R. H. (2016). Method to represent the distribution of qtl additive and dominance effects associated with quantitative traits in computer simulation. *BMC Bioinformatics*, 17(1):73.

Twyford, A. and Ennos, R. (2012). Next-generation hybridization and introgression. *Heredity*, 108(3):179–189.

Visscher, P. M., Haley, C. S., and Thompson, R. (1996). Marker-assisted introgression in backcross breeding programs. *Genetics*, 144(4):1923–1932.

Wang, X., Wang, Y., Zhang, G., and Ma, Z. (2011). An integrated breeding technology for accelerating generation advancement and trait introgression in cotton. *Plant Breeding*, 130(5):569–573.

Yao, J., Zhao, D., Chen, X., Zhang, Y., and Wang, J. (2018). Use of genomic selection and breeding simulation in cross prediction for improvement of yield and quality in wheat (triticum aestivum l.). *The Crop Journal*, 6(4):353–365.

CHAPTER 5. A REINFORCEMENT LEARNING APPROACH TO RESOURCE ALLOCATION IN GENOMIC SELECTION

Saba Moeinizade, Guiping Hu, and Lizhi Wang

Department of Industrial and Manufacturing Systems Engineering, Iowa State University

Modified from a manuscript to be submitted to *Information Sciences*

5.1 Abstract

Genomic selection (GS) is a technique that plant breeders use to select individuals to mate and produce new generations of species. Allocation of resources is a key factor in GS. At each selection cycle, breeders are facing the choice of budget allocation to make crosses and produce the next generation of breeding parents. Inspired by recent advances in reinforcement learning for AI problems, we develop a reinforcement learning-based algorithm to automatically learn to allocate limited resources across different generations of breeding. We mathematically formulate the problem in the framework of Markov Decision Process (MDP) by defining state and action spaces. To avoid the explosion of the state space, an integer linear program is proposed that quantifies the trade-off between resources and time. Finally, we propose a value function approximation method to estimate the action-value function and then develop a greedy policy improvement technique to find the optimal resources. We demonstrate the effectiveness of the proposed method in enhancing genetic gain using a case study with realistic data.

5.2 Introduction

Over the past decades breeding methods have evolved from traditional phenotype-based selection to marker-assisted selection methods. Genomic selection (GS), which was initially proposed by Meuwissen et al. (2001), is a special form of marker assisted selection that estimates the effects of

genome-wide markers in a training population consisting of genotyped and phenotyped individuals. Different statistical and machine learning models are proposed to develop prediction models based on the genotypic and phenotypic data of the training population (Pryce et al., 2011; Neves et al., 2012; Chen et al., 2014; Li et al., 2015; Dong et al., 2016; Liu and Wang, 2017; Crossa et al., 2017; Montesinos-López et al., 2018; Covarrubias-Pazaran et al., 2018; Liu et al., 2018). Then, the prediction model is used to derive the genomic estimated breeding values (GEBVs) for all individuals of the breeding population (BP) from their genomic profile by calculating the sum of the estimated marker effects. Given the genotype information and the estimated marker effects of individuals in a breeding population, there are different decisions that should be made within each breeding cycle. These decisions include selection, mating, and resource allocation which must be made in every generation with the objective of continuously improving individuals subject to deadline constraints.

Recently, Moeinizade et al. (2019) presented the look-ahead selection (LAS) method to optimize selection and mating strategies with a time-dependent approach. A new technique was invented to anticipate the consequences of selection and mating decisions through several generations, which was achieved by quantitatively taking into account recombination frequencies. Recombination, the main source of uncertainty in reproductive biology, is the phenomenon that occurs during meiosis and creates different combinations of alleles in the resulting gametes (Lobo and Shaw, 2008). In Moeinizade et al. (2019), we conducted a case study using realistic maize data and compared LAS with other published selection methods. Simulation results suggested the superiority of LAS to other selection methods. However, the LAS method was unable to optimize resource allocation decisions, e.g., how should the budget be distributed over time? Should it be spent evenly or should more investment be made in earlier generations before genetic diversity deteriorates? how many crosses should be made and how many progeny should be produced? These resource allocation decisions should be optimized systematically, given the cost of making a cross and genotyping progeny, under budget and deadline constraints, considering the uncertainty in recombination in each generation.

In this study, we develop a reinforcement learning-based algorithm to automatically learn to allocate resources across different generations of breeding. The proposed new method integrates the LAS approach in a reinforcement learning framework. The LAS method is capable of anticipating the consequences of the selection and mating decisions under uncertain recombination events efficiently and accurately, whereas the reinforcement learning framework is capable of making a trade-off between cost and time which is necessary to make resource allocation decisions.

Reinforcement learning (RL) is one of the most important research directions of machine learning, which has been widely used in different fields like social sciences, natural sciences, and engineering and has significantly impacted the development of Artificial Intelligence (AI) over the last years (Dayan and Niv, 2008). Sutton and Barto (2018) define Reinforcement learning as learning what to do —how to map situations to actions— so as to maximize a numerical reward signal. The main characters of RL are the agent and the environment. The environment represents the outside world to the agent and the agent interacts with the environment by taking actions and receiving a reward signal. The goal of the agent is to maximize the cumulative reward, named return. To do that, the agent should learn the optimal policy which is an optimal strategy to behave in the environment.

RL problems can be formulated mathematically in the framework of Markovian Decision Processes (MDPs) by defining states, actions, transition probabilities, and rewards (Szepesvári, 2010). The transition and reward functions in MDPs are called the model of environment. A known MDP can be solved by dynamic programming which relies on simplifying a complicated problem by breaking it down into simpler sub-problems in a recursive manner (Bellman, 1966). However, we often do not have the transition and the rewards of the MDP. This class of problems with unknown MDPs are called model-free. While model-based methods rely on planning as their primary component, model-free methods rely on learning (Sutton and Barto, 2018). Model-free methods can be applied to both prediction and control problems. In model-free prediction, the goal is to estimate the value function of an unknown MDP where as model-free control aims at optimizing the value function. The value function represents how good it is for an agent to be in a given state.

In recent years, different solution methods have been proposed to solve model-free RL problems (Mnih et al., 2013; Schulman et al., 2015a,b; Hausknecht and Stone, 2015; Van Hasselt et al., 2015; Wang et al., 2016; Schulman et al., 2017; Heess et al., 2017; Tucker et al., 2018). These solution methods include two main types of algorithms, value-based and policy-based. Value-based algorithms iteratively update the value of a state to finally learn an optimal policy. Policy-based algorithms learn a parameterized policy that can select actions without consulting a value function.

Q-learning, a value-based RL algorithm, is one of the most popular solution methods in reinforcement learning. This algorithm uses Q-values (an estimation of how good it is to take an action at a given state) to iteratively improve the behaviour of the learning agent (Watkins and Dayan, 1992). However, for large-scale problems with an enormous number of state-action pairs, it is difficult to explicitly store all the Q-values. To overcome this challenge, function approximation methods are used where value function is represented by mapping a state description to a value (Gosavi, 2009; Kaelbling et al., 1996; Arulkumaran et al., 2017). Many implementations of RL in real-world problems have used neural networks as function approximators (Mnih et al., 2013; Hausknecht and Stone, 2015; Van Hasselt et al., 2015; Wang et al., 2016). One of the examples is the achievement of AlphaGo in 2016, where a deep Q-network was implemented and trained to predict total reward (Silver et al., 2016). Other approximation methods including kernel methods, nearest-neighbor algorithms, and decision trees can be used to estimate the Q-values (Friedman et al., 2001; Chapman and Kaelbling, 1991; Howe and Pyeatt, 1998). Policy gradient algorithms learn in a more robust way by approximating policy and updating it according to the gradient of expected reward with respect to the policy parameters (Sutton et al., 1999) without the need to construct a value function.

In this study, we propose a value-based algorithm with function approximation and introduce a backward greedy policy approach with respect to the estimated values (i.e., the policy that selects the action with highest estimated value in each state). The idea of the backward approach is to learn the optimal action in a backward manner starting from the final generation to the first generation given that the optimal strategy in the final generation is allocating all remaining resources. In the

remainder of this paper, we formulate the resource allocation problem in an RL framework, discuss the solution methods and finally present a case study to compare our proposed allocation strategy with even allocation using computer simulation.

5.3 Methods

In this section, we first define the genomic selection resource allocation problem and then formulate the proposed problem mathematically in the context of Markov Decision Process (MDP), where reinforcement learning algorithms can be used. Finally, we provide a solution method to solve the proposed MDP and find the optimal policy.

5.3.1 Problem Definition

A classical plant breeding process starts with an initial population and iteratively goes through the selection and reproduction steps until getting the final population. In addition to the selection decisions, in each generation, the breeder should decide how to allocate resources (i.e., the number of crosses to be made and the number of progeny to be produced from each cross). The focus of this study is optimizing the resource allocation strategy in a breeding program.

Let $G_t \in \mathbb{B}^{L\times M\times N}$ represent the genotype of the population at generation t, where L is the total number of alleles, M indicates the ploidy of the plant ($M = 2$ for diploid species) and N is the total number of individuals in the population. For all l, let β_l denote the additive effect of allele l, which is assumed to have been reasonably estimated. Given β and G, the look-ahead selection (LAS) algorithm can optimize the selection and mating steps with a time-dependent approach (Moeinizade et al., 2019) by maximizing the expected GEBV of the best offspring in the terminal generation (T) where GEBV of an individual can be calculated as the sum of all marker effects across the entire genome.

Let the cost of producing one progeny be one unit of budget. Then, spending b units of resources in the current generation will produce b progeny. Given a fixed amount of total budget of B_0 units of resources over T generations, the goal is to find the optimal budget or population

size for each generation, $(b_1, b_2, ..., b_T)$, in order to maximize the performance of individuals in the final generation. Similar with selection and mating, resource allocation decisions should be made in a dynamic manner after observing the genotype of progeny from previous generations, while considering the total budget constraint over T generations: $\sum_{t=1}^{T} b_t \leq B_0$.

5.3.2 Problem Formulation

Here, we present the MDP formulation for the genomic selection resource allocation problem. An MDP process is described by a finite set of states (S), a finite set of actions (A), transition probabilities (T), and a reward function (R). Due to the stochastic nature of the environment, in this problem, we cannot derive the transition probabilities and the reward is delayed until the terminal generation. Hence, we use learning to understand the behavior of the environment by simulating different scenarios of resource allocation (section 5.3.3). In this section, we define the state and action spaces for the MDP.

5.3.2.1 State Space

To capture the full information in each generation, the population genotype would be necessary to define the state space, but it would make the resulting model unsolvable. For example, for a small population of 200 individuals and only $10,000$ pairs of genes, the dimension of the state space would be $3^{2,000,000}$ with each pair of genes taking three possible combinations of two variants of alleles (AA, aa, or Aa). To avoid formidable dimensions, we need to simplify the state space by presenting a compact definition that captures the important information by considering the current genetic value of the population and quantifying the trade-off between time and resources.

At generation t, we define the state by $(g_t^{\max}, C_t, B_{t-1})$, where $g_t^{\max}$ is the highest GEBV of the N individuals at generation t calculated as follows:

$$g_t^{\max} = \max_{n \in \{1,2,...,N\}} \left(\sum_{l=1}^{L} \sum_{m=1}^{2} G_t^{l,m,n} \beta^l \right) \tag{5.1}$$

In this state space definition, $C_t \in \mathbb{R}^{K\times M}$ measures the specific combining ability, and B_{t-1} is the available budget to be spent in generations t to the end. Specifically, $C_t^{k,m}$ is the highest possible GEBV of a gamete that could be assembled from G_t using at most m individuals with recombination events that are more likely than p^k, where $p \in (0,1)$ is an adjustable parameter, depending on the sensitivity of recombination frequency and available resources. The value $C_t^{k,m}$ measures the potential of the genotype G_t to create a gamete with the highest possible GEBV subject to resource and time constraints. The first dimension k reflects the constraint of probabilistic recombinations afforded by remaining resources, and the second dimension m indicates the number of founding parents that the gamete needs to collect alleles from, which would require $\lfloor \log_2 m \rfloor$ generations of breeding. Value $C_t^{k,m}$ can be obtained using the following integer linear program.

$$\max_{x,y,z} \quad C_t^{k,m} = \textstyle\sum_i \sum_c \sum_j \beta_i G_t^{i,c,j} x_{i,c,j} \tag{5.2}$$

$$\text{s.t.} \quad \textstyle\sum_j (x_{i,1,j} + x_{i,2,j}) = 1 \qquad \forall i \tag{5.3}$$

$$\textstyle\sum_i (x_{i,1,j} + x_{i,2,j}) \le L y_j \qquad \forall j \tag{5.4}$$

$$x_{i,c,j} - x_{i+1,c,j} \le z_i \qquad \forall i,c,j \tag{5.5}$$

$$\textstyle\sum_j y_j \le m \tag{5.6}$$

$$\textstyle\prod_i \left(\frac{r_i}{1-r_i}\right)^{z_i} \ge p^k \tag{5.7}$$

$$x, y, z \text{ binary} \tag{5.8}$$

Here, $x_{i,c,j}$ is a binary variable that indicates whether allele (i,c,j) is selected ($x_{i,c,j} = 1$) or not ($x_{i,c,j} = 0$) to assemble the gamete, y_j is a binary variable that indicates whether individual j is used ($y_j = 1$) or not ($y_j = 0$), and z_i is a binary variable that indicates whether there is a recombination between loci i and $i+1$ ($z_i = 1$) or not ($z_i = 0$).

The objective value (5.2) is the maximum possible GEBV of a gamete that can be assembled from the current population. Constraint (5.3) ensures selection of one chromosome for each locus in an individual to assemble the gamete. Constraint (5.4) requires that no alleles from unselected individuals can be used to assemble the gamete. Constraint (5.5) detects whether a recombination

is necessary between loci i and $i+1$. Constraint (5.6) limits the selection of at most m parents among all individuals. Finally, constraint (5.7) requires that the likelihood of necessary recombinations be no less than p^k, which equivalently limits the amount of resources needed to afford such recombination events. Take for example, for very small r_i values, the value of $\left(\frac{r_i}{1-r_i}\right)$ becomes very small, hence we need more resources (larger k) to make that recombination happen.

5.3.2.2 Action Space

The decision maker should take an action in each generation and decide the amount of resources and the selection strategy for that generation. In Moeinizade et al. (2019), we demonstrated the effectiveness of look-ahead selection (LAS) against conventional selection methods. Here, we focus on optimizing the resource allocation and use LAS algorithm to determine the selection strategy.

A common way of obtaining approximate solutions for continuous action spaces is to discretize the action space. In a discrete action space, the agent decides which distinct actions to perform from a finite action set (Masson et al., 2016). In this study, we discretize the action space. Specifically, we assume allocating b amount of resources in one generation is equal to producing b total number of progeny in that generation (making one progeny costs one unit of budget). Hence, action is a discrete value representing the number of progeny in the population.

We define the action space as time dependent set of $(b_1, b_2, ..., b_T)$ values such that $\sum_{t=1}^{T} b_t = B_0$ where b_t is the amount of resources to spend in generation t, T is the total number of generations, and B_0 is the amount of total budget.

5.3.3 Proposed Solution Technique

Figure 5.1 represents the reinforcement learning system. As shown in this figure, there are two main components in the system: agent and environment. The goal of the agent is to find the optimal policy (i.e., the optimal action to take at time t) where policy is defined as a function mapping states to the actions ($\pi : S \rightarrow A$). The environment is the breeding simulation which provides the next state to the agent. The agent evaluates the value of performing action b in state

s using a pretrained value function and then acts greedily to find the best policy. The agent then decides to take the optimal action at time $t+1$ and this process continuous until reaching the deadline.

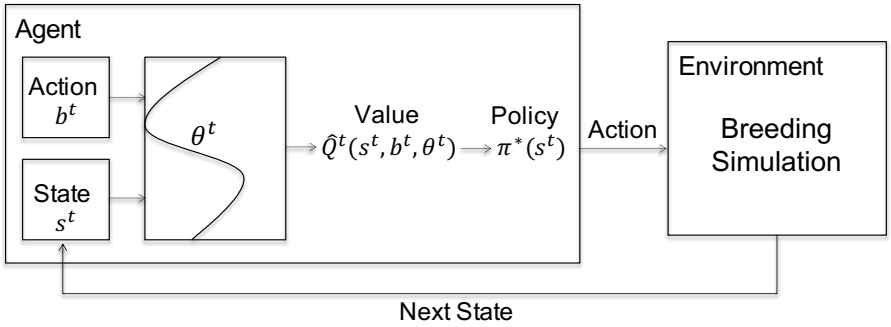

Figure 5.1 The reinforcement learning system representation. The environment is the breeding simulation which provides the next state to the agent. At each time step, the value of the current state is evaluated using a pretrained nonlinear function for a given action. Finally, the policy function determines the optimal action which is passed to the simulation.

Suppose we have a MDP defined as state-action pairs and given some policy π. First, we predict the value function by constructing the action-value function $\hat{Q}(s, b, \theta)$ to represent the objective value for a given state-action pair. Then, we can predict the value of a state given all possible actions and find the optimal policy, $\pi^*(s)$, that maximizes the action-value function. Section 5.3.3.1 describes the value function approximation technique and section 5.3.3.2 elaborates on policy improvement.

5.3.3.1 Value Function Approximation

The value function demonstrates how good each state and/or action is by calculating the expected cumulative reward in long-term. In this problem, the immediate rewards are considered to be zero and the objective is to maximize the genetic gain in final generation. Hence, the value here

represents the GEBV of the best offspring in the final generation, where GEBV of individual n is calculated as the sum of effects across the entire genome (GEBV(n)= $\sum_{l=1}^{L}\sum_{m=1}^{2} G^{l,m,n}\beta^{l}$).

The simplest way of representing a value function is by the use of a lookup table, with the values of each state-action pair stored. However, when the state-action spaces are large, storing and retrieving values become a problem, as it takes up large amounts of computational resources. To solve this problem, function approximators can also be used instead of a lookup table for representing value functions, thereby limiting the memory being used and speeding up the learning process. Therefore, to estimate the value function, $\hat{Q}(s, b, \theta)$, efficiently, we should use a function approximation method (e.g., nonlinear regression, support vector machine, decision tree based models and neural network).

The parameters (θ) need to be learned for each time period, t, separately. We employ a backward approach by optimizing resources from the final generation to the first generation. Given that the objective is to maximize the maximum GEBV in the target generation, ($g_T^{\max}$), the optimal strategy in the final generation is to allocate all the remaining budget ($b_T^* = B_{T-1}$, where B_{T-1} is the remaining resources for the final generation). To find optimal budget, b_t^*, for earlier generations $t \in \{1, ..., T-1\}$, we take advantage of simulation to learn how different budget allocation scenarios impact the final performance by generating learning data as described in algorithm 5.3.3.1. This algorithm presents data collection process for a given generation, τ, which goes backwards from $T-1$ to 1. For generation τ, we record state-action pairs, $(g_\tau^{\max}, C_\tau, B_\tau)$ and b_τ, and the objective value, $g_T^{\max}$. Then, we estimate the value function to map state-action pairs to the objective value. We first define the three functions used in algorithm 5.3.3.1 for data generation and then discuss the value function approximation technique.

Definition 1 *The selection function is defined as follows:* $[S] = \texttt{Select}(G_{t-1}, r, n, b_t)$. *The input parameters are the population genotype at generation* $t-1$, $G_{t-1} \in \mathbb{B}^{L\times 2\times N}$, *recombination frequency vector,* $r \in [0, 0.5]^{N-1}$, *the number of crosses* n, *and amount of resources for generation* t, b_t. *Note*

that the resources correspond to the progeny population size. The output $S = \begin{bmatrix} s_{11} & s_{12} & b_t^1 \\ s_{21} & s_{22} & b_t^2 \\ & \dots & \\ s_{n1} & s_{n2} & b_t^n \end{bmatrix}$ *contains the indices of selected parents in the breeding population and the number of progeny produced from each cross (e.g.,* s_{11} *is crossed with* s_{12} *to produce* b_t^1 *progeny and* s_{21} *is crossed with* s_{22} *to produce* b_t^2 *progeny, where* $\sum_{i=1}^{n} b_t^i = b_t$*).*

Definition 2 *The reproduction function is defined as follows:* $[G_t] = \texttt{Reproduce}(G_{t-1}, S, r)$*. The input parameters are the population genotype at generation* $t-1$*,* $G^{t-1 \in \mathbb{B}^{L \times 2 \times N}}$*, selection matrix,* S*, and the recombination frequency vector,* $r \in [0, 0.5]^{N-1}$*. The output is the genotype of the progeny population. The genetic information are inherited from parents to progeny according to the inheritance distribution defined in Han et al. (2017).*

Definition 3 *The action function is defined as follows:* $[b_t] = \texttt{Action}(t, T, B_{t-1}, A)$*. The input parameters are the current generation,* t*, total number of generations,* T*, the available resources to be spent in generations* t *to the end,* B_{t-1}*, and possible set of actions,* A*. The output is resources or progeny size for generation* t*. We choose an action randomly from a finite set of values* $\tilde{a} \in A$*,* $A = \{a_1, a_2, ..., a_k\}$ *such that* $\frac{\sum_{i=1}^{k} a_i}{k} = \frac{B_0}{T}$*. We produce at least* α *progeny for each generation (*$\alpha = \min_{i=1}^{k} a_i$*). Therefore the output,* b_t*, can be calculated as follows:* $b_t = min(\tilde{a}, B_{t-1} - \alpha \times (T - t))$*.*

Generating learning/training data in complex stochastic environments can be time consuming. Neural networks usually need more training data and thus are not the best approach here since the learning process is considerably time-consuming. After exploring three function approximotors including generalized additive model (gam), support vector machine (SVM), and random forest (RF), we decided to choose random forest considering both efficiency and computational time. The inputs to the random forest model are the data generated using algorithm 5.3.3.1 with $km+3$ features including the maximum current GEBV ($g_t^{\max}$), the potential matrix (C_t), remaining budget (B_{t-1}), and action (b_t) where k and m are the dimensions of the potential matrix. The output is the GEBV of best individual in the final generation, $g_T^{\max}$.

Algorithm 1 Learning data generation

Start with initial population G_0 and total budget B_0
for $t := 1$ to $\tau - 1$ **do**
  $b_t = \texttt{Action}(t, T, B_{t-1}, A)$
  $[S] = \texttt{Select}(G_{t-1}, r, n, b_t)$
  $[G_t] = \texttt{Reproduce}(G_{t-1}, S, r)$
end for
for $b_\tau \in A$ **do**
  for $t := \tau$ to T **do**
    if $t = \tau$ **then**
      Record $(g_\tau^{\max}, C_\tau, B_\tau)$ and b_τ
    else
      if $t = T$ **then**
        $b_t = B_{T-1}$
      else
        $b_t = \texttt{argmax}_{b \in A} \hat{Q}_t(s, b, \theta)$
      end if
    end if
    $[S] = \texttt{Select}(G_{t-1}, r, n, b_t)$
    $[G_t] = \texttt{Reproduce}(G_{t-1}, S, r)$
    Record $g_T^{\max}$
  end for
end for

5.3.3.2 Greedy Policy Improvement

The ultimate goal of the agent is to find an optimal policy π^* that maximizes the value function. After learning the value function, we employ a greedy approach to improve the policy by selecting the action with the highest estimated value in each state. Let $\hat{Q}_t(s, b, \theta)$ represent the approximated value function for each generation except final. We can calculate the optimal policy for all generations from 1 to $T - 1$ as follows:

$$\pi_t^*(s) = \text{argmax}_{b \in A} \hat{Q}_t(s, b, \theta), \ \forall t \in \{1, 2, ..., T-1\} \tag{5.9}$$

Moreover, the optimal policy in the final generation is to allocate all the remaining budget. Therefore $\pi_T^*(s) = B_{T-1}$, where B_{T-1} presents the remaining resources for the final generation.

5.4 Results

5.4.1 Simulation Settings

The genotypic data, marker effects and recombination rates are based on Moeinizade et al. (2019). The genotypic data contains genotypes of 369 maize inbred lines consisting of $L = 1.4$M SNPs distributed across ten maize chromosomes. To reduce the dimension, we define haplotype blocks. The resulted data has $L = 10,000$ markers.

Let's assume there exist 5 total generations of breeding ($T = 5$) and the amount of total budget is 1000. We consider seven possible action values as follows: $A = \{50, 100, 150, 200, 250, 300, 350\}$. Note that the amount of budget in each generation indicates the total number of progeny produced in that generation. Additionally, we consider that no more than 10 crosses are made in each generation.

Figure 5.2 demonstrates the simulation flowchart. We start with the initial population by randomly choosing 200 individuals out of 369. The state is observed by calculating the tuple $(g_t^{\max}, C_t, B_{t-1})$. The C_t matrix is generated by solving the optimization problem presented in (5.2)-(5.8). Here we choose $k = 5$, and $m = 5$. These are parameters and can be changed according to the data and time required for solving the optimization problem. Figure 5.3 shows the heat map for one sample C_t in a simulation. The x-dimension of this plot is representing the possibility of combining more alleles from multiple individuals in case of having more time. Moreover, the y-dimension is representing the possibility of allowing more recombination to happen in case of having more resources. As expected, the performance becomes better towards the right and bottom of the plot by considering more time and resources. In addition, this C matrix indicates that given the current generation, potential genetic gain is more sensitive to resources than to time constraint, which is helpful for the reinforcement learning algorithm to make resource allocation decisions.

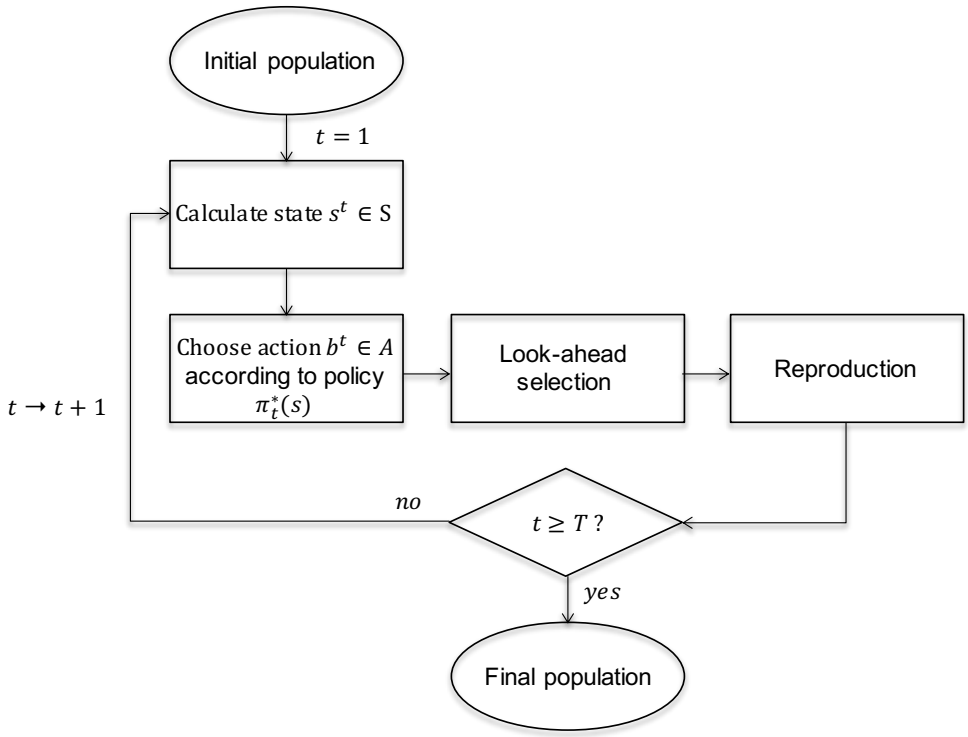

Figure 5.2 The simulation flowchart. The process starts with the initial population and goes through resource allocation, selection and reproduction steps until getting to the deadline. To find the optimal resources, we first calculate the current state and then take the action with highest value according to the optimal policy. The action represents the number of progeny to be produced for that generation.

Next, the optimal policy will be calculated using the current generation action-value function in a greedy approach. Then, candidates are selected according to look-ahead selection as parents to produce next generation. This continuous till reaching the deadline. Finally, we evaluate the performance based on the GEBV of individuals in the final population.

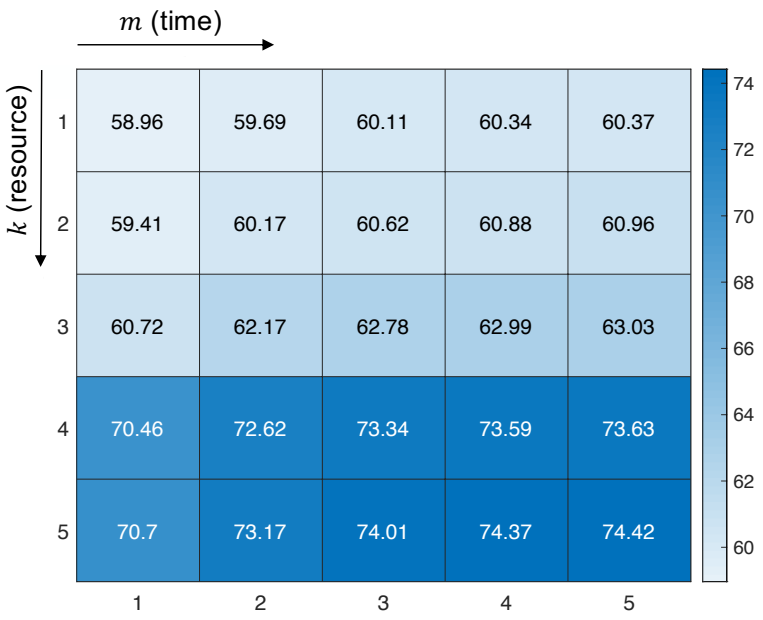

Figure 5.3 Heat map for one sample C matrix where $k = 5$, and $m = 5$. Each square demonstrates the best achieved GEBV value considering different levels of time and resources. The bottom right square has the highest potential GEBV value since it considers having the most time and resources.

5.4.2 Simulation Results

To approximate the action-value function, we first generated learning data including state-action pairs using simulation and then trained a random forest algorithm for each generation separately to estimate the objective value. The size of training observations that were generated in the simulation vary between 1,500 to 6,000 for each generation, and there were a total 28 predictors including the action (b), maximum current GEBV ($g^{\max}$), remaining budget (B) and 25 values from the potential matrix, C. For training the random forest models, we did a search grid over three parameters including the number of selected features, minimum leaf node size, and maximum number of splits. The set of parameters with the least out of bag error were selected. The out of bag mean square errors for the first generation until the fourth generation are 2.39, 2.41, 2.31, 2.25, respectively.

We compared the optimal resource allocation strategy with the even allocation strategy (i.e., allocating resources equally across all generations). Three hundred independent simulations were conducted for each strategy using MATLAB (R2021-a).

Figure 5.4 (A) demonstrates the cumulative distribution functions (CDFs) of the population maximum in the final generation. The performance becomes better as the CDF moves towards the right direction. Take for example, point (60, 92) means 92% of the simulations achieved maximum GEBV less than or equal to 60. As demonstrated in this figure, the optimal allocation strategy outperforms even allocation strategy in almost all percentiles. Although, this improvement is not by a high margin, but it is considerable given that the improvement is across almost all percentiles for 5 generations of breeding. More improvements can be achieved for longer-term breeding. Moreover, if we compare the average performance of top 50 individuals instead of top 1, we can see a wider gap between the two curves as shown in Figure 5.4 (B).

So far, we have observed the improvements of our proposed optimal allocation strategy with respect to the even allocation strategy. Thus, the question arises: what is the behavior of the optimal allocation strategy and why that behavior results in improvements? To understand this better, we examined the histograms of resource allocation among 5 generations for the optimal strategy. As illustrated in Figure 5.5, three different behaviors are observed. In the first generation, the optimal strategy tends to be around the average, in the middle generations, there is more tendency towards spending less resources, and in the final generation almost half of the total budget is spent.

In this case study, we demonstrated the effectiveness of our proposed method against evenly allocating resources across breeding cycles. Our optimal strategy suggests investing in the first generation to increase genetic diversity and then spending moderate amount of resources in the middle generations and finally investing more in the final generation to exploit the best performance that can be achieved.

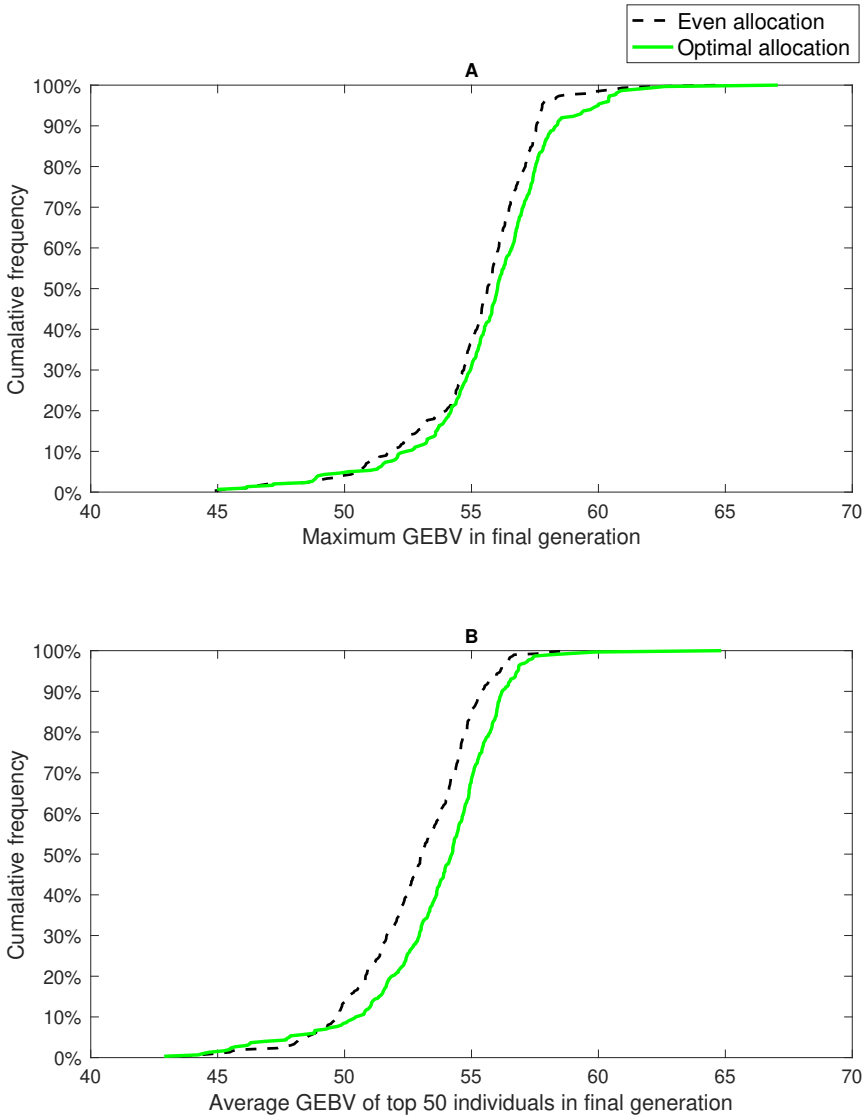

Figure 5.4 Cumulative distribution functions of population maximum in the final generation (A) and average performance among top 50 individuals in the final generation (B) for two strategies of resource allocations among 300 independent simulations. The black dashed curve represents the even allocation strategy and the green curve represents the optimal allocation strategy.

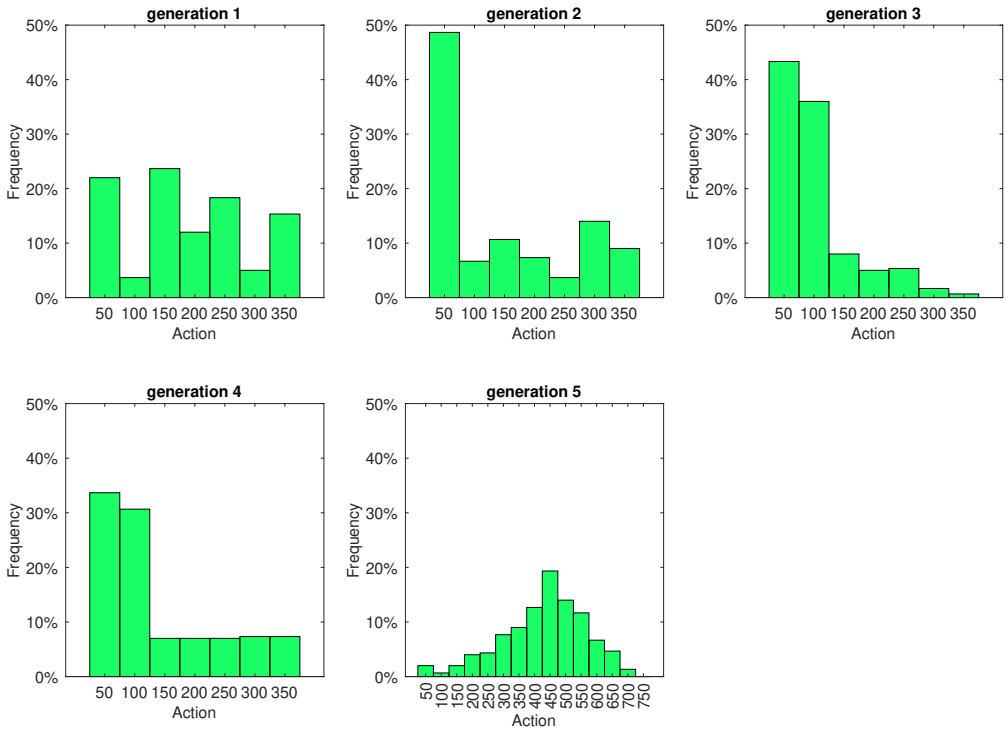

Figure 5.5 Histograms of resource allocation across 5 generations for the optimal resource allocation strategy. The amount of resources that can be spent in all generations till one before final is chosen from a predefined set of actions. Here, we have seven different possibilities (50, 100, 150, 200, 250, 300, 350) for generations one till four and the remaining budget will be spent in the final generation. Note that for the even allocation strategy the action is deterministic which is $b = 200$ for all generations.

5.5 Conclusion

This study provides a framework to find the optimal resources that should be allocated throughout different generations in a breeding program by integrating the recently proposed look-ahead selection algorithm for genomic selection and reinforcement learning techniques. Look-ahead selec-

tion is capable of estimating the consequences of selection and mating decisions under uncertain recombination events. Reinforcement learning is able to balance the trade-off between cost and time but its performance is sensitive to the definitions and dimensions of the state and action spaces. Therefore, look-ahead selection is integrated into the reinforcement learning framework to optimize resources in addition to the selection and mating steps and new solution techniques are proposed to battle the curse of dimensionality.

We considered MDPs with very large and continuous state spaces, and we used random forest to construct an approximate function to store the value functions used by the algorithms. We implemented a greedy policy improvement to learn optimal policy in a backward manner. Numerical results suggested the improvement of the proposed optimal allocation strategy versus even allocation strategy.

The RL framework presented in this work has three major contributions. The first contribution is the definition of the state space. It is analytically and computationally challenging to simplify the state space definition for a large scale stochastic environment. To avoid the explosion of state space, we propose an integer linear program that captures the genomic information of the population by considering the trade-offs between time and resources. The second one is integrating the look-ahead selection and reinforcement learning. Given the optimal allocation strategy, look-ahead selection further improves the genetic gain by optimizing the selection and mating steps. The third contribution is the learning process which is performed in a backward manner. We benefit from the structure of the genomic selection problem and assume we know the best policy in the target generation (spending all remaining budget). Then, we approximate the value function from the last generation to the first one and use it to improve the policy in a greedy way.

Future research is needed to address the limitations of this study. First, the current paper considers discrete action spaces with predefined values. Future research should consider continuous action spaces and investigate algorithms to optimize policy in such spaces. Second, deep neural networks can be used for function approximation if we generate more learning data by making the simulation more efficient. Finally, the case study presented here is for a single data set from a

single crop organism. Future research considering more species is necessary to demonstrate the generalization of our proposed method.

5.6 References

Arulkumaran, K., Deisenroth, M. P., Brundage, M., and Bharath, A. A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38.

Bellman, R. (1966). Dynamic programming. *Science*, 153(3731):34–37.

Chapman, D. and Kaelbling, L. P. (1991). Input generalization in delayed reinforcement learning: An algorithm and performance comparisons. In *IJCAI*, volume 91, pages 726–731. Citeseer.

Chen, L., Li, C., Sargolzaei, M., and Schenkel, F. (2014). Impact of genotype imputation on the performance of gblup and bayesian methods for genomic prediction. *PLoS One*, 9(7):e101544.

Covarrubias-Pazaran, G., Schlautman, B., Diaz-Garcia, L., Grygleski, E., Polashock, J., Johnson-Cicalese, J., Vorsa, N., Iorizzo, M., and Zalapa, J. (2018). Multivariate gblup improves accuracy of genomic selection for yield and fruit weight in biparental populations of vaccinium macrocarpon ait. *Frontiers in Plant Science*, 9:1310.

Crossa, J., Pérez-Rodríguez, P., Cuevas, J., Montesinos-López, O., Jarquín, D., de los Campos, G., Burgueño, J., González-Camacho, J. M., Pérez-Elizalde, S., Beyene, Y., et al. (2017). Genomic selection in plant breeding: methods, models, and perspectives. *Trends in Plant Science*, 22(11):961–975.

Dayan, P. and Niv, Y. (2008). Reinforcement learning: the good, the bad and the ugly. *Current Opinion in Neurobiology*, 18(2):185–196.

Dong, L., Xiao, S., Wang, Q., and Wang, Z. (2016). Comparative analysis of the gblup, embayesb, and gwas algorithms to predict genetic values in large yellow croaker (larimichthys crocea). *BMC Genomics*, 17(1):460.

Friedman, J., Hastie, T., Tibshirani, R., et al. (2001). *The Elements of Statistical Learning*, volume 1. Springer series in statistics New York.

Gosavi, A. (2009). Reinforcement learning: A tutorial survey and recent advances. *INFORMS Journal on Computing*, 21(2):178–192.

Han, Y., Cameron, J. N., Wang, L., and Beavis, W. D. (2017). The predicted cross value for genetic introgression of multiple alleles. *Genetics*, 205(4):1409–1423.

Hausknecht, M. and Stone, P. (2015). Deep recurrent q-learning for partially observable mdps. *arXiv preprint arXiv:1507.06527*.

Heess, N., TB, D., Sriram, S., Lemmon, J., Merel, J., Wayne, G., Tassa, Y., Erez, T., Wang, Z., Eslami, S., et al. (2017). Emergence of locomotion behaviours in rich environments. *arXiv preprint arXiv:1707.02286*.

Howe, A. and Pyeatt, L. (1998). Decision tree function approximation in reinforcement learning. Technical report, Colorado State University.

Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Li, H., Wang, J., and Bao, Z. (2015). A novel genomic selection method combining gblup and lasso. *Genetica*, 143(3):299–304.

Liu, Y., Lu, S., Liu, F., Shao, C., Zhou, Q., Wang, N., Li, Y., Yang, Y., Zhang, Y., Sun, H., et al. (2018). Genomic selection using bayescπ and gblup for resistance against edwardsiella tarda in japanese flounder (paralichthys olivaceus). *Marine Biotechnology*, 20(5):559–565.

Liu, Y. and Wang, D. (2017). Application of deep learning in genomic selection. In *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 2280–2280. IEEE Computer Society.

Lobo, I. and Shaw, K. (2008). Thomas hunt morgan, genetic recombination and gene mapping. *Nature Education*.

Masson, W., Ranchod, P., and Konidaris, G. (2016). Reinforcement learning with parameterized actions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

Meuwissen, T. H. E., Hayes, B. J., and Goddard, M. E. (2001). Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157(4):1819–1829.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Moeinizade, S., Hu, G., Wang, L., and Schnable, P. S. (2019). Optimizing selection and mating in genomic selection with a look-ahead approach: An operations research framework. *G3: Genes, Genomes, Genetics*, 9(7):2123–2133.

Montesinos-López, O. A., Montesinos-López, A., Crossa, J., Gianola, D., Hernández-Suárez, C. M., and Martín-Vallejo, J. (2018). Multi-trait, multi-environment deep learning modeling for genomic-enabled prediction of plant traits. *G3: Genes, Genomes, Genetics*, 8(12):3829–3840.

Neves, H. H., Carvalheiro, R., and Queiroz, S. A. (2012). A comparison of statistical methods for genomic selection in a mice population. *BMC Genetics*, 13(1):100.

Pryce, J., Gredler, B., Bolormaa, S., Bowman, P., Egger-Danner, C., Fuerst, C., Emmerling, R., Sölkner, J., Goddard, M., and Hayes, B. (2011). Genomic selection using a multi-breed, across-country reference population. *Journal of Dairy Science*, 94(5):2625–2630.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. (2015a). Trust region policy optimization. In *International Conference on Machine Learning*, pages 1889–1897.

Schulman, J., Moritz, P., Levine, S., Jordan, M., and Abbeel, P. (2015b). High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Sutton, R. S., McAllester, D. A., Singh, S. P., Mansour, Y., et al. (1999). Policy gradient methods for reinforcement learning with function approximation. In *NIPs*, volume 99, pages 1057–1063. Citeseer.

Szepesvári, C. (2010). Algorithms for reinforcement learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 4(1):1–103.

Tucker, G., Bhupatiraju, S., Gu, S., Turner, R. E., Ghahramani, Z., and Levine, S. (2018). The mirage of action-dependent baselines in reinforcement learning. *arXiv preprint arXiv:1802.10031*.

Van Hasselt, H., Guez, A., and Silver, D. (2015). Deep reinforcement learning with double q-learning. *arXiv preprint arXiv:1509.06461*.

Wang, Z., Schaul, T., Hessel, M., Hasselt, H., Lanctot, M., and Freitas, N. (2016). Dueling network architectures for deep reinforcement learning. In *International Conference on Machine Learning*, pages 1995–2003.

Watkins, C. J. and Dayan, P. (1992). Q-learning. *Machine Learning*, 8(3-4):279–292.

CHAPTER 6. GENERAL CONCLUSIONS

This dissertation proposes simulation-based optimization techniques to identify optimal solutions for sequential decision making problems in plant breeding programs. During the course of a breeding program, several decisions should be made in each generation and all crucial and complex factors that affect crop genetic improvement should be taken into account explicitly. These crucial and complex factors include recombination frequencies, selection and mating strategy, deadline constraints and resource limitations. In this study, we design and develop advanced optimization techniques to address these challenges.

The first paper proposes look-ahead selection (LAS) method to identify individuals within the population that should be selected and crossed as breeding parents to produce the next generation of individuals when incorporating genomic information. Previous selection methods try to maximize the genetic achievement of breeding parents or the best possible progeny without considering time. However, LAS aims at the right objective, by maximizing the expected GEBV of the best offspring in the terminal generation given a limited amount of resources. In this study, we demonstrated the performance of LAS against conventional selection methods using empirical data from a population of maize inbred lines. LAS outperformed previous approaches by achieving more genetic gain and preserving more genetic diversity over the course of a simulated breeding program. LAS has three major contributions. First, LAS is deadline sensitive and makes a trade-off between achieving short-term genetic gains and maintaining genetic diversity long-term by taking time into account. Second, unlike previous methods, LAS optimizes mating strategies by selecting individuals into pairs of breeding parents to achieve further genetic gains. Third, LAS uses a heuristic to increase genetic diversity by producing more progeny from parents with higher potential instead of producing same number of progeny from each cross.

The second paper develops the multi-trait look-ahead selection (MT-LAS) approach to balance trade-offs among multiple traits. This work builds upon the previous study by extending the LAS algorithm to multiple traits such as yield, grain quality, and disease resistance. We focus on maximizing the main trait of interest while straining other traits to fall within flexibly defined ranges. Simulations were designed to compare MT-LAS with the conventional index selection in a case study using realistic data. Index selection method selects individuals based on an index which is defined as a linear combination of different traits where weights are assigned to those traits relative to their economic importance. Simulation results suggest that MT-LAS is more effective at balancing multiple traits compared with index selection. The main contribution of this algorithm is to dynamically adjust the objective function according to the progress of the current population by placing more emphasis on feasibility requirements when most of the individuals are not predicted to fall within the bounds for the bounded traits and focusing on the main trait when most of the individuals become acceptable for the bounded trait. Another contribution of this method is its interpretibility. Defining bounds instead of weights provides a more intuitive description of the objectives considering the original measurement scales of traits.

The third paper introduces the look-ahead Monte Carlo selection (LMC) approach to convert multiple desirable traits from a donor to a recipient cultivar through backcrossing and selfing schemes in a trait introgression program. The underlying concept is to use repeated random sampling for modeling uncertainty involved due to recombination events. Best individuals are then selected based on their estimated performance in the final generation to go through backcrossing until a certain background recovery is achieved and then selfing to make homozygous individuals. We compared the proposed method with two existing approaches in two scenarios of resources using three different data sets with different genetic similarities between the recipient and donor considering practical aspects in trait introgression programs. Computational results demonstrated the improvements of the LMC method over two existing selection approaches in terms of the probability of success after three backcross generations followed by selfing.

Finally, to optimize resource allocation decisions in coordination with parental selection decisions in breeding programs, the last paper integrates the look-ahead selection algorithm in a reinforcement learning framework. First, we mathematically formulate the resource allocation problem in the context of Markov Decision Process by defining state and action spaces. To battle the curse of dimensionality, we simplify the definition of state space by defining a potential matrix which measures the general combining ability of the population to create a gamete with highest possible genetic value. We propose a linear integer program to obtain the potential matrix by making a trade-off between resources and time. To store the value functions, we propose a nonlinear function approximation technique and then implement a greedy policy improvement to learn optimal policy in a backward manner. We designed a case study to compare our approach against even allocation of resources. Numerical results suggest the improvement of the proposed optimal allocation strategy versus even allocation strategy.

For future research, there exist several interesting directions. First, the look-ahead selection relates the objective to the final generation and future research is needed to focus on designing new selection methods that also consider intermediate generations in the objective. Second, we assume the genomic prediction is accurate and if there is any error associated with marker effect estimates that have an equal effect on all selection methods. Future research should address the effect of genomic prediction error on selection accuracy. Third, when designing case studies with realistic data sets, we based our simulations on a single crop organism. Further simulations considering more diverse populations are necessary to demonstrate the general applicability of the proposed methods. Moreover, in the the trait introgression study, we investigated only backcrossing schemes. Future work should consider inter crosses between backcross populations. Finally, in the resource allocation study, we made some simplifying assumptions such as discretizing the action space. Future research should consider continuous action spaces and investigate algorithms to optimize policy in such spaces.