Thesis Report

Generating Evenly Distributed Near-Optimal Investment Alternatives for Large-Scale Power Systems using Genetic Algorithms

Max Le Blansch

Thesis Report

Generating Evenly Distributed Near-Optimal Investment Alternatives for Large-Scale Power Systems using Genetic Algorithms

by

Max Le Blansch (4965329)

Cover image by KOBU Agency on Unsplash

TU Delft

Abstract

This work proposes a new Modelling-to-Generate Alternatives (MGA) method for Energy System Optimisation Models (ESOMs) using a Genetic Algorithm (GA). Instead of generating each alternative one by one, the GA aims to optimise for a diverse set of alternatives, meaning they cover the space of possible alternatives as evenly as possible. Such a diverse set of alternatives has the potential to improve the decision-making process by accelerating the extraction of stakeholder requirements and finding more agreeable compromises. Before designing the algorithm, we investigate what diversity metric is most suitable to optimise. The components of the GA are designed to exploit useful properties of ESOMs to increase efficiency. The performance of the GA is tested in terms of output quality and scalability for increasingly large ESOMs, showing promising performance in terms of output quality for a similar computational burden as state-of-the-art MGA methods. A potential issue caused by the curse of dimensionality is formulated, requiring further investigation on its impact on the quality of the method's output. We show the generated output of applying the proposed method to the European power system, which encourages further testing of the method on increasingly large ESOMs.

Contents

List of Figures

List of Tables

List of Symbols & Abbreviations

Symbol	Description
$f(\mathbf{x})$	Minimum cost objective function of the ESOM.
$\mathbf{c}$	Coefficient vector of ESOM objective function $f(\mathbf{x})$.
$d(\mathbf{x},\mathbf{y})$	Distance between vector $\mathbf{x}$ and $\mathbf{y}$
ε	The allowed cost slack of the alternatives as a fraction of the minimum cost
S_ε	The near-optimal space
v	Number of investment variables
$\mathbf{i}$	v-dimensional individual
$\mathbf{p}$	v-dimensional parent individual
$\mathbf{o}$	v-dimensional offspring individual
$\mathbf{d}$	Direction vector

Abbreviation	Meaning
ESOM(s)	Energy System Optimisation Model(s)
MGA	Modelling to Generate Alternatives
GA(s)	Genetic Algorithm(s)
LP(s)	Linear Program(s)
MILP(s)	Mixed-Integer Linear Program(s)

1 Introduction

Index terms– Genetic Algorithms, Expansion planning, Modelling-to-Generate Alternatives, Near-optimal investments

1.1. Background and motivation

To aid in the energy transition, sectors with high energy demand — such as transport, production, and heating — are becoming increasingly electrified. As this demand is only expected to increase, it is vital to invest in the energy system's generation, storage and transmission capacities. The untimely development of these projects due to a slow decision-making process can result in network congestion or generation inadequacy, slowing down the electrification of high-polluting sectors and the energy transition itself. Deciding what investments to make in the energy generation and transmission infrastructure to meet the demand with minimal total cost is called an expansion planning problem. One of the factors complicating the process is the non-transparent nature of the preferences of the potentially many stakeholders, making it harder to find compromises considering these preferences [21].

Energy System Optimisation Models (ESOMs) help in this process to optimise for a minimum cost investment plan. They model the technical constraints of the energy infrastructure using historic and/or forecasted data on renewable resource availability, energy demands and network topology. However, as the optimised investment plan only minimises the system cost, alternative plans with a slight increase in cost might be preferred for some objectives not included in the ESOM. For example, public acceptance of large infrastructure investment, ease of implementation and international inequality.

In an iterative decision-making process, the minimum-cost investment plan then helps to partially identify stakeholder preferences in terms of secondary objectives, after which the ESOM can be updated according to those preferences to obtain an alternative investment plan. This iterative process then repeats until an agreeable alternative is found for all stakeholders.

Instead of manually updating the ESOM to obtain new alternative investment plans, the Modelling-to-Generate Alternatives (MGA) method can help by automatically generating alternative investment plans without manually modifying the ESOM. All the generated alternatives lie within a predefined cost range of the minimum cost investment plan, making them near-optimal alternatives in terms of cost. These alternative plans potentially aid in uncovering stakeholders' preferences and finding compromises, also considering unmodelled objectives. Also, analysis of alternative plans can identify common and absent characteristics, resulting in *must haves* and *must avoids* [14]. We want the generated alternatives to evenly cover the near-optimal space S_ε of possible alternatives, serving as realistic options and examples of compromises between technologies and investment decisions [27].

1.2. Research gap

There are many different MGA methods, most of which modify the ESOM by adding a constraint, after which the objective function is modified to find different alternatives every time the ESOM is solved. Each of these methods differs in how the objective function is modified after finding each alternative [25]. Since optimising the ESOM always results in extremes between investment decisions, the usefulness of the alternatives found by the algorithm is limited. Other MGA approaches that attempt to find these non-extreme alternatives only

support a relatively low number of investment variables [27]. Because of this, their applicability is limited to ESOMs of minimal size or using aggregated variables — such as total capacity expansion for each technology — for bigger ESOMs.

Often, these methods employ heuristics to determine the coefficients of the objective function, attempting to find alternatives that are maximally different. Instead, directly optimising the even coverage of S_ε by the generated alternatives, covering extreme regions and regions between them. The field of diversity optimisation aims to achieve this even coverage, often through the employment of Genetic Algorithm (GA) approaches [41].

This work proposes a novel Genetic Algorithm (GA) as an MGA method that focuses on finding alternatives that evenly cover the space of possible alternatives and includes non-extreme alternatives. Its novelty compared to other MGA methods, which are based on a GA, lies in the optimisation of even coverage of a subset of the decision variables and its operators being tailored to exploit the mathematical properties of ESOMs to improve efficiency. When compared to standard GAs [43], the GA will differ by optimising a measure of even coverage of a set of solutions, in contrast to optimising an objective value defined for a single solution. For each solution, only the investment decision variables are considered in the GA, using the ESOM to optimise the operational variables given the investment decision variables to calculate the cost. Compared to other MGA methods, our approach will not be limited in the number of included investment variables while generating both extreme and non-extreme alternatives.

As the GA is tailored to ESOMs and only needs to optimise the ESOM for the operational variables, the GA can generate at least as many alternatives as the state-of-the-art MGA method with the same computational budget. When also considering all the alternatives the algorithm found throughout the optimisation process, it can even generate more alternatives. However, the set of all alternatives found throughout the search is not optimised to evenly cover S_ε.

The performance of the proposed GA is tested by comparison with the Modelling-All-Alternatives [27] method in terms of output quality, and its scalability is contrasted against a weighted sum MGA method [25]. Finally, to showcase the relevance of our contribution, our approach will be applied to the Europe-wide model used in the SECURES project [39].

1.3. Report structure

The rest of the report is structured as follows. Chapter 2 will explain ESOMs and MGA methods, explaining their mathematical definition and the context in which they are used. Also, the chapter explains the basic concepts of GAs and the challenges when designing such algorithms. Next, Chapter 3 will discuss the state-of-the-art MGA methods and identify their gaps for improvement. Then, it highlights relevant work in the field of diversity optimisation and other GA approaches for MGA methods to establish the intended contribution of this work. Chapter 4 explains the design process of the proposed GA, first formulating the optimisation model that the GA is solving, followed by the explanation of how the most suitable metric to measure diversity is chosen, after which potential challenges of using the GA with a high number of investment variables are identified. Consequently, all components of the GA are specified, which leads to the experiment definition and experiment hypotheses. The experiments are designed to (1) test the GA's performance on an artificial and real-life ESOM. (2) the result quality and computational cost compared to other MGA methods. Chapter 5 sets out the results of the experiments and establishes insights by reflecting on the hypotheses with the derived outcomes. Chapter 6 reflects on the research, discussing unexpected results and stating potential future work. Finally, Chapter 7 concludes the work, summarising the most relevant decisions and findings.

2

Theoretical background

Before highlighting relevant works in literature, this chapter provides the appropriate background knowledge. First, we discuss what Energy System Optimisation Models (ESOMs) are, the ESOM used in this work and the motivation for finding alternative solutions. Then, we formulate the optimisation problem of the MGA method and how it is generally solved. Finally, we explain general information regarding GAs and their relevant modelling challenges.

2.1. Energy System Optimisation Models

The problem of deciding what investments to make in the energy infrastructure to meet the demand with minimal total cost is called an expansion planning problem. Such problems can be formulated as a Linear Program (LP) or Mixed-Integer Linear Program (MILP) to find the optimal investments concerning a predefined objective. The resulting ESOMs aim to determine capacities at given locations of generation, transmission, and distribution to satisfy the projected power demand within a set of technical and economic constraints [6, 13] over multiple years, and play a crucial role in planning energy transition pathways and understanding their impacts [4]. Decision variables include invested generation, storage and transmission capacities and their operation at each timestamp in the considered period.

Generally, ESOMs are defined as shown in Equations 2.1–2.3. Here, $f(\mathbf{x})$ is the objective function — most generally signifying cost — of a solution $\mathbf{x}$ and the cost coefficients $\mathbf{c}$. The solution $\mathbf{x}$ consists of investment decisions and values for the operational variables that need to satisfy the constraints of Equations 2.2 and 2.3, which include energy demand, unit commitment and other technical requirements on the system.

$$\min f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x} \tag{2.1}$$

$$s.t. \quad C\mathbf{x} \leq \mathbf{d} \tag{2.2}$$

$$A\mathbf{x} = \mathbf{b} \tag{2.3}$$

2.1.1. Limitations of ESOMs in decision-making

The investment plans output by the model serve as a basis for decision-making. More specifically, the output can either be a single plan optimising a single objective — such as cost — or multiple plans resulting from a Pareto-front in a multi-objective setting — e.g. minimising cost while also minimising emissions [6]. These investment plans are often accompanied by a sensitivity analysis, giving insight into the uncertainty of the output based on uncertainty in the input data — e.g. weather data — and model design — e.g. temporal resolution [34]. Such an analysis helps reduce the technical knowledge gap between stakeholders and modellers, improving the stakeholders' trust in the validity of the result.

Besides, the stakeholders might also have requirements which are not directly included in or optimised by the ESOM. For instance, while a minimum-cost investment plan may be economically optimal, it may fall short in terms of social acceptance or national self-sufficiency, resulting in a plan that does not fully satisfy all stakeholders. Therefore, such a single solution could partially identify stakeholder requirements, which can be used to modify the ESOM such that more agreeable plans can be found. Each new alternative plan resulting from updating the model can serve as a compromise between stakeholders' requirements,

potentially improving on some requirements with an acceptable cost increase. The reviewing process can then be repeated until a plan is found that satisfies all stakeholders. However, such a process can be very slow, especially when many stakeholders are involved — e.g. having one stakeholder for each country in Europe — and having to manually update the model to obtain alternative plans.

Instead of only the minimum cost investment plan, knowing multiple alternative investment plans with an acceptable increase in cost would enable stakeholders to then analyse the alternative plans based on their requirements and order them according to preference [21]. This more informed basis of decision-making could reduce the number of iterations needed to come to a compromise. However, when these alternative plans are in-diverse, promising compromises may be poorly represented, which can prolong the process of identifying an agreeable plan to all stakeholders. The problem of generating these alternative plans given an ESOM is what MGA aims to solve, which will be detailed in Section 2.2.

2.1.2. The SECURES project ESOM

This work will apply the proposed approach to the ESOM used in the SECURES project [39]. This ESOM is used to optimise the investment plans for different weather and climate change scenarios, highlighting challenges and opportunities for the energy system of tomorrow to ensure a reliable, sustainable and cost-efficient power supply under climate change.

Their work analyses each scenario with reference years 2030 and 2050, where 2030 only considers the 'typical' years. The analysis is done with an hourly resolution and spatial resolution of one node per country in the EU, resulting in a total of 27 nodes, as can be seen in Figure 2.1.

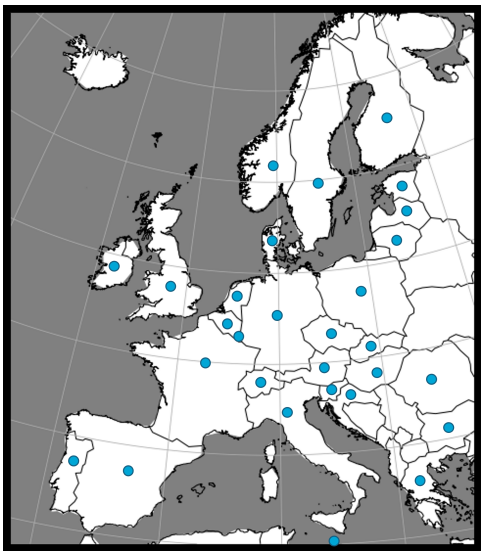


Figure 2.1: Map of countries included in the SECURES model.

The SECURES model is implemented in IESopt, an easy-to-use modelling tool used to define the components of an energy system on a high level, without compromising on performance [38].

2.2. Modelling to Generate Alternatives

Even though secondary objectives like reduced emissions and social acceptance cannot always be directly included in the model, policymakers and investors still need to incorporate them in the decision-making process. Therefore, the optimal investment decision resulting from the ESOMs, which disregards these objectives, is of little aid in the decision-making process. Rather, it represents a theoretical least-cost scenario optimised according to a modelled version of reality.

One way to mitigate this is to analyse the near-optimal space S_ε to allow human decision-makers to pick a better alternative resulting from a trade-off between a higher total cost while improving other factors not directly optimised in the model. Additionally, these alternative solutions can help identify must-haves and must-avoids as investment decisions that are part of all or no alternative solutions, respectively.

The technique to generate solutions within S_ε is called Modelling to Generate Alternatives (MGA), which

uses the optimal solution as an anchor point to explore the surrounding decision space for maximally different solutions [10].

.

2.3. Genetic Algorithms

The relevant basic principles of Genetic Algorithms (GAs) are explained in this section. For extensive details, we refer to the work of Watanabe et al. [43].

Most generally, GAs serve as heuristic optimisation algorithms. Compared to conventional, exact optimisation like (MI)LPs or gradient-based methods, they require less problem-specific knowledge while still achieving satisfactory results. In their simplest forms, they only require knowing the potential solution space — either continuous, discrete or a mix — and a way to evaluate how 'good' a solution to a problem is, also known as its fitness. This flexibility comes at a cost of guarantees on the quality of the resulting solution and the solving time. The problems where GAs are favoured over exact optimisation methods are often computationally intractable — e.g. NP-Hard problems — or lack gradient information. The quality of problems without gradient information is often not defined in equations but relies on simulation results.

GAs come in many variants, which, in essence, all keep track of a set number of solutions, called a population of individuals. The algorithm iteratively updates the population by switching between the variation and selection phases, where the updated population is called the next generation. During the variation phase, the algorithm generates new, potentially better, individuals (exploration) while the selection phase makes sure the next generation contains, on average, slightly better individuals (exploitation). All the alternatives generated by an MGA method can be considered a population, where the minimum cost solution is the best individual, and the other individuals are near-optimal alternatives [15].

2.3.1. Components

GAs in their simplest form consist of three phases, being the initialisation, variation and selection phases. This section will explain the general goal of these phases and how they are to be interpreted.

2.3.1.1. Initialisation

In the initialisation phase, the first population is created at the start of the optimisation process. The goal is to create individuals evenly divided over the space of possible solutions to require less exploration by the GA.

Generally, the individuals of this population are generated by some random procedure, dependent on the type of values an individual consists of. The simplest way is to draw from a uniform distribution for each continuous or discrete value defining an individual. However, note that this requires some problem-specific knowledge about the minimum and maximum value of each value. As a rule of thumb, the more problem-specific knowledge that can be included to more accurately generate individuals from the space of possible solutions, the less the GA needs to explore.

2.3.1.2. Variation phase

The variation phase first aims to mix information from two or more parent individuals **p** to create one or more offspring individuals **o** using a crossover operator. Typically, two parents create two offspring. Different crossover operators can be used depending on how individuals are represented. Often, the choice of the crossover operator depends on the representation of the individuals and is often designed to preserve problem-specific structures that improve the fitness of an individual.

After crossover, new information can be added to each offspring **o** by applying a mutation operator. The operator applies a random modification, such as flipping a bit in a binary variable or adding a value drawn from a zero-centred Gaussian to a continuous variable of an individual.

The variation phase may use both the crossover and mutation operators, but their application is not strictly required. There is no consensus among researchers on which operator is more important, but some argue that crossover might be "simulated" by mutation.

However, when both operators are used, the GA generally uses a predefined probability of applying the operators. The crossover operator generally has a high probability– often in the range of 0.8-0.9– of being applied to increase the odds of offspring with high fitness. Parent individuals are copied as offspring when the crossover operator is not applied. The mutation operator has a low probability– often in the range of 0.05-0.2– such that it does not disrupt individuals with high fitness too often.

All the parents and offspring combined form the candidate population. In essence, the variation phase is a heuristic search for individuals with higher fitness without discarding the current solutions.

2.3.1.3. Selection phase

During the selection phase, individuals from the candidate population are selected to create the population for the next generation. The general principle is that individuals with higher fitness have a higher probability of being part of the next generation. This means that the average fitness of the population increases over the generations.

The most common selection operator is tournament selection. Here, a random uniform sample of a certain size is taken from the candidate population, where the individual with the highest fitness is selected to survive for the next generation. This process is then repeated until we obtain a population with the same size as the population of parent individuals. Figure 2.2 shows an example of how the Selection phase selects the blue individuals from the candidate population as the next generation.

(a) Candidate population before selection

(b) Next generation's population after selection

Figure 2.2: Example of the selection phase of a GA

2.3.2. Exploration vs. exploitation

As mentioned above, the population maintained by GAs aims to both explore the search space and exploit the best found solutions to eventually find the global optimum. However, an inherent challenge in GAs is the trade-off between exploration and exploitation. Not enough exploration might lead the search to converge to a local optimum, where too much exploration can slow down the convergence to the optimum solution, negatively impacting the required computation time. This trade-off is partially caused by the fact that it is impossible to know whether the best solution found so far is the global optimum. And since the most computationally intense part of a GA is generally the evaluation of the fitness of an individual, too much exploration can greatly increase the computational cost of each generation, reducing the amount of generations the GA can perform given a computational budget.

2.3.2.1. When to stop a GA?

Since it is not known when the GA has found the global optimum, it remains a challenge to decide when a GA has finished its search. A simple approach to this challenge is to set a number of generations for which the algorithm will run. This number can then be updated using trial and error in an attempt to get more satisfactory results in a parameter optimisation fashion. Unfortunately, GAs often have multiple parameters to optimise, which can influence their convergence speed and computational cost, like the size of the maintained population and crossover and mutation probabilities, making the number of generations needed to sufficiently explore the search space a problem-specific parameter.

Another way to decide when to stop a GA is to use a stopping criterion. These generally aim to use some measure of how well the value of the best found solution has converged as an indicator of a sufficient trade-off between exploration and exploitation. Defining a criterion of convergence essentially aims to measure when the quality of the best individual barely improves throughout the generations. It's often used in combination with a high number of maximum generations or a computation budget, stopping the algorithm when it is already 'done' to prevent unnecessary computation. Unfortunately, this only works as an indicator for convergence, so there are no guarantees that the global optimum has been found.

2.3.3. Using multiple populations

Some population-based methods guide their evolution using multiple populations concurrently. These alternative populations can be used to improve the exploration and/or exploitation abilities of GA approaches. Generally, they are created and updated either separately from each other or are extracted from a main population. The following sections will discuss research using these types of concurrent populations.

2.3.3.1. Separate populations

The work of [41] makes use of two separate populations, each maximising their respective objective function. These populations can be mixed during variation before each population selects individuals using the population's objective, aiming to improve the exploration and exploitation trade-off.

2.3.3.2. Extracted populations

The work of [42] uses a total of four populations, of which three are extracted from the single main population and are used to determine the next generation of this main population. The first alternative population consists of the k best individuals of the main population with respect to the optimisation problem's objective function, which affects the entire main population as the main functioning of the Gravitational Search Algorithm employed in this work. The next one consists of the historical personal-best locations of the k best individuals of the main population, which only influences these k individuals in the main population, aiming to prevent convergence of these individuals to local optima. The final population consists only of the historical global best individual, affecting all individuals in the main population, aiming to prevent stagnation of the population in the late search process, improving the exploitation ability of the algorithm.

3

Literature review

As we now have the necessary background of the relevant concepts of this work, this chapter will discuss the most relevant literature. Firstly, we explain the problem MGA aims to solve in more detail, the main strategies used for solving and the limitations of state-of-the-art methods.. Next, the field of diversity optimisation is introduced, specifying the different types of diversity metrics which are commonly used and the reasons for using them. Afterwards, other GA approaches of MGA methods are reviewed. Finally, we give an overview of the contribution of this work compared to other work.

3.1. MGA approaches

In literature, several implementations of MGA methods as introduced in Section 2.2 exist, varying in how they generate near-optimal alternatives. In general, the different methods try to systematically find maximally different alternatives which satisfy a near-optimality constraint. A way to do so is by employing the Hop-Skip-Jump (HSJ) algorithm, which iteratively reformulates the objective function to minimise the sum of decision variables that appeared in the previous solutions, which results in a new alternative after solving the adjusted optimisation model of Equations 3.1–3.4 where $K \in \{1,..,v\}$ represent the indices of the decision variables present in the previously found alternatives [10].

$$\min \sum_{k \in K} w_k \mathbf{x}_k \tag{3.1}$$

$$s.t. \quad C\mathbf{x} \leq \mathbf{d} \tag{3.2}$$

$$A\mathbf{x} = \mathbf{b} \tag{3.3}$$

$$f(\mathbf{x}) \leq (1+\varepsilon) f(\mathbf{x^*}) \tag{3.4}$$

Different weighted sum MGA methods can be used to steer the next search iterations in different directions, for example, to favour investments in technologies at locations where they have not been fully utilised in previous iterations, compared to the fully utilised ones [19], or to improve the algorithm's ability to find extremes of the multi-dimensional decision space [20].

Note that all weighted sum methods require the MGA algorithm to solve the complete optimisation model from scratch for every alternative it generates. This means the more complex the model, e.g. with higher spatial and/or temporal resolutions, the more time it takes to generate each alternative, while intuitively more alternatives would be needed to sufficiently explore the trade-offs when the decision space of these models is defined by an increased number of variables.

Alternatively, the MGA objective functions directly maximise some notion of distance between new alternatives and all previously found alternatives [29] or variously minimise and maximise sums of pre-defined groups of investment variables — like subsets of generation, storage and transmission capacities — which can also be formed by region and technology — i.e. minimising the sum of onshore wind capacity in a specific country when the model consists of multiple countries or the total volume of transmission expansion [25]. Weights can also be generated randomly [2, 5], optimising in random directions without considering the alternatives found so far.

However, there are limitations to the weighted sum methods. For instance, the resulting alternatives do not span the solution space evenly [20]. This conclusion follows from the fact that every constraint defining the feasible space (Equations 3.2 and 3.3) and the near-optimality constraint (Equation 3.4) is linear, meaning the near-optimal feasible space is a convex polyhedron. Therefore, we know this space is closed, such that the optimal solution to any objective function will lie on the boundary of this space [27]. This shows that the weighted sum methods will find no alternatives within this space, while points within this boundary could prove useful to speed up the decision-making process, as explained in Section 2.1.1.

Next to that, since the near-optimality bound ε is an input parameter of MGA, the resulting alternatives can vary greatly given different values of this parameter, meaning bias is introduced based on this parameter input. As this input bound is manually defined, it should serve as some upper limit of the allowed total cost, such that producing more alternatives with this exact cost deviation than those with a lower deviation does not make sense.

3.1.1. Sampling alternatives inside the near-optimal space

Other work aims to sample points inside the near-optimal space S_ε, relying on the estimation of the boundary of S_ε.

Firstly, an extension of MGA called Modelling-All-Alternatives (MAA) [27] aims to improve upon the weighted sum methods by covering the S_ε uniformly after approximating the convex shape of the space from which solutions are sampled uniformly. A uniform coverage of S_ε would allow for correlation analysis between (groups of) variables, some notion of the probability density of certain values of variables throughout the space, and provide robustness of the extracted insights. To approximate the polyhedron containing S_ε, the MAA method requires optimising the ESOM 244 times, while Neumann's weighted sum MGA method [25], used as a comparison of the output of the MAA method, required solving the same ESOM only 12 times. Due to computational limitations, the method studies only the total capacity of individual technologies instead of all the investment decision variables to limit the dimensionality of the problem, allowing the use of the Quickhull algorithm [1]. This aggregation limits the guarantee of uniformly sampling the near-optimal space of the actual variables since some alternative solutions can be achieved with more system configurations than others.

The work of Schricker et al. [33] extends the MAA approach to support discrete variables, using the assumption that all alternatives within the approximated convex polyhedron are feasible and near-optimal. First, they discretise the continuous capacity variables by limiting them to discrete capacity steps. Then, after the continuously relaxed S_ε is approximated in the same way as the MAA method, the method creates hyperrectangles encapsulating the relaxed S_ε. All vertices of the hyperrectangle are checked to satisfy the constraints. If all or none of the vertices satisfy the constraints, we know all the alternatives contained by the hyperrectangle are feasible or infeasible, respectively. If at least one vertex is feasible, the hyperrectangle is subdivided into smaller hyperrectangles, and the process repeats until all hyperrectangles are feasible or infeasible. Finally, discrete solutions within each feasible hyperrectangle can be enumerated to create the resulting alternatives. However, this method excludes some edge cases, potentially failing to identify some alternatives compared to a brute-force approach. The authors justify this by stating that their approach drastically decreases the required computation time and RAM, especially for high-dimensional solution spaces, since they do not rely on the Quickhull algorithm.

The idea of first identifying the boundary of the near-optimal space can also be applied to reduce parametric uncertainty by finding a robust near-optimal space [12]. However, they approximate the hull for each optimisation year separately, after which they compute the intersection of the near-optimal spaces of all the years considered to find the most 'robust' near-optimal space. This is done using fewer aggregated variables to make the approach computationally feasible. From the single resulting near-optimal space, they extract a solution from the interior of this space, which is as far away from the boundary as possible, representing the most robust solution. The resulting space could also be used to sample alternatives similar to the other methods described in this section. But again, its ability to evenly cover the full decision space of investment variables is limited due to the usage of aggregated variables.

3.2. Diversity as a goal

Another way to look at generating an even coverage of a space, is to think of it as finding some set of solutions P maximising a measure of diversity $D(P)$ such that every solution $s \in P$ satisfies the constraints of the

(MI)LP model, and are near-optimal. Intuitively, this diversity measure should be better for solution sets that cover the same space more evenly, and ones that cover a bigger space in a similarly even fashion. In general, the concept of diversity is defined by three properties of a set of species that are classified into types. These properties are (1) the number of types used, (2) the distribution of species into types, and (3) how different types are between them [31, 37]. Since this notion of classes does not work when species are defined in terms of a point in d-dimensional space instead of classes, only the third property could be applied, where each unique point is treated as a different type, and the diversity measure refers to how different each type is from one another. We will use the population of individuals as terminology for a set of species, as this is customary in the field of GA.

Diversity when dealing with the numerical concept of individuals is usually quantified in three different ways: as a distance measure between individuals, as a measurable attribute of the individuals (individual diversity) or as a characteristic of the population as a whole (population diversity) [36]. The work of Solow and Polasky [35] quantifies the measure of diversity from a biological point of view based on a set of pairwise distances between all individuals in a population and introduces the Solow-Polasky measure for it, aiming to quantify the dissimilarity between individuals. Additionally, they define the following requirements for a diversity measure:

1. **Monotonicity in species.** The diversity of the population should not decrease when adding an individual i that is not yet in the population of solutions P. That is, $D(P \cup i) \geq D(P)$ for some $i \notin P$.

2. **Twinning.** The diversity should not increase when adding an individual i that is already in the population P. This means $D(P \cup i) = D(P)$ for some $i \notin P$.

3. **Monotonicity in distance.** The diversity should not be decreased when the distance between individuals in the population strictly increases. When considering a one-to-one mapping $f(x)$ of population P to P' having $d(i, j) \leq d(f(i), f(j)), \forall i, j \in P$, meaning $D(P) \leq D(P')$.

When using a diversity measure to compare two populations of equal size, the first two properties are irrelevant. Similar properties are also defined in the context of diversity in chemical space [44]. Their property of dissimilarity is also relevant when only considering populations of equal size, which states that $D(\{i, j\}) \geq D(\{i, k\})$ if $d_{ij} \geq d_{ik}$ for any individuals i, j, k.

The work of Mironov et al. [22] proposes different criteria of diversity metrics when only considering the comparison of equally large populations:

1. **Monotonicity**. If one or several pairwise distances increase while keeping all other distances fixed, the diversity must increase. In contrast to the **Monotonicity in distance** property, it states that the diversity must increase. Meaning $D(P) < D(P')$ instead of $D(P) \leq D(P')$. Figure 3.1 shows two different populations of equal size where each pair of individuals in P can be mapped one-to-one to a pair in P', serving as an example when the **monotonicity** property applies, such that we expect $D(P') > D(P)$.

   Intuitively, the goal of this property is to make the diversity metric favour populations that include the more extreme regions of S_ε.

(a) Population P (b) Population P'

Figure 3.1: Example populations of when the monotonicity property applies.

2. **Uniqueness**. Considering two equally sized populations where P' only differs from P by one individual, which is unique in P' but a duplicate in P as shown in Figure 3.2. In this case, we expect $D(P) < D(P')$. Figure 3.2 shows an example of when this property applies: two different populations of equal size that only differ in one individual, which is a duplicate in P — represented by the large dot in the middle of Figure 3.2a — and unique in P'. The uniqueness property then states $D(P) < D(P')$.

(a) Population P (b) Population P'

Figure 3.2: Example populations of when the uniqueness property applies.

3. **Continuity**. A diversity function must be continuous.

Note that these properties do not require all individuals of a population to be unique, as the authors aim to be able to compare the diversity of populations even if they have duplicates.

After defining these axioms, the authors prove none of the diversity measures used in literature satisfy these properties and define new diversity measures which do satisfy them, but which are NP-hard to compute, leaving open the problem of defining a polynomial-time measure that satisfies all the properties.

3.2.1. Measuring diversity

In the literature, several different diversity measures are introduced. They are generally defined in terms of a sum of pairwise distances 3.5 and fractional pairwise distances 3.6 used in [3]), discrepancy [48], or size of the space covered by the points in terms of inter-quartile ranges, diameter or volume for a distance measure d_{ij} between points i and j and some $\gamma > 0$. A complete overview of the metrics considered can be found in Appendix A.1.

$$\sum_{i<j} d_{ij}^{\gamma} \quad \gamma > 0 \tag{3.5}$$

$$\sum_{i<j} \frac{1}{d_{ij}^{\gamma}} \quad \gamma > 0 \tag{3.6}$$

When using diversity measures based on distance metrics, the distance metric puts requirements on the space in which the points are defined. The work of Ulrich et al. [41] use the properties of a diversity measure by Solow and Polasky [35] stated above, and specify that the solution space X is not required to be Euclidean or that the triangle inequality is satisfied. Instead, they require X to be a semi-metric space, ensuring the following properties of the distance between any two points $i, j \in X$ to be non-negative $d(a,b) \geq 0$, symmetric $d(a,b) = d(b,a)$ and to have the identity of indiscernibles $d(a,a) = 0$.

A different field of research related to evenly filling a space with points is the generation of low-discrepancy sets of points. The intuition behind low-discrepancy populations is to think about the population with optimal diversity within a square space to have the number of individuals of a population that lie within a box within the square space be proportional to the volume of the box. In other words, we would like the difference $\text{Vol}([a,b]) - \frac{1}{n}|X \cap [a,b]|$ to be minimal, for all possible boxes $[a,b]$ within the square space [23]. From this follows the definition of discrepancy in Equation 3.7 to be the largest of such a deviation after transforming the square space to the unit square space.

$$D(X) = \sup\left\{\text{Vol}([a,b]) - \frac{1}{n}|X \cap [a,b]| \mid a \leq b \in [0,1]^d\right\} \tag{3.7}$$

Generating a population with low-discrepancy in higher dimensions is an optimisation problem that proved challenging by itself, where GAs have also been employed to create such populations, as it is an optimisation problem not admissible by traditional analytical approaches [23]. Several different notions of computing the discrepancy of a given population exist, where they generally differ in the strategy of selecting boxes within the unit square anchored to individuals and/or boundaries of the square space [48].

However, there are some complications when using this metric for convex spaces, since this metric is defined for square spaces. When trying to solve this by taking the minimum axis-parallel square containing the convex space, either with or without normalizing each dimension, there would be an issue that the box with the maximum deviation as defined in Equation 3.7 would be between the convex space and the square space around it when the points within the convex space have a sufficiently low discrepancy. Next to that, the exact convex shape of the space is not known beforehand in the context of applying MGA, where approximating this shape can come at a big computational cost as discussed in Section 3.1.1.

3.2.2. Diversity optimisation

Closely related to MGA, other research has aimed to find a set of solutions that maximise the diversity of the set with a minimum quality threshold (e.g. such that the solutions are near-optimal) [24, 40]. For this purpose, GAs can be used in a single- [11, 40] and multi- [24] objective optimisation setting, where the next population is created from a candidate population by iteratively removing the individual that least contributes to the diversity. These approaches are further reviewed in section 3.3.2.

The concept of novelty search is related to diversity optimisation. Its goal is to generate individuals that are significantly different from previously found individuals to escape deceptive traps in GAs, which can be interpreted as overcoming local optima. However, it is generally used to find different individuals in terms of behaviour (i.e. objective space) as opposed to genotype (i.e. decision space). It does so by neglecting the original objective function and maximising a novelty measure [32], or considering both in a multi-objective setting [18].

3.2.3. Comparing metrics

There have been works to compare these diversity metrics in terms of their ability to lead to diverse sets of solutions [24]. The work demonstrates how a few diversity measures can be used as an out-of-the-box tool to explore a space.

The work that considered properties of diversity measures when comparing equally sized populations in Section 3.2 [22], also compares several diversity measures, showing some undesirable behaviour for using them to compare two different populations and as an optimisation goal. They show that using metrics based on averages of pairwise distances (e.g. ones defined as a sum of distances) pushes individuals to the boundary of the space, leaving central areas empty. Using a diversity measure based on population diameter has a similar result. When using a diversity measure based on energy, the optimisation behaviour enforces a more uniform distribution by pushing away the closest elements. The only flaw of such a measure is that it doesn't work well for comparing two populations when duplicates are present, as those will result in a value of $-\infty$.

3.3. GA for MGA

This section will first discuss GA methods in the literature that incorporate diversity without optimising it, and ones that optimise diversity. Then, we will discuss how other work handles constraints to ensure the resulting population is feasible and near-optimal.

3.3.1. Using diversity

The work of Squillero et al. [36] survey different GA methods that promote diversity to avoid premature convergence on a sub-optimal solution. Some approaches make use of subpopulations that are treated separately during selection, introduce diversifiers located inside relatively empty parts of the space [17] or add an additional diversity-related objective function of an individual to use in a multi-objective GA [9].

However, these methods do not aim to optimise the diversity of the total population but rather employ GAs to find a single optimal point, or multiple non-dominated points in a multi-objective setting, such that no other point is objectively better according to the given objective function of the optimisation problem. In the MGA scenario, this objective comparison between points is much harder when the objective function is a diversity measure defined over the entire population instead of a single individual. This means no two individuals can be compared without considering the rest of the points.

3.3.2. Optimising diversity

When using a GA to optimise diversity, the commonly used parts of GAs, as discussed in Section 2.3, require changes. One approach applying a GA as an MGA method consists of multiple separate subpopulations, as introduced in Section 2.3.3.1, to be maximally different from each other [16, 45–47]. Even though these methods can have a larger number of individuals in total, they produce a similar number of alternatives as previously mentioned MGA methods, since each subpopulation results in a single alternative. These methods use a distance metric intending to generate maximally different solutions, by incorporating the distance of an individual to the centroid of all other subpopulations in the objective function. This objective function pushes away subpopulations from one another in a way similar to the weighted sum methods, aiming to find alternatives that are maximally different from each other. However, it does not indicate how well the resulting alternatives cover the (near-optimal) space, even though one could argue that the alternatives resulting from these methods cover the solution space more evenly since the subpopulations are optimised concurrently instead of generating the alternatives one by one.

Another approach uses a single population to optimise both a single objective value and the population diversity by iteratively switching between these optimisation goals [40]. This means they iteratively optimise an individual- and population-level objective, which they call a 'mixed multiobjective problem'. Again, the Solow-Polasky measure is used to measure the diversity of a population.

To ensure the resulting population is sufficiently performing in terms of the objective value, a near-optimal bound ε is provided, and the initial bound is tightened every iteration until it reaches ε. After creating a candidate population using variation, the individuals that contribute least to the diversity are sequentially discarded until a new population of a specified size is obtained that maximises diversity. Since computing the diversity of all possible subsets of the specified size is infeasible, this approach serves as a greedy strategy for a maximally diverse population, optimising a surrogate measure of diversity.

Other work introduces a framework to concurrently optimise the diversity in the objective and decision space in a multi-objective setting [41]. They do so by optimising two populations in parallel, where one approximates the Pareto front of the original multi-objective problem and the other maximises the diversity of the population within a given bound ε of distance to the Pareto front. As a measure of the diversity of both populations, they use the Solow-Polasky diversity measure. However, since this measure is costly to compute due to needing matrix inverses, they approximate how much an individual contributes to the diversity $D(P \cup \mathbf{i}) - D(P)$ with $\min_{s \in P \setminus \mathbf{i}} d(\mathbf{i}, s)$. So the distance to the closest other individual in the population is used to approximate how much that individual contributes to the overall diversity.

Finally, a simple GA can be modified such that all near-optimal individuals have the optimal objective value, and use fitness sharing to promote diversity of solutions.

3.4. Contribution

This work aims to improve the diversity of the resulting set compared to the weighted sum MGA approaches, resulting in increased confidence that the trade-offs in the near-optimal space are sufficiently represented in the resulting solutions and removing introduced bias by choice of the near-optimality constraint. To create the alternatives, it will not require approximating the shape of the near-optimal space, which is a required precomputation of the MGA approaches that can generate alternatives within the near-optimal space, preventing the need to use aggregated variables for a reduced dimensionality to ensure a feasible computational cost.

There are multiple differences between the state-of-the-art MGA methods and the method proposed in this work. First of all, the weighted sum MGA methods modify the ESOM and solve it to obtain each alternative individually, which are guaranteed to lie on the boundary of the near-optimal space S_ε, while this work optimises the diversity of a set of alternatives, being able to find points within S_ε and only needing the ESOM to solve for unit commitment. The MAA method is also able to find alternatives within S_ε. However, it requires an approximation of the boundary of S_ε, which greatly limits the number of investment variables of the ESOM. This precomputation is not required by the GA approach this work proposes.

Finally, it explores whether the usage of a GA has computational benefits as it does not require solving the optimisation problem for each generated alternative, potentially allowing for the generation of more alternatives with similar computational cost compared to other approaches applied to ESOMs.

4

Methodology

This chapter explains the steps taken to design the GA used to perform MGA on ESOMs. First of all, we formulate the optimisation solved by the GA in Section 4.1, introducing new properties for the diversity metric optimised by the GA. Then, we define how we will compare different diversity metrics in Section 4.2 using example populations to test whether the metrics can order the populations similarly to our expectations in terms of diversity. Afterwards, the challenge of finding inner points in high-dimensional spaces is explained in Section 4.4. Then, we define all the components of the GA in Section 4.5 and describe how its performance is evaluated in Section 4.6. Finally, in Section 4.7, we describe how the GA is applied to the SECURES model, an ESOM used to optimise investments for the whole of Europe.

4.1. Formulating diversity optimisation for MGA

In Section 3.2, we discussed how doing MGA can be seen as optimising the diversity of a population in a near-optimal space and discussed some properties of measuring this diversity. This section formulates the problem of optimising the diversity metric with a GA and introduces new properties tailored to our goals, which guide the choice of the metric to be optimised.

To rewrite the problem formulation, we start with the adjusted optimisation model for MGA in Equations 3.1–3.4. Our formulation changes the objective function in Equation 4.1 to maximise the diversity metrics over the investment variables of all alternatives $\mathbf{x}$, consisting of investment variables $\mathbf{x}_{inv}$ and operational variables $\mathbf{x}_{op}$ and forming the population P, such that $\mathbf{x} = \begin{pmatrix} \mathbf{x}_{inv} \\ \mathbf{x}_{op} \end{pmatrix} \in P$. Then, each alternative $\mathbf{x}$ is constrained to be feasible according to the original ESOM (Equations 4.2 and 4.3) and near-optimal (Equation 4.4).

$$\max_{\mathbf{x}_{inv}} \quad D(\{\mathbf{x}_{inv} | \mathbf{x} \in P\}) \tag{4.1}$$

$$s.t. \quad C\mathbf{x} \leq \mathbf{d} \tag{4.2}$$

$$A\mathbf{x} = \mathbf{b} \tag{4.3}$$

$$f(\mathbf{x}) \leq (1+\varepsilon) f(\mathbf{x^*}) \tag{4.4}$$

In our formulation, we want the population that maximises the diversity metric D to cover the near-optimal space S_ε as uniformly as possible, including individuals located at and between the most extreme regions of S_ε. We aim for the diversity metric to differentiate between various populations P based on their uniform coverage of a space, enabling our GA to optimise for this metric to obtain an even coverage of S_ε. To formalise our intuition on more diverse populations, we introduce the following properties of a diversity metric :

- **Covering radius distribution.** When considering the mean μ_P^r and standard deviation σ_P^r of the minimum Euclidean distances of all individuals to any other individual (Equation 4.5) of some population P. If some other population P' with the same number of non-duplicate individuals has $\mu_{P'}^r \geq \mu_P^r$ and $\sigma_{P'}^r \leq \sigma_P^r$, where at least one of the conditions is strictly greater or smaller, we want $D(P) < D(P')$. An

example of two populations where these conditions apply is shown in Figure 4.1 shows an example of two different populations of equal size where the covering radius distribution property applies, such that we expect $D(P') > D(P)$.

Another way to interpret this property is that populations for which the maximum distance from any individual in the space to any individual in the population — e.g.$\max_{\mathbf{x}\in S_\varepsilon} \min_{\mathbf{i}\in P} d(\mathbf{x},\mathbf{i})$ — is lower, should be more diverse. Intuitively, this property attempts to prevent the diversity metric from favouring individuals that lie on the boundary, as is often the result when maximising average pairwise distance, as shown by Mironov et al. [22].

$$\begin{aligned} \mu_P^r &= \frac{1}{|P|}\sum_{i\in P} \min_{j\in P, i\neq j} d(i,j) \\ \sigma_P^r &= \sqrt{\frac{1}{|P|}\sum_{i\in P}\left(\left(\min_{j\in P, i\neq j} d(i,j)\right) - \mu_P\right)^2} \end{aligned} \tag{4.5}$$

(a) Population P (b) Population P'

Figure 4.1: Example populations of when the **covering radius distribution** property applies.

- **Rotation invariance.** Transforming a population P by rotating all the individuals in P around a set point to obtain P' — as exemplified in Figure 4.2 — should not change its diversity such that $D(P) = D(P')$. Figure 4.2 shows P and the transformed population P' obtained by rotating P around the centre of the space by 45°clockwise. Serving as an example of the **rotation invariance** property such that $D(P) = D(P')$.

  This property essentially states our goal of using a diversity metric that is not axis-aligned, as we are interested in covering S_ε and not independently covering each investment variable.

(a) Population P (b) Population P'

Figure 4.2: Example populations of when the **rotation invariance** property applies.

The properties discussed above serve as an addition to the ones from the literature — as discussed in Section 3.2 — which are relevant to our use case. This excludes the twinning and monotonicity in species properties, as we do not change the size of the population for which we want to optimise the diversity.

4.2. Selection of diversity metric

To motivate the choice of diversity metric, we begin by explaining the types of metrics considered. Afterwards, several test populations are constructed and ordered relative to each other. The metrics will be assessed on how they distinguish between more and less diverse populations, of which the results are presented in Section 5.1

4.2.1. Types of diversity metrics

We consider many different diversity metrics from the literature, which can generally be categorised as follows:

- *Sum of distances.* The measure is a function of the sum of distances $\sum_{a,b\in P} d(a,b)^r$ with positive constant c.
- *Sum of fractional distances.* The measure is a function of the fractional distances $\sum_{a,b\in P} \frac{1}{d(a,b)^r}$ with some positive constant c.
- *Discrepancy.* Some approximation of the largest difference between the volume of a square subspace and the fraction of the total number of individuals in that space, as defined in Equation 3.7. See Section 3.2.1 for further explanation.
- *Other.* Other metrics found in literature relate to some population volume or radius, dimensional interquartile range or variance of distance to the average individual.

Note that the discrepancy is only defined for axis-aligned square spaces, making it unsuitable for our GA approach. However, as low-discrepancy populations serve as a good example that uniformly cover the space, we still consider these measures to check whether our preferences between populations make sense. For the full overview of all considered diversity metrics, see Appendix A.1.

4.2.2. Testing properties of diversity metrics

Properties from the literature (Section 3.2) and the ones we defined (Section 4.1) are tested using the example populations from Figures 3.1, 3.2, 4.1 and 4.2. An overview of what properties the different types of diversity metrics generally have is given in Table 4.1.

The reason why the metrics based on the sum of distances do not satisfy the covering radius distribution property is that increasing μ_P^r and decreasing σ_P^r can result in a lower sum of distances, as is the case in Figure 4.1. The sum of fractional distances does not adhere to the uniqueness property as we cannot divide by a distance of zero. Discrepancy-based metrics do not satisfy the rotation invariance property as it is defined by the axis-aligned box with the largest difference in its relative volume and the relative number of points it contains. The same definition is the reason the discrepancy-based metrics do not satisfy the covering radius distribution property, as there can be a population where a modification to a point does not change this axis-aligned box while increasing μ_P^r and decreasing σ_P^r.

Property \ Metric type	Sum of Distances	Sum of Fractional distances	Discrepancy
Monotonicity	✓	✓	✓
Uniqueness	✓	-	✓
Continuity	✓	✓	✓
Covering radius distribution	-	✓	-
Rotation invariance	✓	✓	-

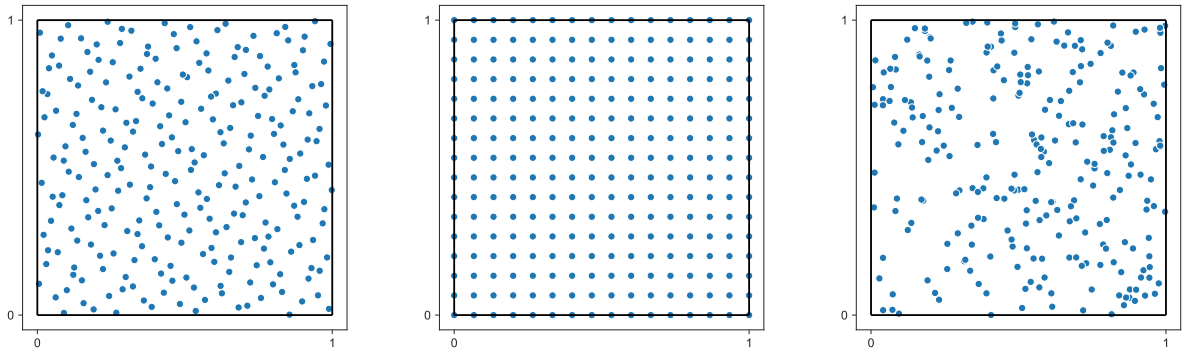
Table 4.1: The properties which each type of diversity metric generally has.

4.2.3. Testing populations in Unit Square space

In the experiments defined in the next subsections, we consider the different types of diversity metrics found in literature as discussed in Section 3.2.1 We test different interpretations of uniform coverages, how modifying the concentration of individuals changes the diversity, whether diversity metrics are biased towards the boundary and whether a suboptimal population arising during the optimisation process is less preferred.

4.2.3.1. Uniform coverages

We use a Sobol sequence, uniform grid, and a multivariate uniform sample to test whether a diversity metric prefers a specific uniform distribution, as shown in Figure 4.3.



(a) Sobol sequence P_s (b) Uniform grid P_g (c) Multivariate uniform sample $P_{mv.u}$

Figure 4.3: Three different uniformly covering populations of 256 individuals in unit square space.

We consider both the Sobol sequence and the uniform grid as the most uniform coverage in our experiment. The reason we do not prefer one of these coverages over the other is that the Sobol sequence is better in terms of discrepancy — as explained in Section 3.2.1 — as it is a quasi-random sequence specifically designed to be low-discrepancy [23]. The uniform grid, on the contrary, is worse in terms of diversity, but is better according to the covering radius distribution property discussed in Section 4.1. The multivariate uniform sample serves as a comparison to check whether a diversity metric considers the Sobol sequence and uniform grid more diverse compared to a population we would obtain if we could randomly sample individuals within the space.

4.2.3.2. Coverages with modified concentration

Alternative populations are included, which should be worse than the uniform populations and are tested for some desirable preferences between them. These alternative populations highlight how sensitive the diversity metrics are to having more and less concentrated parts of the space.

Firstly, the Sobol sequence population is modified to include non-overlapping areas with a relatively high and low concentration of individuals as shown in Figure 4.4. The population is modified by creating 'blind spots' or concentrated spots. A concentrated spot is created by defining a circle centred at $\mathbf{c}$ with radius r such that every individual $\mathbf{i}$ within the circle is modified to create $\mathbf{i}_c$ by moving it towards $\mathbf{c}$ centre such that the distance towards the centre is halved, as defined in Equation 4.6.

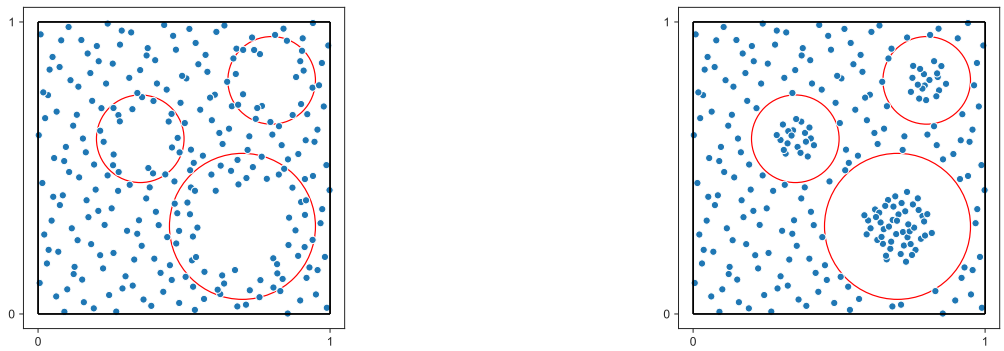
$$\mathbf{i}_c = \mathbf{c} + \frac{1}{2}(\mathbf{i} - \mathbf{c}) \quad \forall \mathbf{i} \in P : d(\mathbf{c}, \mathbf{i}) < r \tag{4.6}$$

A 'blind spot' is created similarly, halving the distance to the closest point on the border of the circle $g(\mathbf{i}, \mathbf{c}, r)$ — as defined in Equation 4.7a — instead of the centre $\mathbf{c}$ to create $\mathbf{i}_b$. See the definition in Equation 4.7b.

$$g(\mathbf{x}, \mathbf{c}, r) = \mathbf{c} + \frac{r}{d(\mathbf{c}, \mathbf{x})}(\mathbf{x} - \mathbf{c}) \tag{4.7a}$$

$$\mathbf{i}_b = g(\mathbf{i}, \mathbf{c}, r) + \frac{1}{2}(\mathbf{i} - g(\mathbf{i}, \mathbf{c}, r)) \quad \forall \mathbf{i} \in P : d(\mathbf{c}, \mathbf{i}) < r \tag{4.7b}$$

We hypothesise that both of the newly introduced populations are less diverse than the Multivariate uniform sample from Figure 4.3c, as they leave relatively large areas uncovered. However, since $\mu^r_{P_{mv.u}} < \mu^r_{P_{s.b}} < \mu^r_{P_{s.c}}$ and $\sigma^r_{P_{mv.u}} < \sigma^r_{P_{s.b}} < \sigma^r_{P_{s.c}}$ we expect $D(P_{mv.u}) < D(P_{s.b}) < D(P_{s.c})$ according to the covering radius distribution property. The differences between the value of the diversity metric of the Multivariate uniform sample, the Sobol sequence with blind spots and the Sobol sequence with concentrated spots population will highlight how sensitive the metric is to individuals that lie relatively close to each other.



(a) Sobol sequence with blind spots $P_{s.b}$

(b) Sobol sequence with concentrated spots $P_{s.c}$

Figure 4.4: Modified Sobol sequence populations in unit square space with spots of relatively high and low concentration of individuals.

4.2.3.3. Testing bias of diversity metrics towards the border of the space

Next, we define populations to test whether diversity metrics are biased towards the border of the space. We again modify the Sobol sequence to create a population that leaves the middle of the vertical axis empty, and one that covers a smaller square inside the unit square.

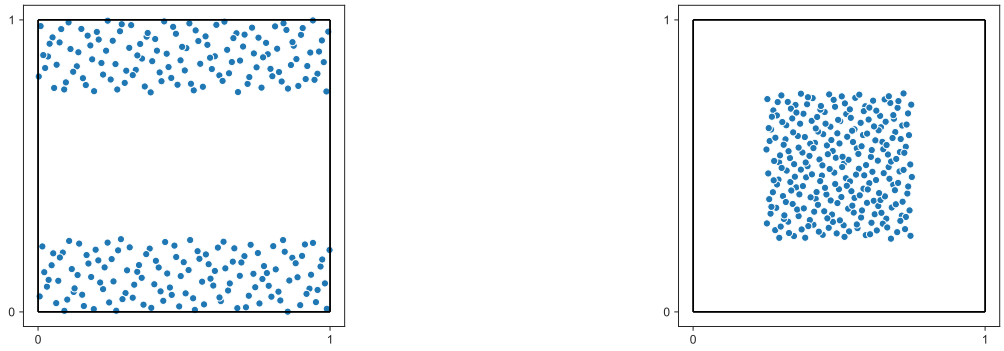
The population $P_{p.sep}$ that splits the vertical y-axis by applying the mapping $f_{sep}(y)$ on the y value of individuals $\mathbf{i} = (x, y)$ to obtain $\mathbf{i}_{s.sep} = (x, g(y))$. As defined in Equation 4.8, the mapping changes the y-value for individuals in the top half of the space, e.g. $y \in [0.5, 1]$, to a value in $[0.75, 1]$ while keeping the relative distance to the upper and lower value of the range the same. The y-value of the individuals in the bottom half is mapped to a value in $[0, 0.25]$ in a similar manner.

To create the population that covers the smaller square $P_{s.cen}$, we generate a Sobol sequence and rescale each individual to fit in the smaller square defined by diagonal points at $(0.25, 0.25)$ and $(0.75, 0.75)$ by applying the mapping mapping $f_{cen}(a)$ — defined in Equation 4.9 — to each $\mathbf{i} = (x, y)$ to obtain individual $\mathbf{i}_{s.cen} = (f_{cen}(x), f_{cen}(y))$.

$$f_{sep}(y) = \begin{cases} \frac{y-0.5}{2} + 0.75 & y \in [0.5, 1] \\ \frac{y}{2} & y \in [0, 0.5) \end{cases} \tag{4.8}$$

$$f_{cen}(a) = \frac{a}{2} + 0.25 \tag{4.9}$$

Since $P_{s.cen}$ covers a quarter of the unit square, and $P_{s.sep}$ half of it, we expect that $D(P_{s.cen}) < D(P_{s.sep})$. As both new populations are a modification of the Sobol sequence P_s with less coverage, we expect to have $D(P_{s.cen}) < D(P_s)$ and $D(P_{s.sep}) < D(P_s)$.



(a) Sobol sequence with vertical separation $P_{s.sep}$

(b) Sobol sequence in center square $P_{s.cen}$

Figure 4.5: Modified Sobol sequence populations in unit square space used to test bias of diversity metrics to the boundary of the space.

4.2.3.4. Suboptimal coverage

Finally, we use a Multivariate normal sample $P_{mv.n}$ — as shown in Figure 4.6 — as a baseline for a suboptimal uniform coverage. The sample can be interpreted as one of the populations generated by a GA before con-

verging to a population with optimal diversity. When the initial population of the GA is initialised close to the centre of the unit square, we would expect subsequent populations to get closer towards the boundary of the space, eventually creating a similar population as in Figure 4.6.

Each individual $\mathbf{i}$ is drawn from a multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$ having mean vector $\mu = \begin{pmatrix}0.5 & 0.5\end{pmatrix}^T$ and covariance matrix $\Sigma = \begin{pmatrix}0.05 & 0 \\ 0 & 0.05\end{pmatrix}$. Every sampled $\mathbf{i}$ which does not lie within the unit square is redrawn until the population consists of 256 individuals.

We expect the diversity $D(P_{mv.n})$ to be worse than all the uniform populations of Figure 4.3. The Sobol sequence with blind or concentrated spots $P_{s.b}$ and $P_{s.c}$ and with vertical separation $P_{s.sep}$ are also expected to be more diverse following the covering radius distribution property. However, when the diversity of the Sobol sequence with vertical separation is significantly higher than that of the population with uniformly sampled multivariate individuals, according to a diversity metric, it indicates a potential bias towards the border of the space.

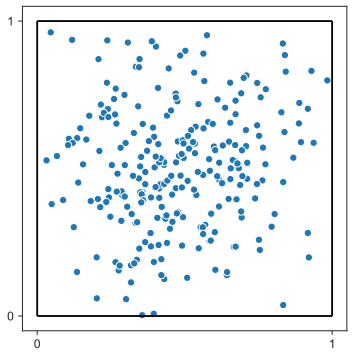


Figure 4.6: Multivariate normal sample population $P_{mv.n}$

Table 4.2 provides an overview of all pairwise orders of diversity of the test populations. In the table, '>' indicates $D(P_{row}) > D(P_{column})$, and '<' indicates the reverse. Entries marked with '(>)' or '(<)' follow from the covering radius distribution property. Additional populations considered, but not included in this comparison, are listed in Appendix A.2.

Type	**Population**	P_s	P_g	$P_{mv.u}$	$P_{s.b}$	$P_{s.c}$	$P_{s.sep}$	$P_{s.cen}$	$P_{mv.n}$
Uniform	P_s	=	(<)	(>)	>	>	>	>	>
	P_g	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	(<)	(<)	=	(<)	(<)	-	-	>
Concentration	$P_{s.b}$	<	(<)	(>)	=	(<)	-	-	(>)
	$P_{s.c}$	<	(<)	(>)	(>)	=	-	-	(>)
Bias to border	$P_{s.sep}$	<	(<)	-	-	-	=	>	(>)
	$P_{s.cen}$	<	(<)	-	-	-	<	=	-
Suboptimal	$P_{mv.n}$	<	<	<	(<)	(<)	(<)	-	=

Table 4.2: Overview of expected order of diversity between test populations in unit square space.

4.2.4. Testing populations in convex space

To make sure the diversity metrics work similarly in a convex space that is not axis-aligned, we use variants of the populations mentioned in Section 4.2.3 in a convex subspace of the unit square. We test whether each diversity metric ranks the populations in convex space in the same order as their counterparts in unit square space.

The populations generated by a Sobol sequence or sampling from a distribution are modified to cover the convex subspace in Figure 4.7 in the following way: First, each individual is created by the Sobol sequence and then transformed to fit in the square space containing the convex space defined by diagonal points at $(0.2, 0)$ and $(1, 0.8)$ following from the minimum and maximum values of each dimension of all the space's

vertices, as defined in Equation 4.10.

$$g(a) = \frac{4a}{5} + 0.20 \tag{4.10}$$

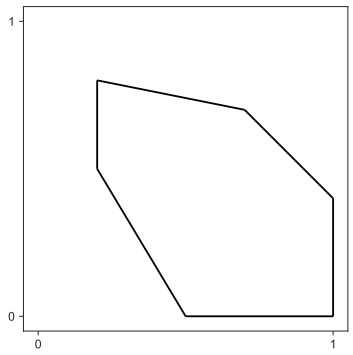


Figure 4.7: The convex space in which each population is transformed to.

The individuals from the multivariate uniform and normal distributions are respectively sampled from $\mathcal{U}_{[0.2,1]\times[0,0.8]}$ and $\mathcal{N}(\mu,\Sigma)$ with $\mu = \begin{pmatrix}0.6 & 0.4\end{pmatrix}^T$ and $\Sigma = \begin{pmatrix}0.04 & 0\\ 0 & 0.04\end{pmatrix}$.

Then, any modifications are applied where blind or concentrated spots are applied using the same circles defined in the unit square spaces and the vertical separation and centre square with respect to the square containing the convex space. Finally, all individuals that lie outside the space's boundary are removed, and the procedure is repeated until the population contains 256 individuals. The uniform grid population is excluded as it is non-trivial to make it fit in the convex space. These populations are visualised in Appendix A.2.2.

Considering the populations modified to fit in the convex space, two expectations of Table 4.2 change according to the covering radius distribution property. First, the order of the Sobol sequence with blind spots $P_{s.b}$ and the Sobol sequence with concentrated spots $P_{s.c}$ is flipped such that $D(P_{s.b}) < D(P_{s.c})$. Next to that, the comparison of the Sobol sequence with vertical separation $P_{s.sep}$ and the multivariate normal sample $P_{mv.n}$ is inconclusive.

Type	**Population**	P_s	$P_{mv.u}$	$P_{s.b}$	$P_{s.c}$	$P_{s.sep}$	$P_{s.cen}$	$P_{mv.n}$
Uniform	P_s	=	(>)	>	>	>	>	>
	$P_{mv.u}$	(<)	=	(<)	(<)	-	-	>
Concentration	$P_{s.b}$	<	(>)	=	(>)	-	-	(>)
	$P_{s.c}$	<	(>)	(<)	=	-	-	(>)
Bias to border	$P_{s.sep}$	<	-	-	-	=	>	-
	$P_{s.cen}$	<	-	-	-	<	=	-
Suboptimal	$P_{mv.n}$	<	<	(<)	(<)	-	-	=

Table 4.3: The expectations of how each pair of test populations performs against each other in terms of diversity in unit square space. '>' indicates $D(P_{row}) > D(P_{column})$, and '<' indicates the reverse. Entries marked with '(>)' or '(<)' follow from the covering radius distribution property.

4.3. The near-optimal investment space

Now that we have specified how we will compare different diversity metrics, we will further detail the space covered by a population. As shown in Equation 4.1, we only care about the diversity of the investment variables of each $\mathbf{x}_{inv}$. However, given some investment as part of an individual $\mathbf{x}_{inv}$, we need to solve the original ESOM as given in Equations 2.1–2.3 for unit commitment to make sure the alternative is feasible and within our cost slack ε.

Fortunately, solving the model for unit commitment reduces the ESOM's complexity, meaning we can perform more solves for unit commitment only compared to the full ESOM for the same computational bud-

get. This difference in solving time will be further explored in the scalability experiment discussed in Section 4.6.4.

Since the feasible space of an LP is always convex, the near-optimal space S_ε of investment variables is also convex. This property is used in the operators of our GA to reduce the number of times the ESOM needs to be solved for unit commitment, as will be explained in Section 4.5.

4.4. Covering high-dimensional spaces

According to the monotonicity property, we expect the diversity of populations consisting only of individuals that lie on the boundary of S_ε to be preferred to those that do not for certain combinations of population size $|P|$ and number of investment variables v, e.g. the dimensionality, considered in the ESOM. This section will substantiate the following expectations:

- When $|P| \leq 2v$, we always expect the most diverse population to only consist of individuals on the boundary of S_ε.
- When $2v < |P| \leq 2^v$, it depends on the convex shape of S_ε whether the most diverse population consists of only boundary individuals.
- When $2^v < |P|$, we always expect the most diverse population to include at least a single individual that does not lie on the boundary of S_ε.

4.4.1. Covering high-dimensional hypercubes

To illustrate the intuition behind this relation between the population size and number of investment variables, we consider a population to cover S_h, defined as a v-dimensional hypercube with C as the set of all its 2^v corners. If $|P| = |C|$, a population $P = C$ is optimal, because substituting any individual with some $\mathbf{i} \notin C$ will only result in pairwise distances being decreased, making the population $P = C$ more diverse according to the monotonicity property.

If $|P| < |C|$, replacing any individual $\mathbf{i} \in P$ on the boundary of the minimum convex hull containing P and not on the boundary of S_h by any individual $\mathbf{i}' \in S_h$ on the line passing through $\mathbf{i}$ and the centroid of the convex hull $\bar{\mathbf{i}}$ such that $\mathbf{i}' = \mathbf{i} + a(\mathbf{i} - \bar{\mathbf{i}})$ with $a > 0$ to create P'. Then $D(P') > D(P)$ according to the monotonicity property as some pairwise distances will only increase.

When $\mathbf{i}$ is not on the boundary of the minimum convex hull, it is inside this hull. In that case, there exists a individual $\mathbf{i}'$ on the boundary of the hypercube which can replace $\mathbf{i}$, such that the distance to all other individuals increases, to create a population P' having $D(P') > D(P)$ according to the monotonicity property. This process of replacing individuals can then be repeated until a population P^* is created where all individuals lie on the boundary of S_h. This means, if we have $n \leq 2^v$, we expect the most diverse population only to contain individuals on the boundary of S_h.

4.4.2. Covering high-dimensional convex spaces

The near-optimal space S_ε is not necessarily a hypercube, but can be of any convex shape as mentioned in Section 4.3. Then, suppose we normalise S_ε using the minimum and maximum value of each dimension. In that case, we can define an axis-aligned hypercube based on the minimum and maximum value of each dimension within S_ε, such that each side of the hypercube intersects with S_ε in at least a single point and S_ε is a subspace of the hypercube. Next, we assume there are no redundant dimensions in which the minimum and maximum value are equal, and S_ε is some diamond shape defined by the set $|C|$ consisting of $2v$ vertices, such that every vertex intersects with exactly one side of the hypercube. For such a space S_ε, the same intuition when $|P| \leq |C|$ when S_ε is a hypercube holds, meaning we expect the population with maximum diversity to only contain individuals on the boundary of S_ε when $|P| \leq 2v$.

When one or more vertices defining S_ε intersect with more than one side of the minimum hypercube containing S_ε, this intuition does not hold as well, such that there are examples where the most diverse population contains individuals not on the boundary of S_ε. However, we assume the number of those spaces is limited due to the normalisation of the dimensions of S_ε.

4.5. GA setup

The GA proposed in this work is based on the standard GA procedure of using the initialisation, crossover, mutation and selection operators. It modifies these operators to perform diversity optimisation for ESOMs

and incorporates the feasibility checking and normalisation of individuals. See an overview of the GA loop in Algorithm 1.

Algorithm 1 The main loop. Update population by performing crossover, mutation and selection until the maximum number of generations or the stopping criteria is reached.

function CHECKFEASIBILITY(P)
  $F \leftarrow \{b_1, ..., b_{|P|}\}$
  for $j \leftarrow 1, |P|$ **do**
    $\mathbf{x} \leftarrow$ SOLVE($\mathbf{i}_j$) ▷ Retrieve the full solution of the ESOM given investment variables
    $b_j \leftarrow \mathbf{x}$.feasible **and** $f(\mathbf{x}) \leq (1+\varepsilon) f(\mathbf{x}^*)$ ▷ Individual is feasible if solution is feasible and near-optimal
  end for
  return F
end function

function GALOOP(n_{ind}, $\#gen_{max}$, p_c, α, $p_{m.ind}$, $p_{m.var}$, β, $\#gen_{conv}$)
  $P \leftarrow$ INITIALISE(n_{ind})
  $F_P \leftarrow$ CHECKFEASIBILITY(P)
  $d_{max} \leftarrow -\infty$
  $counter \leftarrow 0$ ▷ Number of generations without improvements
  for $gen \leftarrow 1, \#gen_{max}$ **do**
    $P \leftarrow$ NORMALISE(P)
    $O \leftarrow$ CROSSOVER(P; p_c, α) ▷ Generate n_{ind} offspring
    $O \leftarrow$ MUTATE(O; $p_{m.ind}$, $p_{m.var}$, β)
    $F_O \leftarrow$ CHECKFEASIBILITY(O)
    $P \leftarrow$ SELECT($P \cup O, F_P \cup F_O$; n_{ind})

    $d \leftarrow D(P)$
    if $d > d_{max}$ **then**
      $d_{max} \leftarrow d$
      $counter \leftarrow 0$
    else
      $counter \leftarrow counter + 1$
      if $counter = \#gen_{conv}$ **then**
        return P
      end if
    end if
  end for
  return P
end function

4.5.1. Parameters

An overview of the parameters used in the GA loop is shown in Table 4.4.

4.5.2. Initialisation

Some of the GAs that optimise diversity — as discussed in Section 3.3.2 — assume the initial population all adhere to the minimum quality requirement, i.e. being near-optimal [11, 24]. At the same time, we want the initial population to have as much variation as possible to aid in the exploration and exploitation trade-off as mentioned in Section 2.3.2.

As stated in Section 2.3.1.1, the more problem-specific knowledge we can use, the less the GA needs to explore. However, for our problem, we have limited knowledge about the size and bounds of the near-optimal space S_ε. The only useful information we could use is the minimum cost investment plan, as we know it is within S_ε, and the coefficients of the investment decision variables $\mathbf{c}_{inv}$ of the ESOM's objective function in Equation 2.1.

Parameter	Description
n_{ind}	Population size
$\#gen_{max}$	The maximum number of generations the GA runs for
$\#gen_{conv}$	The number of generations without improvement of the diversity after which the algorithm is stopped early
p_c	Crossover probability for each pair of individuals
α	Factor to increase the range where offspring can be randomly generated in crossover.
p_m^{ind}	Mutation probability of each individual
p_m^{var}	Mutation probability of each variable during the mutation of an individual
β	Extrapolation factor in mutation operator

Table 4.5: Overview of parameters used in the GA.

Given this knowledge, we want to find some initial individuals that are as far away from the minimum cost investment plan, but are still near-optimal. Fortunately, $\mathbf{c}_{inv}$ can be useful to find such individuals. Equation 4.11 shows how we calculate the individual $\mathbf{x}'_{inv}$ by increasing one or more investment decisions of the minimum cost individual $\mathbf{x}^*_{inv}$ such that the increase in the investment cost is equal to the allowed slack from the near optimality constraint (Equation 4.4). If the investment variables have an upper bound $\mathbf{x}^{\max}_{inv}$, we set each investment variable to its maximum value if it exceeds it.

$$\mathbf{c}_{inv}^T(\mathbf{x}'_{inv} - \mathbf{x}^*_{inv}) = \varepsilon f(\mathbf{x}^*) \tag{4.11}$$

We are certain that the resulting individual is feasible and near-optimal, as the investment variables are a capacity acting as an upper bound for the operational variables in the constraints in Equations 4.2 and 4.3. Therefore the alternative $\mathbf{x}' = \begin{pmatrix} \mathbf{x}'_{inv} \\ \mathbf{x}^*_{op} \end{pmatrix}$ will be a feasible individual with $f(\mathbf{x}') = (1+\varepsilon) f(\mathbf{x}^*)$. Note that the values of operational decision variables $\mathbf{x}^*_{op}$ might not be the operational variables with minimal cost for the given investment decisions $\mathbf{x}'_{inv}$. E.g. there might exist an $\mathbf{x}'_{op} \neq \mathbf{x}^*_{op}$ such that $f\left(\begin{pmatrix} \mathbf{x}'_{inv} \\ \mathbf{x}'_{op} \end{pmatrix}\right) \leq (1+\varepsilon) f(\mathbf{x}^*)$.

The $\mathbf{x}'_{inv}$ of each of the n_{ind} individuals of the initial population is created in the following way: The first individual is the minimum cost investment decision $\mathbf{x}^*_{inv}$, the remaining $n = n_{ind} - 1$ are then created as follows. We first create v directions $\mathbf{d}$ by increasing one of the investment variables for each individual. When $v > n$, we pick n of such directions at random. When $v < n$, we additionally generate $n - v$ random direction vectors $\mathbf{d}$ of length v such that all the elements are positive. Each direction is then used to generate a $\mathbf{x}'_{inv}$ by increasing the investment variables $\mathbf{x}^*_{inv}$ as shown in Equation 4.12. The direction is scaled by r, which uses up the remaining cost slack, before being added to $\mathbf{x}^*_{inv}$, with r calculated as in Equation 4.13. This way, an initial population of feasible individuals is created without needing to solve the original ESOM for unit commitment of each individual.

$$\mathbf{x}'_{inv} = \mathbf{x}^*_{inv} + r\mathbf{d} \tag{4.12}$$

$$r = \frac{\varepsilon f(\mathbf{x}^*)}{\mathbf{d}^T \mathbf{c}_{inv}} \tag{4.13}$$

Note that the investment variables of all the individuals of the initial population are greater than the ones of the minimum cost individual $\mathbf{x}^*_{inv}$. Since individuals with investment variables smaller than the minimum cost individual might exist, we rely on the crossover and mutation operators to fill this space.

4.5.3. Normalisation

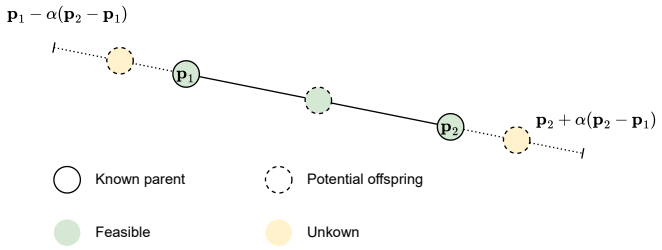
When considering the near-optimal space S_ε, the ranges where there are near-optimal individuals might differ between investment decision variables. One of the reasons this occurs might be the differences in the cost per capacity of different technologies. When a diversity metric is defined in terms of the distance between individuals , these differences can result in favouring variance for some investment variables over others, meaning possible alternatives of some investment variables are explored more than others.

This problem can be prevented by normalising the investment space. However, we do not know the minimum and maximum values for each investment variable within S_ε. Therefore, we keep track of the bounds throughout generations and update the bounds when a new near-optimal individual is found outside the bounds.

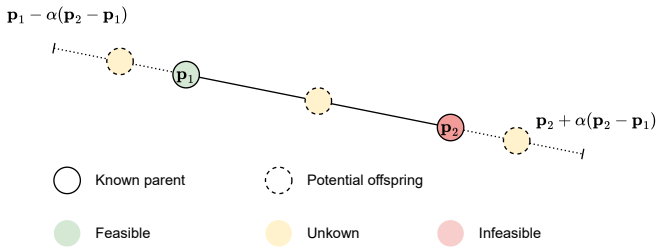
4.5.4. Crossover

Our GA approach uses the Blend-α crossover operator from literature [30]. To generate two offspring from two parents $\mathbf{p}_1$ and $\mathbf{p}_2$, we pick a random value uniformly in the interval $r \in [-\alpha, 1+\alpha]$ for each offspring $\mathbf{o}$, which is defined as a linear combination of the parents as such that $\mathbf{o} = r \cdot \mathbf{p}_1 + (1-r) \cdot \mathbf{p}_2$.

Having $r < 0$ or $r > 1$ can be interpreted as performing exploration of a potentially unexplored area. When $0 \leq r \leq 1$, the algorithm exploits already covered area, such that the offspring is a convex combination of the parents. In that case, we are certain that the offspring is feasible if both parents are feasible, due to the convexity of the near-optimal space S_ε, as can be seen in Figure 4.8a. When at least one of the parents is not feasible, we are uncertain whether the offspring is feasible, as illustrated in Figure 4.8b. The ESOM needs to be solved for unit commitment using the individual's investment decision to check its feasibility.



(a) Blend crossover with two feasible parents



(b) Blend crossover with a single infeasible parent

Figure 4.8: How blend crossover can create feasible offspring

4.5.5. Mutation

The mutation operator is applied with probability p_m^{ind} to each of the individuals in the offspring generated by the crossover operator. This operator aims to add more variation to the population. It does so by pushing the individuals away from the middle of the normalised space by multiplying the difference vector Δ of the normalised individual $\mathbf{i} \in \mathbb{R}^d$ with respect to the middle of the normalised space $[0.5]^v$ as defined in Equation 4.14. The multiplication factor $g(x)$ depends on the Euclidean distance to the centre $\|\Delta\|$.

We want individuals with an equal distance to the centre as the corners of the normalised space not to be moved (Equation 4.15) and individuals with a distance equal to individuals positioned on the boundary, closest to the centre, to be moved away from the middle with a factor of $1+\beta$ (Equation 4.16) as illustrated in Figure 4.9. Since we want individuals close to the centre to be pushed outwards relatively more, the factor is defined as an inverse of the distance of the individual to the centre, fit to the above-mentioned individuals (Equation 4.17). Using the factor $g(\|\Delta\|)$, the mutated individual $\mathbf{i}^*$ is calculated as shown in Equation 4.18. As a final step, each variable of the individual $\mathbf{i}_k, k \in 1...d$ is set to the updated variable $\mathbf{i}_k^*$ with probability p_m^{var}.

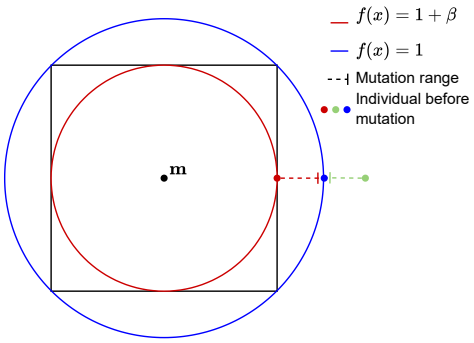


Figure 4.9: How individuals are modified by the mutation operator.

$$\Delta = \mathbf{i} - [0.5]^{\nu} \tag{4.14}$$

$$g(x) = 1 \qquad : x = \|\Delta\| = \sqrt{0.25d} \tag{4.15}$$

$$g(x) = 1 + \beta \qquad : x = \|\Delta\| = 0.5 \tag{4.16}$$

$$g(x) = 1 + \frac{a}{x} + b$$
$$a = \frac{-\beta\sqrt{0.25d}}{-\sqrt{d}+1} \tag{4.17}$$
$$b = \frac{\beta}{-\sqrt{d}+1}$$

$$\mathbf{i}^* = [0.5]^{\nu} + g(\|\Delta\|) \cdot \Delta \tag{4.18}$$

Finally, we want to make sure the mutation operator does not change the relative order of individuals such that $\|\mathbf{k} - [0.5]^{\nu}\| < \|\mathbf{l} - [0.5]^{\nu}\| \rightarrow \|\mathbf{k}^* - [0.5]^{\nu}\| < \|\mathbf{l}^* - [0.5]^{\nu}\| \forall \mathbf{k}, \mathbf{l}$. This condition is satisfied if we can make sure $\|g(\|\Delta\|)\Delta\|$ is increasing for an increasing $\|\Delta\|$. Thus, we need to pick β such that $\left[\|g(\|\Delta\|)\Delta\|\right]' > 0$. Picking $\beta < \sqrt{d} - 1$ ensures such that $f'(\|\Delta\|) > 0$ following the derivation in Equation 4.19.

$$\begin{aligned}
\left[\|g(\|\Delta\|)\Delta\|\right]' &= \left[g(\|\Delta\|)\|\Delta\|\right]' > 0 && \forall \Delta : \|\Delta\| > 0 \\
\left[g(x)x\right]' &= \left[(1 + \frac{a}{x} + b)x\right]' = [x + a + bx]' = b + 1 \\
b + 1 &> 0 \\
\frac{\beta}{-\sqrt{d}+1} + 1 &< 0 && \forall d > 1 \\
\beta &< \sqrt{d} - 1 && \forall d > 1
\end{aligned} \tag{4.19}$$

Note that the offspring generated by the crossover operator can still be outside of the normalised space, since our definition results in $g(\|\Delta\|) < 1$ for $\|\Delta\| > \sqrt{0.25d}$, individuals which are further away from the centre than the corners of the normalised space are moved closer towards the centre.

The crossover and mutation operators are crucial for exploring the space outside of the convex hull of the current population. Each of the mutated individuals needs to be checked for feasibility, even if the offspring before mutation is known to be feasible due to the convex combination of the crossover operator.

4.5.6. Selection

During selection, we want to select the subpopulation P' of size n_{ind} out of the candidate population C with maximum diversity as the new population. However, as the fitness of an individual is defined as how much it contributes to the diversity of the population, such that the fitness of an individual cannot be evaluated in

isolation, a simple selection operator — like tournament selection as explained in Section 2.3.1.3 — will not work.

Due to the combinatorial explosion, testing every possible subset of size n_{ind} and picking the one with maximum diversity is computationally infeasible. Therefore, the greedy strategy is used to iteratively remove the individual $\mathbf{i} \in C$ that contributes least to the overall diversity, i.e. with the lowest $D(C) - D(C/\{\mathbf{i}\})$ as explained in Section 3.3.2. After removing an individual, how much each remaining individual contributes to the overall diversity needs to be recomputed.

To mitigate the issue of duplicates, as explained in Section 4.2.4, we remove duplicates from the candidate population before performing selection. Since we then never have any duplicates, we do not care whether the diversity metric satisfies the uniqueness property as there will never be any duplicates.

Algorithm 2 The greedy selection operator

```
1: function SELECT(C, F; n_ind)
2:     while |C| > n_ind do
3:         C ← C − (min_{i∈C} D(C) − D(C − {i}))        ▷ Remove least diverse feasible individual
4:     end while
5:     return C
6: end function
```

4.5.6.1. Prioritizing feasible individuals

If we were to perform the selection procedure described above directly on the candidate populations, sometimes individuals that lie outside of the near-optimal space S_ε are preferred over some near-optimal individuals. For example, if we consider a population P with only near-optimal individuals and take any individual $\mathbf{i}$ on the boundary of S_ε, moving $\mathbf{i}$ further away from the middle of S_ε can only improve the diversity according to the monotonicity property following the same argument as explained in Section 4.4.1.

To ensure each population contains as many near-optimal individuals as possible, we prioritise these individuals over ones outside the near-optimal space. Considering feasible set $F \subseteq C$ and infeasible set $I \subseteq C$ such that $|F| + |I| = |C| = 2n$ and $F \cap I = \varnothing$, the selection procedure works as follows:

- If $|F| > n$, discard individuals in I and continue with selection as described above.
- If $|F| \leq n$, add all feasible individuals to P' and iteratively add infeasible individuals that least contribute to the diversity of the selected set so far $P' = \min_{i \in I} D(P' \cup \{\mathbf{i}\}) - D(P')$.

The reason the infeasible individuals, which contribute least to the diversity, are selected when $|F| \leq n$ is that we want to select the ones closest to being feasible. However, this can only happen in the first few generations if not all individuals of the initial population are feasible. Since the previous population is always part of the candidate population, this never occurs once a population consists of only feasible individuals. The updated selection operator is defined in Algorithm 3.

4.5.6.2. Evaluation speedup

To select the new population P' out of $2n$ individuals in C, we need to iteratively remove n individuals that contribute least to the overall diversity. Since diversity is defined over the full population, we must recompute each individual's contribution to the diversity after each removal.

Considering the diversity metrics that sum over all (fractional) pairwise distances, computing $D(P)$ would be $O(n^2)$. Recomputing the diversity $D(C/\{\mathbf{i}\})$ — in line 3 of Algorithm 2 and 8 of Algorithm 3 — at least n times for all remaining individuals i results in $O(n^3)$. However, since the removal of an individual only affects the sum of pairwise (fractional) distances considering the removed individual, we can reduce the complexity of recomputing $D(C/\{\mathbf{i}\})$.

Comparing $D(C) - D(C/\{\mathbf{i}\})$ for each individual can be reduced to comparing the sum of the (fractional) pairwise distances to all other individuals — as shown in the derivation of Equations 4.20–4.25 for the mean distance and harmonic mean distance diversity metrics. Therefore, we keep track of the sum of the (fractional) distances to all the other individuals and subtract the (fractional) distance of the removed individual. This reduces the calculation of $D(C/\{\mathbf{i}\})$ after each update to $O(n)$ for every remaining individual, resulting in a complexity of $O(n^2)$ for every update. Finally, as n individuals need to be removed from C, the time complexity of the full selection procedure is $O(n^3)$ in time and $O(n^2)$ in space since we need to keep track of all

Algorithm 3 The selection operator

function SELECT($C, F;\ n_{ind}$)
  $F := \{b_1, ..., b_{|C|}\}$
  $b_j \in \{TRUE, FALSE\}$
  $P \leftarrow \{\mathbf{i}_j | \mathbf{i}_j \in C,\ b_j\}$
  $I \leftarrow C - P$ ▷ All infeasible candidate individuals
  if $|P| > n_{ind}$ **then**
    while $|P| > n_{ind}$ **do**
      $P \leftarrow P - (\min_{\mathbf{i} \in P} D(P) - D(P - \{\mathbf{i}\}))$ ▷ Remove least diverse feasible individual
    end while
  else
    while $|P| < n_{ind}$ **do**
      $\mathbf{i} \leftarrow \min_{\mathbf{i} \in I} D(P + \{\mathbf{i}\}) - D(P)$
      $P \leftarrow P \cup \{\mathbf{i}\}$ ▷ Add least diverse infeasible individual
      $I \leftarrow I - \{\mathbf{i}\}$
    end while
  end if
  return P
end function

the pairwise distances.

$$\min_{i \in C} D(C) - D(C/\{\mathbf{i}\}) \tag{4.20}$$

$$= \max_{i \in C} D(C/\{\mathbf{i}\}) \tag{4.21}$$

$$= \max_{i \in C} \frac{(|C|-1)(|C|-2)/2}{\sum_{k,l \in C: k \neq l \neq i} \frac{1}{d(k,l)}} \qquad = \max_{i \in C} \frac{\sum_{k,l \in C: k \neq l \neq i} d(k,l)}{(|C|-1)(|C|-2)/2} \tag{4.22}$$

$$= \min_{i \in C} \sum_{k,l \in C/\{\mathbf{i}\}: k \neq l} \frac{1}{d(k,l)} \qquad = \max_{i \in C} \sum_{k,l \in C: k \neq l \neq i} d(k,l) \tag{4.23}$$

$$= \min_{i \in C} \sum_{k,l \in C: k \neq l} \frac{1}{d(k,l)} - \sum_{k \in C: k \neq i} \frac{1}{d(k,i)} \qquad = \max_{i \in C} \sum_{k,l \in C: k \neq l} d(k,l) - \sum_{k \in C: k \neq i} d(k,i) \tag{4.24}$$

$$= \max_{i \in C} \sum_{k \in C: k \neq i} \frac{1}{d(k,i)} \qquad = \min_{i \in C} \sum_{k \in C: k \neq i} d(k,i) \tag{4.25}$$

4.5.7. Early stopping

When diversity stops improving over generations, we would like to stop our GA loop. However, due to the random nature of GAs, it is possible that the diversity slightly worsens between generations. Therefore, we use a convergence criterion, tracking the number of subsequent generations that have not improved the diversity. The GA loop is exited if the number of generations exceeds a set parameter $\#gen_{conv}$. Generally, the value of $\#gen_{conv}$ is chosen based on a maximum computational budget to avoid stopping the GA before it converges.

When the normalisation bounds increase by a newly found individual, the space in which we aim to find the most diverse population changes. In this increased space, a population which had a lower diversity than the most diverse one found so far before updating the normalisation bounds may be more diverse. Therefore, we reset the best diversity found so far and the count of generations without diversity improvement. This reset aims to give the GA sufficient attempts to optimise the diversity in the updated space.

4.5.8. Component comparison

To make sure the GA components described above improve the performance of the GA, we start with a base case and include each component one by one in the GA setup and test whether the performance of the GA improves. The base case is a GA as simple as possible that could solve the problem given sufficient time. Then, the components listed in Table 4.6 are added from top to bottom, resulting in five setups to compare, where the final one includes all the components. All setups use the harmonic mean distance metric and assume the

minimum cost individual is known.

The base case is a GA that initialises the population using a Gaussian distribution centred around the minimum cost individual with $\sigma = 0.1$ for each investment variable. For creating offspring, it copies the population and mutates each copy with a Gaussian distribution centred around the point with $\sigma = 1$. The choices of σ are relatively low, as there is no information available on the size of the near-optimal space S_ε. We prefer to be conservative to make sure we gradually improve the diversity of each generation.

4.5.8.1. ESOM showcase

Each approach will be run using an ESOM showcase designed to easily scale the dimensionality of the problem. The model includes the three technologies for electricity generation at every node, being wind, PV and gas. To compare the components, we will use the ESOM showcase with a single node, resulting in a total of three investment variables. However, the number of nodes can easily be varied up to seven nodes, which will be used in the performance evaluation of the GA described in Section 4.6.

The load factors of wind and PV at each node are calculated using data from the year 2022 of Austria, Belgium, Germany, the Netherlands, France, Spain and Italy with time steps of one hour. The load factor is calculated as the fraction of the total generation capacity being generated at each timestep.

As the ESOM does not include transmission line capacities, the model can be scaled up to 21 investment variables, one for each combination of technology and country.

Component	**Setup**	
	Base	*Tailored component*
Crossover	None	Blend crossover
Normalisation	None	Near-optimal min/max
Initialisation	Gaussian around optimum	Slack increments
Mutation	Gaussian permutation	Push outwards

Table 4.6: How adding each component modifies the base case.

Name	Value
n_{ind}	100
$\#gen_{max}$	1000
$\#gen_{conv}$	10
p_c	0.9
α	0.5
$p_{m.ind}$	0.1
$p_{m.var}$	0.9
β	0.5

Table 4.7: The parameter values used in the component comparison

To compare the performance of each setup, we keep track of the following criteria:

- The generation where the population's diversity is equal to the diversity of the output of other MGA methods, being the Modelling-All-Alternatives (MAA) method and a weighted sum MGA method that first iteratively optimises in the minimum and maximum direction of each investment variable independently — generating $2v$ alternatives — after which it searches in random directions until it found the desired amount of alternatives.

  We expect that the final GA setup, including all the components, will reach a higher diversity than the other MGA methods. Therefore, we expect that adding each component should not result in the diversity of the other methods being reached at a later generation.

- We track the convergence speed using three convergence criteria measured in normalised space, explained here from most lenient to strictest. The first criterion tracks the first generation where the diversity increase is lower than some tolerance $\Delta = 10^{-4}$. The second criterion tracks the first generation with a diversity increase lower than the same tolerance compared to ten generations earlier. The

final criterion is the early stopping criterion used in the GA loop as explained in Section 4.5.7, being that the diversity has not improved for ten generations.

Keeping track of these different convergence criteria allows us to track how fast the different setups reach certain stages of convergence, where reaching the lenient criterion indicates faster exploration early on in the optimisation process, and the strict criteria indicate how long it takes to reach optimal diversity. We expect that adding each component will not reduce the speed of convergence, meaning the generation of convergence will not increase.

- For each of the convergence criteria, we track the diversity of the first generation satisfying the criteria. This allows for comparison of the diversity of different setups at different stages of convergence.

  We expect adding each component will not reduce the diversity at each stage of convergence. However, it could be that for a convergence criterion, adding a component increases the convergence speed, but reduces the diversity at the moment of convergence. We expect such a situation to be more likely to occur for the lenient convergence criteria, since adding a component could improve the early exploration capabilities of the GA, reducing the number of generations needed to reach a point after which the diversity increases at a lower rate, potentially meeting one of the convergence criteria earlier.

  When the diversity differs a lot between different convergence criteria for a single setup, it shows that the more lenient criterion is less suited for that specific setup in practice. However, when the difference is low, using the stricter convergence criteria would mean more computational resources are used for relatively little improvement in diversity.

- To compare the distribution of the population to the distribution of the alternatives found by the MAA method and the convex hull of S_ε used by the MAA method, we estimate the KL-divergence of the population and the MAA output and of the population and the aforementioned convex hull according to [28].

  The difference between the KL-divergence values at different convergence criteria helps us to reason whether a setup has sufficiently converged. As the population manages to find all borders of S_ε throughout the optimisation process, we expect the difference in the KL-divergence to the convex hull to reduce more than the KL-divergence to the output alternatives, meaning the difference in KL-divergences is expected to reduce as the population becomes more diverse.

  Comparing the two values of the KL-divergence serves as an indicator of when the dimensionality of the ESOM is too high with the population size, as we then expect the population to be biased towards the border of S_ε and therefore have a lower KL-divergence to the convex hull used by the MAA method.

- Finally, we keep track of how many individuals need to be solved for unit commitment to check whether they are near-optimal. How many individuals turn out not near-optimal is also tracked, such that we can calculate what percentage of all generated individuals need to be checked and turn out not near-optimal.

  Assuming a parent population of only near-optimal individuals, we would expect a fraction of 0.55 of the offspring created by the crossover to be near-optimal, as shown in Equation 4.26 using the parameters in Table 4.7. Therefore, we expect the setups with only crossover to need to check about $100\% - 55\% = 45\%$ of the individuals for near-optimality.

$$(1 - p_c) + p_c * \frac{1}{1+2\alpha} = 0.1 + 0.9 * \frac{1}{2} = 0.55 \tag{4.26}$$

  Of these individuals, a fraction of $p_{m.ind}$ are mutated and need to be checked for near-optimality, meaning a fraction of $0.55 * (1 - p_{m.ind}) = 0.495$ are expected to be near-optimal, meaning we expect setups with the mutation operator to need to check $100\% - 49.5\% = 50.5\%$ of the individuals for near-optimality.

  These percentages again give insight into the exploration and exploitation trade-off of the GA and computational cost of the method. A higher percentage of non-near-optimal individuals indicates more exploration, which in our application means more effort is being put into the exploration of the boundary of S_ε. A lower percentage of individuals needed to be checked indicates more exploitation — which means more individuals are created as a convex combination of feasible parents in the crossover operator — indicating more effort is being put in improving diversity within the already explored boundaries.

Approach	Initialisation	Inner point generation	Assumptions		
			Optimum known	*Boundary optimisation*	*Convex hull*
MAA sampling	MGA	MAA	✓	✓	✓
Weighted sum MGA	MGA	-	✓	✓	-
Boundary initialisation	MGA	GA	✓	✓	-
Blind GA	Heuristic	GA	✓	-	-

Table 4.9: Overview of approaches for generating alternatives compared in performance evaluation.

Each of the criteria is measured considering the population in the unnormalised space and normalised with respect to the minimum and maximum values found for each investment variable of the hull used in the MAA method. Keeping track of both allows us to gain insights into the impact of the normalisation component. Finally, we run each setup five times with different seeds to avoid the impact of the inherent randomness of the GA method.

4.6. Performance evaluation

To evaluate the performance of the approach defined above, it is compared to the MAA and a weighted sum MGA method from the literature.

4.6.1. Baseline methods

Four different approaches of MGA methods, stated in Table 4.9, will be used. Each approach differs in its initialisation (or pre-processing) and individual generation method. For the initialisation method, they are either able to compute boundary points using the random weighted sum MGA method, or can only use information from the ESOM in a heuristic fashion, without optimisation. Regarding the inner point generation, the approaches either use the sampling method defined in the MAA approach or optimise the diversity of the inner points using the GA approach discussed above.

The four approaches have different computational requirements. The MAA sampling approach requires being able to compute the convex hull, which greatly limits the dimensionality of the problem, as described in Section 3.1.1. The 'weighted sum MGA' and 'boundary initialisation' approach require the ability to find boundary points of the near-optimal space using the random weighted sum MGA method. This is the same pre-computation used in the 'MAA sampling' approach to find the convex hull. Finally, the 'Blind GA' approach only optimises the full ESOM to find the minimum cost individual. The approach then only uses the ESOM to optimise for unit commitment, given the investment variables, to check satisfiability of an individual, and not to generate new individuals.

Intuitively, the 'MAA sampling' approach serves as a baseline for the quality of the results, as this approach uniformly samples from the near-optimal space. Then, the 'boundary initialisation' approach enables us to compare diversity optimisation to the sampling done in the 'MAA sampling' approach, given the same information from the boundary points, but without being limited in dimensionality by relying on computing the convex hull. The approach represents the exploitation ability of the GA to maximise diversity, having all initial points on the boundary of the near-optimal space.

Finally, the 'blind GA' approach serves as an example to show how well the new approach considered in this work can cover all parts of the near-optimal space without knowing any boundary points beforehand. Compared to the 'boundary initialisation' approach, the 'blind GA' approach will have a less diverse initial population as it uses less information about the near-optimal space.

4.6.2. Experiment 1: Comparing result quality throughout execution

The first experiment will measure the diversity of the considered population. For the 'MAA sampling' approach, this means that the diversity of the population is measured after adding a newly sampled point to it. For the 'Boundary initialisation' and 'Blind GA' approaches, the diversity of the population is measured after every feasibility check, and the diversity of all feasible individuals found so far is measured after adding a new near-optimal individual. As the 'MAA sampling' approach is limited in the number of investment variables of the ESOM, this experiment's comparison aims to show how similar the individual quality of our approach's

output is to the uniformly sampled MAA output, when applicable.

We expect this experiment to show that the diversity of the current population is higher than the value of the 'MAA sampling' approach for the same population size. This follows the reasoning from Section 4.2.3.1, stating that the Sobol sequence and uniform grid in unit square space are expected to be more diverse than the multivariate uniform sample. Intuitively, this results from the random nature of the sampling that sometimes produces individuals that are relatively close to each other. This means, for increasing population size and sample size, we expect the difference between the diversity of the final population and the sampled population of the same size to converge towards zero. Furthermore, the diversity of the final population of the 'Blind GA' approach should have a worse initial diversity and converge later compared to the diversity of the population of the 'Boundary initialisation' approach. In addition, the measures of any of the approaches should not improve the diversity compared to the 'MAA sampling' approach. In this experiment, the 'weighted sum MGA' approach is not relevant to compare, as the resulting population is the same as the initial population of the 'Boundary initialisation' approach.

4.6.3. Experiment 2: Comparing result quality for different dimensionality

The next experiment aims to show how the problem's dimensionality influences the output quality. For every approach, the diversity of the population after initialisation and at the three convergence criteria discussed in Section 4.5.8 is measured for a population of 50, 100 and 300 individuals in normalised space, varying the dimensionality of the problem on which it is being applied. As the 'MAA sampling' approach has only been applied up to 10 dimensions due to computational limitations [27], it will not show the expected change in result quality for higher dimensions.

When the dimensionality of a normalised space increases, we expect the maximum diversity to improve following the monotonicity property since we expect the pairwise distances to be larger when considering the same number of points. Therefore, for any approach and output size, the result quality should improve when the problem's dimensionality increases. For a similar reason, the maximum diversity of a higher number of points within the same dimensionality is expected to be lower for a given approach. Generally, the output diversity of different approaches with the same output size should have similar values, where the diversity of the 'Blind GA' approach is at most as high as that of the 'Boundary initialisation' approach.

4.6.4. Experiment 3: Comparing scalability

To show how well our approach scales compared to weighted sum MGA methods, we will track the computational cost per generated individual. This will be done by running both approaches using models of increasing dimensions. The GA approaches will then be run, measuring the time they take to reach the three convergence criteria as mentioned above. For the GA approaches, we will consider the computational cost per individual of the population at the different moments of convergence and the computational cost per all individuals found throughout the generations until the moment of convergence. Finally, the spatial and temporal resolution of the model will be independently varied and separately compared.

This experiment is aimed at showing how the computational benefit of only solving the ESOM for unit commitment relates to the higher number of solves needed by the GA approaches compared to the 'weighted sum MGA' approach. Overall, we expect the computational cost of the GA approaches to be in the same order of magnitude as the 'weighted sum MGA' approach. We expect that the extra computational cost of a higher number of model solves required by the GA approaches is sufficiently mitigated by each solve having a lower computational cost as they solve the model with given investment variables. Finally, the computational cost per alternative when considering all the alternatives found throughout the generations is expected to always be lower than the cost per alternative of the MGA method.

When the population size is relatively low compared to the number of investment variables of the ESOM — as described in Section 4.4 — we expect all the individuals of the most diverse population to lie on the boundary of the near-optimal space. In this case, we expect the diversity of the 'weighted sum MGA' approach to be reached quickly by the GA model and to improve relatively little over it, meaning the convergence criteria are satisfied earlier in the optimisation process. Then, the lower number of generations would mean a lower computation cost per alternative.

4.7. Generating alternatives of the SECURES model

After testing this work's GA on the ESOM showcase in previous experiments, we also test it on the ESOM of the SECURES model discussed in Section 2.1.2. Due to time limits, we only run our GA on six connected countries — as shown in Figure 4.10 — resulting in a total of 51 investment variables.

To give insights into how well the resulting alternatives cover different options in the investment plans, the capacities per technology are summed for all the countries. Then, we compare the range of values of the alternatives to the minimum cost investment plan to get an idea of how well the space is covered.

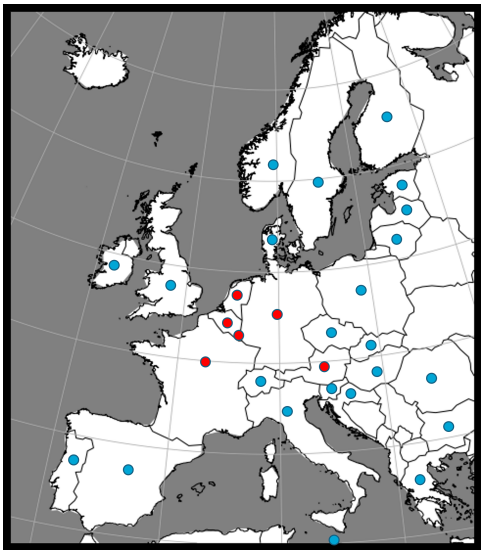


Figure 4.10: Map of countries used in SECURES experiment. Red dots indicate the included countries.

5
Results

This chapter first discusses the results of the process to pick the most suitable diversity metric, resulting in a choice of diversity metric used in the remainder of this work. Then, the results of testing the components of our GA approach are discussed, showing whether the components improve the performance of the GA. Next, we discuss the performance evaluation of our proposed GA approach compared to other MGA methods in terms of output quality and scalability. Finally, we show the output of applying the GA approach to the European power system and how it relates to our expectations.

5.1. Selecting the diversity metric

The hypotheses of the preferred populations discussed in Section 4.2 are tested using one diversity metric from each type of diversity metric. The results for the three diversity metrics are presented in Table 5.1. Results for all diversity metrics considered are shown in Table A.2.

5.1.1. Testing populations in unit square spaces

The first experiment compares how different types of diversity metrics distinguish more and less diverse populations in unit square space.

5.1.1.1. Sum of distances metrics

Considering the uniform populations, we see that the multivariate uniform population $P_{mv.u}$ is more diverse than the Sobol sequence population P_s. Even though the difference is small, we want the metric to prefer P_s as it contains fewer individuals that are relatively close to each other.

The expectations regarding the Sobol sequence populations with modified concentration were inconsistent, where the Sobol sequence with blind spots $P_{s.b}$ improved over P_s. Combined with the fact that the Sobol sequence with vertical separation $P_{s.sep}$ is more diverse than all the other populations, it indicates that the diversity metric prefers populations with individuals closer to the boundary of the space, even if it results in less uniform coverage of the space. This preference will result in a bias towards the boundary of the space when used in the optimisation process of the GA.

5.1.1.2. Sum of fractional distances

Overall, the harmonic mean distance metric agrees very well with our expectations. Only the expectations of the Sobol sequence with concentrated spots population $P_{s.c}$ show some inconsistencies with the covering radius distribution property. Intuitively, this inconsistency comes from the fact that the harmonic mean has more weight on low values compared to the mean. This means that relatively closer-together individuals have a higher impact on diversity.

Such an increased impact of small distances between points will result in the optimisation process favouring populations with individuals that are farther apart, hopefully preventing a bias towards the boundary of the space. The comparison of the $P_{s.sep}$ shows how a more even coverage throughout the space is preferred to individuals closer to the boundary of the space.

5.1.1.3. Discrepancy metric

As the Sobol sequence is designed to minimise discrepancy, it turns out to be the most diverse population according to the metric. The reason why the metric is inconsistent with the expectations of the uniform grid population P_g is that the regular spaces between the individuals align with the axes such that the discrepancy is large in those areas (see Section 3.2.1 for clarification on the discrepancy metric). The same reasoning can be used for the inconsistency with the expectation of comparing $P_{s.sep}$ to the multivariate normal sample population $P_{mv.n}$.

Type	**Population**	$D(P)$	P_s	P_g	$P_{mv.u}$	$P_{s.b}$	$P_{s.c}$	$P_{s.sep}$	$P_{s.cen}$	$P_{mv.n}$
Sum of distances – Mean distance										
Uniform	P_s	0.523	=	(<)	(>)	>	>	>	>	>
	P_g	0.557	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.524	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.528	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	0.514	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.605**	<	(<)	>	>	>	=	>	(>)
	$P_{s.cen}$	0.262	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.369	<	<	<	(<)	(<)	(<)	>	=
Sum of fractional distances – Harmonic mean distance [1]										
Uniform	P_s	0.360	=	(<)	(>)	>	>	>	>	>
	P_g	**0.388**	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.339	(<)	(<)	=	(<)	(<)	>	>	>
Concentration	$P_{s.b}$	0.358	<	(<)	(>)	=	(<)	>	>	(>)
	$P_{s.c}$	0.331	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	0.335	<	(<)	<	<	>	=	>	(>)
	$P_{s.cen}$	0.180	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.234	<	<	<	(<)	(<)	(<)	>	=
Discrepancy – Wrap-around discrepancy[1]										
Uniform	P_s	**0.000**	=	(<)	(>)	>	>	>	>	>
	P_g	0.004	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.002	(<)	(<)	=	(<)	(<)	>	>	>
Concentration	$P_{s.b}$	0.001	<	(<)	(>)	=	(<)	>	>	(>)
	$P_{s.c}$	0.002	<	(<)	(>)	(>)	=	>	>	(>)
Bias to border	$P_{s.sep}$	0.056	<	(<)	<	<	<	=	>	(>)
	$P_{s.cen}$	0.113	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.045	<	<	<	(<)	(<)	(<)	>	=

Table 5.1: Whether our expectations of how each pair of test populations perform against each other in terms of diversity in unit square space are correct. '>' indicates $D(P_{row}) > D(P_{column})$, and '<' indicates the reverse. Entries marked with '(>)' or '(<)' follow from the covering radius distribution property. The green coloured cells are correct, the red ones are incorrect, and the yellow ones are the relations defined using the output diversities. The bold diversity value indicates the highest value.

5.1.2. Convex space

Testing the populations modified to be in a convex subspace of the unit square space — as shown in Appendix A.2.2 — showed the same results for the diversity metrics based on Table 4.3.

5.1.3. Takeaways

The diversity metrics based on the sum of fractional distances are the best fit for optimisation and will therefore be used in further experiments. Even though multiple options of diversity metrics based on the sum of fractional distances would be viable to be used for optimisation, the remainder of this work uses the harmonic mean distance. This comes from the fact that the metric has been shown to be one of the metrics resulting in the least biased optimisation when used in optimisation, is more sensitive to changes in populations due

[1] A lower value of this metric signifies a more diverse population

to not using minimum or maximum distance, and is easy to understand as its value can be understood as a distance itself.

This is in line with the work of [22], which mentions that the Energy metric — which is based on a sum of fractional distances — can be safely used as a target for optimisation.

5.2. GA approach setup

Results of the experiments explained in Section 4.5.8 are listed and discussed next.

5.2.1. Speed of convergence

Firstly, the generations at which each diversity comparison or convergence criteria explained in Section 4.5.8 is satisfied are listed in Table 5.2. Each column corresponds to reaching the diversity of the MAA method, the weighted sum MGA method — either in the investment (unnormalised) space or normalised space — or one of the convergence criteria.

Setup	**Convergence generation of diversity comparison or covergence criteria**						
	Investment space		*Normalised space*				
	MAA diversity	MGA diversity	MAA diversity	MGA diversity	Lenient convergence	Moderate convergence	Strict convergence
Base	337 ± 6.9	466 ± 8.0	256 ± 6.6	467 ± 6.5	469 ± 155.0	807 ± 0.0	1000 ± 0.0
Crossover	46 ± 2.3	50 ± 2.9	46 ± 1.7	79 ± 38.3	$\mathbf{1 \pm 0.0}$	$\mathbf{116 \pm 25.2}$	$\mathbf{237 \pm 45.6}$
Normalisation	48 ± 2.4	53 ± 1.3	48 ± 2.0	60 ± 1.3	$\mathbf{1 \pm 0.0}$	178 ± 36.2	266 ± 21.6
Initialisation	8 ± 0.5	12 ± 1.5	3 ± 0.5	11 ± 0.7	47 ± 7.3	$\mathbf{137 \pm 37.9}$	$\mathbf{208 \pm 35.3}$
Mutation	$\mathbf{5 \pm 0.5}$	$\mathbf{9 \pm 0.4}$	$\mathbf{2 \pm 0.4}$	$\mathbf{8 \pm 0.8}$	45 ± 5.0	$\mathbf{152 \pm 28.0}$	$\mathbf{220 \pm 57.1}$

Table 5.2: Overview of the first generations where each diversity comparison or convergence criteria is satisfied. The results are averaged over 5 runs with distinct seeds, in the format of mean ± standard deviation.

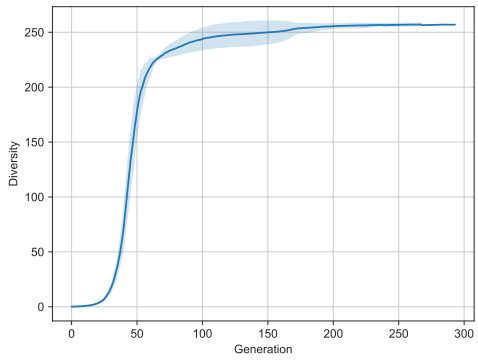
In Table 5.2, we see that adding components lowers the number of generations needed to reach the diversity of the MAA method and the weighted sum MGA method. The same does not hold for the convergence criteria, as it remains inconclusive whether they are satisfied earlier.

Without the initialisation component, the lenient convergence criterion is already satisfied after the first generation. As the initial population is sampled from a Gaussian distribution around the minimum cost individual, each pair of individuals might not be close enough for the crossover operator to generate an individual outside of the tight convex hull containing the current population. Therefore, the diversity increases very little in the first generations, as the space is already covered quite uniformly, and its size increases slowly.

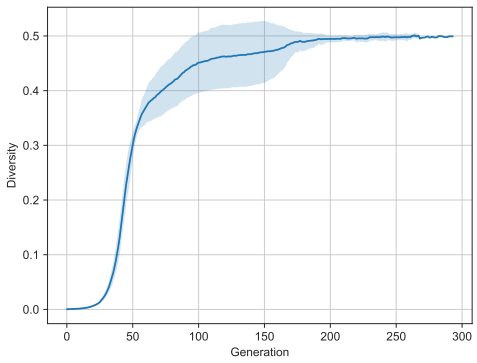
Next to that, adding the normalisation component slowed down the convergence to the moderate and strict convergence criteria. A possible explanation could be that without normalisation, the diversity in normalised space fluctuates more, such that the moderate convergence criterion is met too early. Figures 5.1 and 5.2 confirm this suspicion, as Figure 5.1b shows more fluctuations around the moments of convergence of the moderate and strict criteria.

5.2.2. Diversity at convergence

Table 5.4 shows the diversity of the population at the points where the diversity comparison is satisfied, or the convergence criterion is met, similarly to Table 5.2.

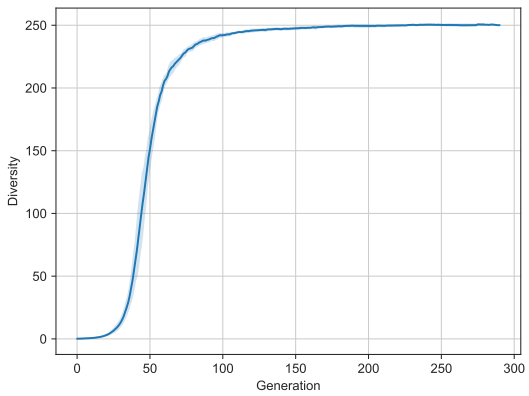


(a) Diversity over generation in unnormalised investment space.

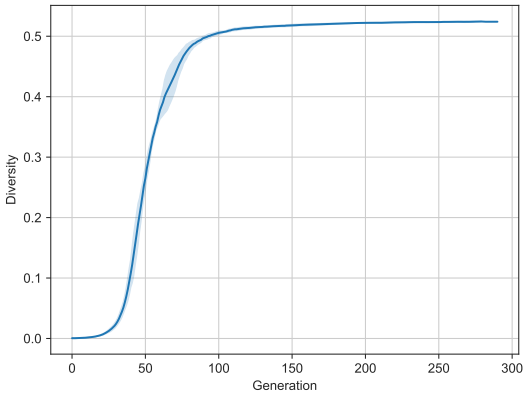


(b) Diversity over generations in normalised space.

Figure 5.1: The diversity over generation of the crossover setup of Table 4.6 averaged over 5 runs with distinct seeds.



(a) Diversity over generation in unnormalised investment space.



(b) Diversity over generations in normalised space.

Figure 5.2: The diversity over generation of the normalisation setup of Table 4.6 averaged over 5 runs with distinct seeds.

Setup	**Diversity at convergence generation**						
	Investment space		*Normalised space*				
	MAA diversity	MGA diversity	MAA norm. diversity	MGA norm. diversity	Lenient convergence	Moderate convergence	Strict convergence
Base	132.4 ± 0.1	170.7 ± 0.1	104.5 ± 1.5	170.9 ± 1.5	163.9 ± 45.6	212.1 ± 0.0	228.1 ± 1.0
Crossover	$\mathbf{136.7 \pm 4.5}$	$\mathbf{176.8 \pm 2.9}$	$\mathbf{138.2 \pm 10.6}$	$\mathbf{217.3 \pm 21.4}$	0.1 ± 0.0	$\mathbf{247.4 \pm 12.1}$	$\mathbf{256.4 \pm 1.3}$
Normalisation	$\mathbf{135.1 \pm 2.5}$	$\mathbf{175.3 \pm 3.3}$	$\mathbf{137.9 \pm 5.4}$	$\mathbf{204.6 \pm 5.7}$	0.2 ± 0.0	$\mathbf{248.9 \pm 1.9}$	250.4 ± 0.6
Initialisation	$\mathbf{139.4 \pm 2.2}$	$\mathbf{173.7 \pm 2.1}$	89.8 ± 1.3	165.6 ± 3.7	$\mathbf{236.9 \pm 2.4}$	$\mathbf{247.6 \pm 2.1}$	249.6 ± 0.7
Mutation	$\mathbf{141.2 \pm 7.2}$	$\mathbf{174.2 \pm 2.8}$	97.6 ± 5.4	169.0 ± 6.4	$\mathbf{239.7 \pm 2.7}$	$\mathbf{248.6 \pm 1.4}$	249.7 ± 1.7
Diversity measured in normalised space							
Base	0.29 ± 0.00	0.37 ± 0.00	0.23 ± 0.00	0.37 ± 0.00	0.35 ± 0.10	0.45 ± 0.00	0.48 ± 0.00
Crossover	0.24 ± 0.02	0.30 ± 0.03	$\mathbf{0.24 \pm 0.01}$	0.37 ± 0.00	0.00 ± 0.00	0.46 ± 0.06	0.50 ± 0.00
Normalisation	0.24 ± 0.01	0.31 ± 0.01	$\mathbf{0.24 \pm 0.01}$	0.38 ± 0.01	0.00 ± 0.00	$\mathbf{0.52 \pm 0.00}$	$\mathbf{0.52 \pm 0.00}$
Initialisation	$\mathbf{0.33 \pm 0.00}$	$\mathbf{0.39 \pm 0.00}$	$\mathbf{0.24 \pm 0.00}$	0.38 ± 0.01	$\mathbf{0.50 \pm 0.00}$	$\mathbf{0.52 \pm 0.00}$	$\mathbf{0.52 \pm 0.00}$
Mutation	$\mathbf{0.33 \pm 0.01}$	$\mathbf{0.39 \pm 0.00}$	$\mathbf{0.25 \pm 0.01}$	0.38 ± 0.01	$\mathbf{0.50 \pm 0.00}$	$\mathbf{0.52 \pm 0.00}$	$\mathbf{0.52 \pm 0.00}$

Table 5.4: Overview of the diversity of the population at the first generations where each diversity comparison or convergence criteria is satisfied. The results are averaged over 5 runs with distinct seeds, in the format of mean ± standard deviation.

Method	Space	
	Investment	Normalised
MAA	170.6	0.36
MGA	132.2	0.23

Table 5.5: Diversity of the MAA and weighted sum MGA output used in component comparison.

In the investment space, the diversity is close to the diversity of the output of both the MAA and weighted sum MGA method, which are listed in Table 5.5. Considering the normalised diversity, adding the initialisation component helps to improve the normalised diversity when the unnormalised diversity of the other MGA methods is reached. At the same time, when the normalised diversity of the other MGA methods is reached, we notice that adding the initialisation component resulted in a lower unnormalised diversity. A possible explanation could be the fact that the initial population is much more diverse in terms of unnormalised diversity, which, combined with the fact that the diversity is reached early on, results in a lower unnormalised diversity.

When looking at the convergence criteria, adding a component only reduced the diversity for the strict convergence criterion, which can again be attributed to faster convergence. However, after the normalisation component is added, we see that the difference in unnormalised diversity for different convergence criteria is low, indicating the stricter criteria to be less useful in practice.

5.2.3. KL-divergence at convergence

Table 5.6 and 5.7 report the KL-divergence to the MAA method's output and the convex hull used by the MAA method. The 'Strict convergence all' column was added to Table 5.7, stating the KL-divergences of all the alternatives found throughout the optimisation process.

Setup	KL divergence	KL-divergence at convergence generation			
		Investment space		*Normalised space*	
		MAA diversity	MGA diversity	MAA norm. diversity	MGA norm. diversity
Base	MAA result	**0.68 ± 0.06**	**0.26 ± 0.16**	**1.44 ± 0.10**	**0.26 ± 0.17**
	MAA hull	7.51 ± 0.11	6.10 ± 0.08	8.26 ± 0.05	6.09 ± 0.08
Crossover	MAA result	**1.16 ± 0.27**	**0.63 ± 0.28**	**1.16 ± 0.26**	**0.70 ± 0.37**
	MAA hull	6.84 ± 0.40	5.30 ± 0.45	6.80 ± 0.50	3.49 ± 1.35
Normalisation	MAA result	**1.08 ± 0.30**	**0.67 ± 0.27**	**1.02 ± 0.21**	**0.39 ± 0.23**
	MAA hull	6.55 ± 0.43	5.44 ± 0.49	6.48 ± 0.54	4.38 ± 0.59
Initialisation	MAA result	**0.72 ± 0.15**	**0.29 ± 0.21**	**1.65 ± 0.03**	**0.38 ± 0.23**
	MAA hull	5.71 ± 0.21	4.53 ± 0.18	7.02 ± 0.06	4.75 ± 0.20
Mutation	MAA result	**0.32 ± 0.27**	**−0.07 ± 0.14**	**1.22 ± 0.15**	**−0.09 ± 0.14**
	MAA hull	5.74 ± 0.42	4.40 ± 0.39	6.83 ± 0.15	4.57 ± 0.29
Normalised space					
Base	MAA result	**0.99 ± 0.08**	**1.08 ± 0.08**	**1.63 ± 0.12**	**1.09 ± 0.07**
	MAA hull	7.57 ± 0.13	6.04 ± 0.12	8.36 ± 0.03	6.02 ± 0.14
Crossover	MAA result	**2.71 ± 0.66**	**2.27 ± 0.86**	**2.71 ± 0.51**	**1.90 ± 0.76**
	MAA hull	7.24 ± 0.46	5.95 ± 0.37	7.18 ± 0.53	4.37 ± 1.15
Normalisation	MAA result	**2.70 ± 0.37**	**2.00 ± 0.52**	**2.68 ± 0.35**	**1.36 ± 0.47**
	MAA hull	7.09 ± 0.28	6.16 ± 0.48	7.02 ± 0.40	5.18 ± 0.61
Initialisation	MAA result	**1.27 ± 0.11**	**0.73 ± 0.19**	**2.03 ± 0.07**	**0.85 ± 0.28**
	MAA hull	5.88 ± 0.16	4.83 ± 0.29	6.92 ± 0.04	5.03 ± 0.17
Mutation	MAA result	**0.88 ± 0.19**	**0.43 ± 0.04**	**1.72 ± 0.14**	**0.49 ± 0.07**
	MAA hull	5.99 ± 0.39	4.75 ± 0.38	6.83 ± 0.04	4.90 ± 0.32

Table 5.6: Overview of the KL-divergence of the population to the output and hull of the MAA method at the first generations where each diversity comparison is satisfied. The results are averaged over 5 runs with distinct seeds, in the format of mean ± standard deviation.

In general, we see that the KL-divergence to the MAA result is generally higher at the generations when the diversity surpasses the diversity of the other MGA methods. This is as expected, since we discussed that the diversity of the other MGA methods is reached quite fast in Section 5.2.1, which means borders of the space are relatively unexplored, resulting in a higher KL-divergence to the hull.

As components are added, we do see a lower KL-divergence to the MAA result. However, as it gets lower for reaching both the diversity of the MAA method and the weighted MGA method, it shows that comparing when the optimisation surpasses the diversity of the output of other MGA methods, it is not a very useful indication of whether the population is biased towards the boundary.

Setup	**KL divergence**	**KL-divergence at convergence generation**			
		Normalised space			
		Lenient convergence	Moderate convergence	Strict convergence	Strict convergence all
Base	MAA result	**0.66 ± 0.83**	**0.35 ± 0.00**	**0.29 ± 0.07**	**3.67 ± 0.07**
	MAA hull	6.19 ± 1.44	3.99 ± 0.00	2.91 ± 0.27	8.49 ± 0.32
Crossover	MAA result	**4.51 ± 0.00**	**0.70 ± 0.20**	0.62 ± 0.07	**0.40 ± 0.18**
	MAA hull	9.99 ± 0.00	1.68 ± 0.69	0.84 ± 0.44	4.10 ± 0.61
Normalisation	MAA result	**4.51 ± 0.00**	**0.73 ± 0.07**	**0.71 ± 0.10**	**0.06 ± 0.15**
	MAA hull	9.99 ± 0.00	1.45 ± 0.24	1.18 ± 0.29	4.28 ± 0.38
Initialisation	MAA result	**0.47 ± 0.14**	0.73 ± 0.06	0.71 ± 0.10	**−0.21 ± 0.11**
	MAA hull	1.76 ± 0.14	0.54 ± 0.26	**0.24 ± 0.09**	3.20 ± 0.11
Mutation	MAA result	**0.47 ± 0.13**	0.69 ± 0.08	0.68 ± 0.09	**−0.22 ± 0.09**
	MAA hull	1.69 ± 0.69	0.55 ± 0.44	0.31 ± 0.52	3.07 ± 0.25
KL-divergence measured in normalised space					
Base	MAA result	**1.47 ± 0.57**	**1.42 ± 0.00**	**1.36 ± 0.03**	**4.49 ± 0.19**
	MAA hull	6.14 ± 1.54	3.92 ± 0.00	2.89 ± 0.30	8.55 ± 0.33
Crossover	MAA result	**5.19 ± 0.00**	1.69 ± 0.61	1.49 ± 0.02	**1.07 ± 0.04**
	MAA hull	10.16 ± 0.00	2.24 ± 1.06	1.20 ± 0.56	4.68 ± 0.69
Normalisation	MAA result	**5.19 ± 0.00**	1.39 ± 0.11	1.44 ± 0.05	**0.74 ± 0.09**
	MAA hull	10.16 ± 0.00	1.53 ± 0.24	1.24 ± 0.28	4.62 ± 0.35
Initialisation	MAA result	**1.16 ± 0.06**	1.43 ± 0.06	1.45 ± 0.08	**0.47 ± 0.08**
	MAA hull	2.24 ± 0.19	0.67 ± 0.32	0.28 ± 0.13	3.50 ± 0.13
Mutation	MAA result	1.21 ± 0.14	1.39 ± 0.10	1.43 ± 0.05	**0.58 ± 0.17**
	MAA hull	1.99 ± 0.75	**0.66 ± 0.47**	**0.46 ± 0.58**	3.39 ± 0.31

Table 5.7: Overview of the KL-divergence of the population to the output and hull of the MAA method at the first generations where each convergence criteria is satisfied. The results are averaged over 5 runs with distinct seeds, in the format of mean ± standard deviation.

After adding the initialisation component, we see the KL-divergence to the MAA result increases instead of decreases for stricter convergence criteria, while the KL-divergence to the hull still increases. This indicates a possible bias towards the border of the near-optimal space S_ε. A possible reason why this did not show up before the initialisation component is added is that the initialisation and mutation components improve the GA's exploration ability, such that no boundary remains unexplored.

Measuring the KL-divergences considering all the alternatives found throughout optimisation generally shows the KL-divergence to both the MAA result and hull to reduce when adding components. This can be interpreted as the alternatives to better cover both the inner space and the boundary of S_ε.

5.2.4. Average number of feasibility checks on non-near-optimal individuals at convergence

Setup	Mean percentage of individuals checked for near-optimality per generation at convergence generation						
	Investment space		*Normalised space*				
	MAA diversity	MGA diversity	MAA norm. diversity	MGA norm. diversity	Lenient convergence	Moderate convergence	Strict convergence
Base	100%	100%	100%	100%	100%	100%	100%
Crossover	45.3% ± 0.5%	45.4% ± 0.6%	45.4% ± 0.5%	45.4% ± 0.4%	72.8% ± 6.1%	45.0% ± 0.4%	45.1% ± 0.2%
Normalisation	45.8% ± 0.7%	45.7% ± 0.8%	45.8% ± 0.7%	45.6% ± 0.7%	81.0% ± 3.5%	45.1% ± 0.3%	**45.0% ± 0.3%**
Initialisation	**31.6% ± 1.7%**	**36.4% ± 2.2%**	**3.3% ± 9.2%**	**35.4% ± 1.5%**	**42.4% ± 1.1%**	**44.2% ± 0.2%**	**44.5% ± 0.2%**
Mutation	**31.2% ± 4.1%**	**38.5% ± 2.2%**	**4.7% ± 8.5%**	**37.6% ± 2.6%**	48.4% ± 0.8%	50.0% ± 0.5%	50.1% ± 0.2%

Table 5.8: Percentage of the offspring needing to be checked for near-optimality on average at each generation for different points of measurement and setups. The results are averaged over 5 runs with distinct seeds, in the format of mean ± standard deviation.

In Table 5.8, the percentage of the offspring that needs to be checked for near-optimality, averaged over each generation, is listed. The percentage is given for each of the setups explained in Table 4.6 at the different points of measurement, as listed in Table 5.2. Note that the base case always needs to check for near-optimality, because it exploits no properties of the ESOM to know when offspring must be near-optimal.

Generally, the more components are added, the fewer individuals need to be checked for near-optimality, as can be seen in Table 5.8 at the moments where the diversity of the population passes the diversity of other MGA methods, even below the expected 45% as explained in Section 4.5.8. The main reason is that diversity is achieved so early in the optimisation that the advantage of not checking the initial population for optimality significantly reduces the average number of checks required per generation.

Considering the three convergence criteria, the expected values of 45% and 50.5% are reached for most setups. For setups without the initialisation component, we observe a significantly higher percentage, as convergence occurs early in the optimisation process, and the full initial population must be checked for near-optimality.

Table 5.9 shows the percentage of offspring individuals which is non-near-optimal, averaged over each generation.

Setup	Mean percentage of non-near-optimal individuals per generation at convergence generation						
	Investment space		*Normalised space*				
	MAA diversity	MGA diversity	MAA norm. diversity	MGA norm. diversity	Lenient convergence	Moderate convergence	Strict convergence
Base	12.9% ± 0.4%	14.3% ± 0.3%	12.7% ± 0.4%	14.3% ± 0.2%	**15.4% ± 1.9%**	**22.9% ± 0.0%**	**27.1% ± 0.4%**
Crossover	**9.0% ± 1.6%**	**9.7% ± 1.7%**	**9.0% ± 1.5%**	**14.3% ± 3.8%**	47.6% ± 4.5%	**21.8% ± 4.0%**	30.3% ± 2.3%
Normalisation	**10.0% ± 0.9%**	**11.1% ± 0.7%**	**10.0% ± 0.8%**	**12.8% ± 0.4%**	48.8% ± 10.4%	29.7% ± 2.1%	34.0% ± 0.5%
Initialisation	**12.4% ± 2.3%**	15.3% ± 2.5%	**8.2% ± 2.0%**	**14.4% ± 2.0%**	28.7% ± 1.6%	36.7% ± 1.4%	38.6% ± 0.7%
Mutation	15.1% ± 2.7%	18.7% ± 1.8%	**10.7% ± 3.1%**	18.0% ± 2.3%	36.2% ± 1.0%	43.4% ± 1.0%	44.6% ± 0.7%

Table 5.9: Percentage of the offspring individuals that are non-near-optimal on average at each generation for different points of measurement and setups. The results are averaged over 5 runs with distinct seeds, in the format of mean ± standard deviation.

The results show that adding the mutation component consistently increases the percentage of non-near-optimal individuals, indicating an improvement in the GA's exploration ability. Again, because of the immediate convergence of the lenient convergence criteria results in a higher percentage of non-near-optimal individuals for the setups without the initialisation component.

As the convergence criteria become stricter, the percentage of non-near-optimal individuals increases. This is as expected, since the algorithm continues to explore even after it has found the boundary of S_ε.

For reference, the total number of individuals checked for and infeasible individuals can be found in Appendix B.

5.3. Performance evaluation

The following sections discuss the results from the experiments described in Section 4.6 to test the performance of the GA compared to the baselines discussed in Section 4.6.1.

5.3.1. Experiment 1: Diversity throughout optimisation

This section discusses how the diversity changes throughout the optimisation process for the different approaches. Figures 5.3 and 5.4 show the diversity of the population and all the near-optimal alternatives found so far for the different approaches, excluding the weighted sum MGA approach, as it does not generate points inside the near-optimal space S_ε. Each column of plots corresponds to a different approach, and each row corresponds to a node being added to the ESOM showcase discussed in Section 4.5.8.1, increasing the dimensionality by three for each added node. The colour of the lines indicates how many alternatives are generated by the MAA approach or the population size for the GA-based approaches. For the MAA approach in the first column, the The dashed lines measure the diversity of the current population at the number of individuals generated in total, meaning the diversity is measured after selection is performed. The solid line shows the diversity of all the near-optimal alternatives found so far after adding each alternative. Figure 5.3 shows the dimensions 3 to 9 where the MAA method can still be applied, while Figure 5.4 shows the dimensions 12 to 21 where MAA cannot be applied.

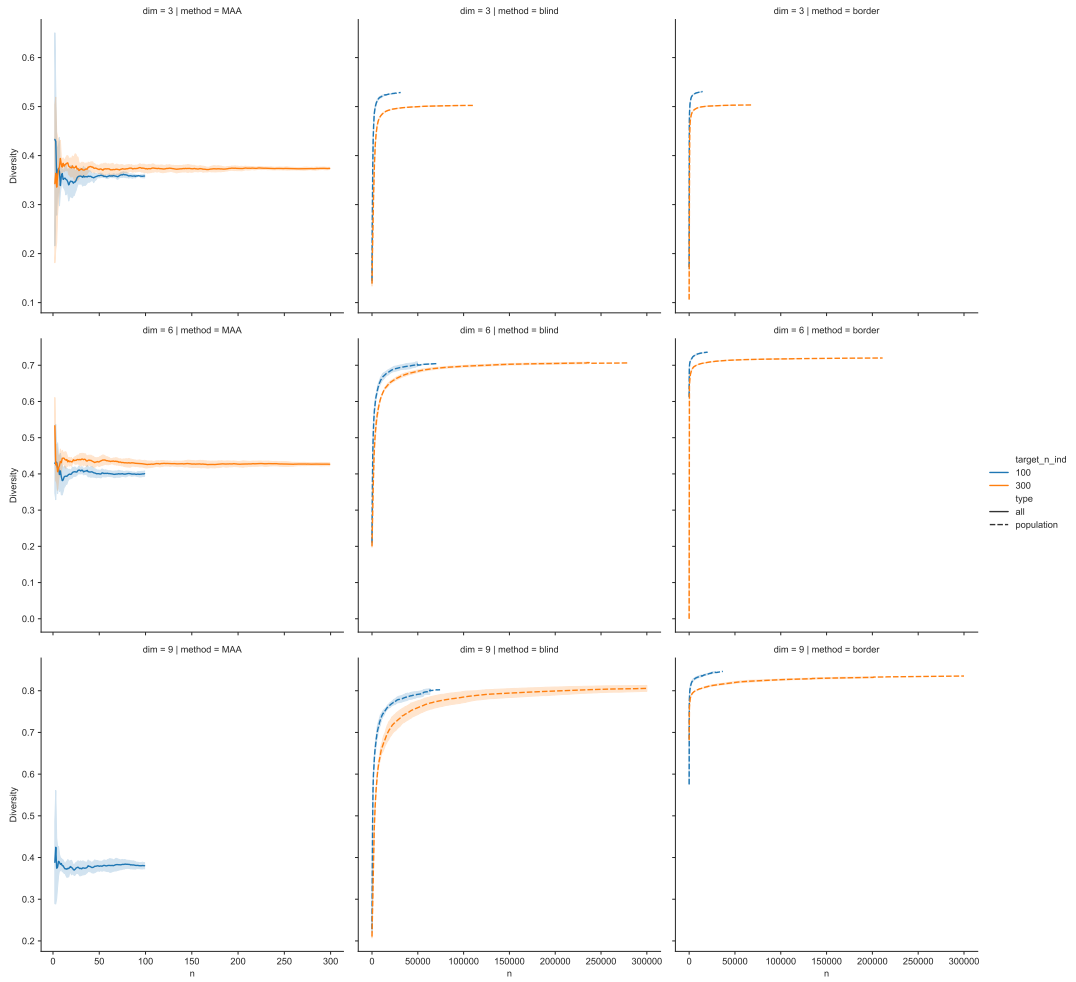


Figure 5.3: The diversity of the set of alternatives found so far throughout execution of the approaches. Each column represents one of the methods on a problem, with the rows corresponding to the number of investment variables from 3 to 9 of the ESOM showcase. The different sizes of the alternative set or population are indicated by different colours in the graph. Data is averaged over 5 runs with distinct seeds.

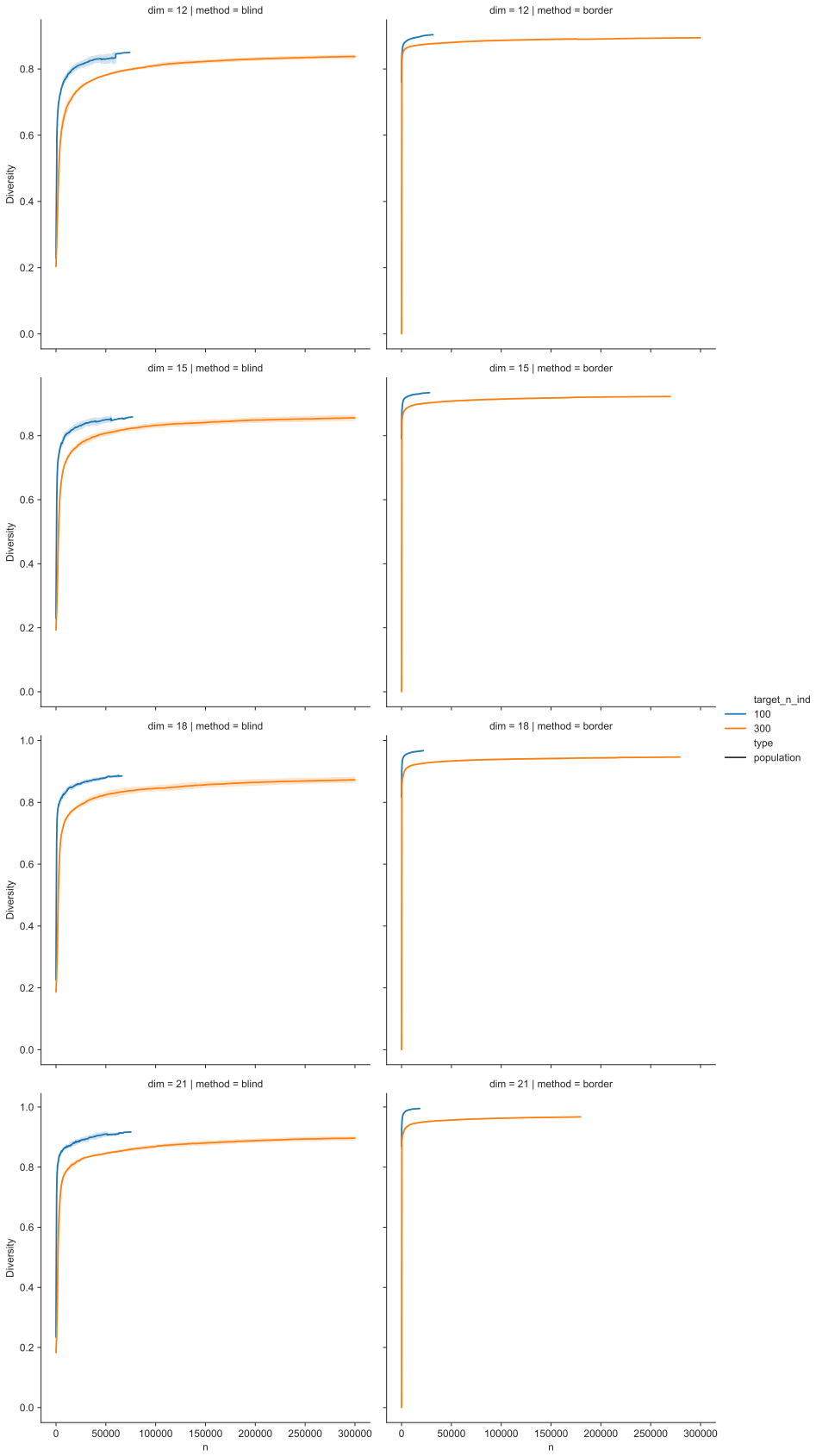


Figure 5.4: The diversity of the set of alternatives found so far throughout execution of the approaches. Each column represents one of the methods on a problem, with the rows corresponding to the number of investment variables from 3 to 9 of the ESOM showcase. The different sizes of the alternative set or population are indicated by different colours in the graph. Data is averaged over 5 runs with distinct seeds.

In the figures, we see that the GA based approach consistently achieves a higher diversity than the MAA approach, as expected. Our expectations of the 'border GA' to converge faster than the 'blind GA' approach also hold.

For higher dimensions, the 'border GA' converges to a higher diversity than its respective 'blind GA' line, either indicating the 'blind GA' would need more time to converge to a similar diversity, or that it's not sufficiently capable to fully explore the near-optimal space. This observation will be further discussed in Section 6.3.1.

5.3.2. Experiment 2: Diversity for increasing dimensionality

The following section will discuss the results of the experiment to track the diversity while increasing the number of investment variables of the ESOM showcase. Figure 5.5 shows how the diversity changes for the different approaches. Every column corresponds to a different convergence criterion. Each approach has a different line colour, while the difference in line style shows the different number of alternatives generated or the population sizes.

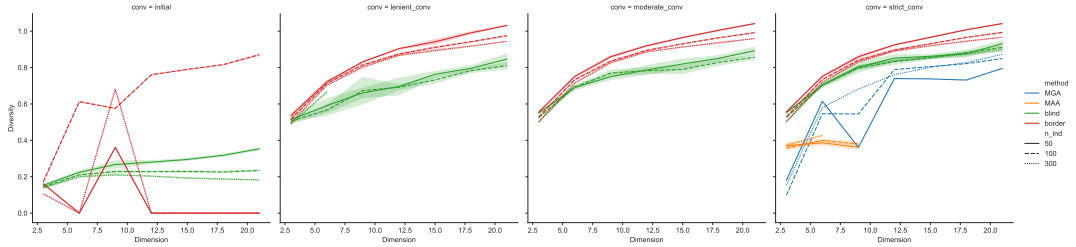


Figure 5.5: The diversity of the populations at the start of the optimisation and different moments of convergence for different dimensionalities of the ESOM showcase. The approaches and population sizes or number of alternatives are varied with line styling. For each plot that is more to the right, stricter convergence criteria are used. Data is averaged over 5 runs with distinct seeds.

From the figure, we conclude that our expectations regarding the diversity to increase when the number of investment variables increases, and that larger populations have lower diversity, are correct.

Similar to the results discussed in the previous section, the diversity of the 'border GA' approach is higher than that of the 'blind GA' approach. This observation will be further discussed in Section 6.3.1.

5.3.3. Experiment 3: Scalability

The next section discusses the result of the scalability experiment. The line colours correspond to the different approaches, and the line style indicates which convergence criterion is met. Figure 5.6 shows the computational cost per alternative in seconds for the different approaches for an increasing number of investment variables of the ESOM showcase. The columns correspond to whether to total number of alternatives found so far or the population size is considered to calculate the cost per alternative. In every row, the number of alternatives generated or the population size is altered.

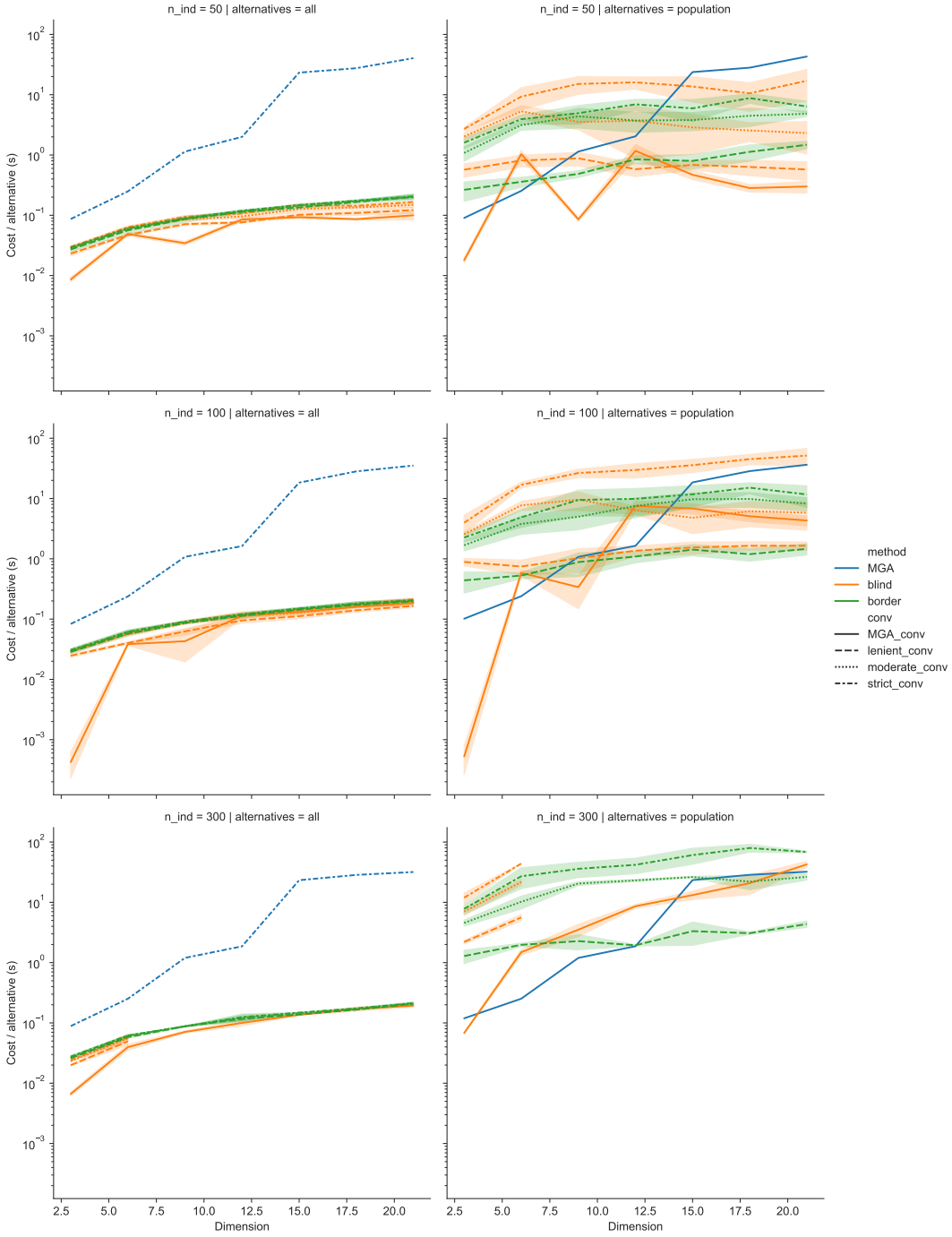


Figure 5.6: The cost per alternative for diversity of the populations at the start of the optimisation and different moments of convergence for different approaches and population sizes or number of alternatives. For each plot that is more to the right, stricter convergence criteria are used. Data is averaged over 5 runs with distinct seeds.

The figure confirms our expectations of the cost per alternative of the GA method to be in the same order of magnitude as the weighted MGA method. Also, the computational cost per alternative when considering all alternatives found throughout the optimisation process is much lower compared to the cost of the weighted sum MGA method, as expected. Note that the line for a population size of 300 for the 'blind GA' approach is cut short, as the convergence criteria were not met for a dimensionality higher than 6.

When the population size is relatively low compared to the dimensionality of the ESOM showcase — as described in Section 4.4 — we expect the computational cost per alternative to reduce when the dimensionality increases for each of the convergence criteria. For the smallest population — having a size of 50 — the issue

is the most visible, as the cost per alternative of the population slightly decreases for the 'blind GA' approach when the dimensionality increases. Since the initial population of the 'border GA' contains only individuals, this bias to the boundary is expected to result in a lower cost per alternative. However, we only clearly see this lower cost for the strictest convergence criterion, possibly indicating the bias towards the border of the near-optimal space. We further reflect on whether this bias is occurring in Section 6.4.

5.4. Alternatives of the SECURES model

Figure 5.7 shows the alternatives found by the GA method applied to the SECURES model. For each technology, the capacities invested in the different countries are summed to a total for each alternative. Therefore, every alternative is represented by a marker in every category. The green line shows the summed capacities for the minimum cost investment plan.

Note that the capacities shown are the increase in the capacities. In other words, the ESOM already contains installed capacities, and the investments decide how much these capacities are increased. Also, the logarithmic scale may give the impression of a hard 'cutoff value' with many alternatives clustered for some technologies, while in reality, these alternatives may still be well distributed.

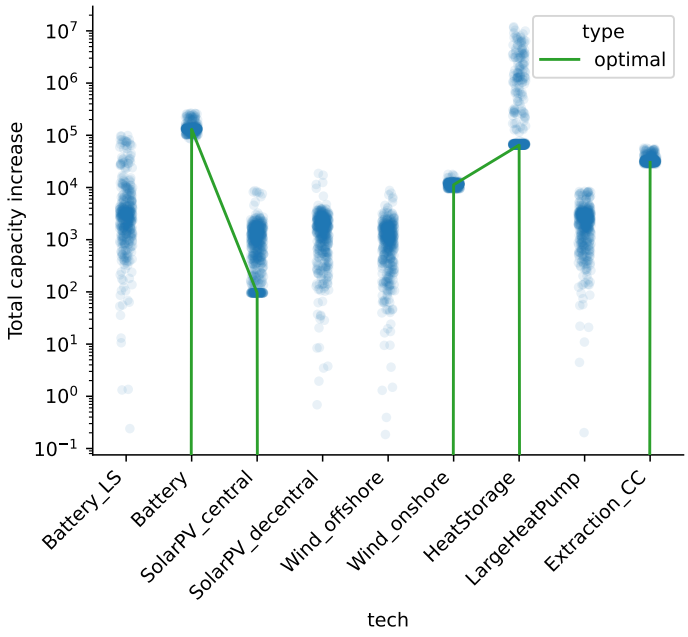


Figure 5.7: Distribution of total capacity per technology for the alternatives found in the SECURES model.

The plot shows that many alternatives lead to higher invested capacities than the minimum-cost solution. This may indicate that the GA is not sufficiently covering regions of the near-optimal space S_ε where capacities are reduced compared to the minimum cost investment plan, or because these regions are relatively small, causing aggregated capacities per technology to appear consistently higher.

6

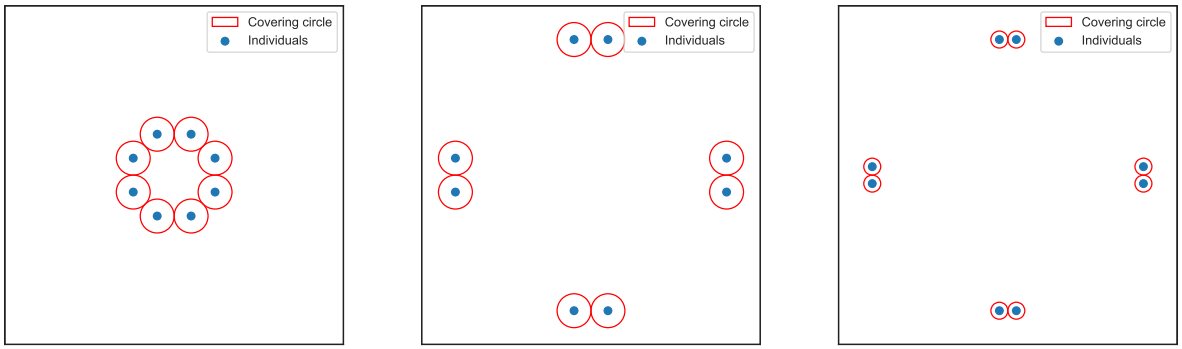
Discussion

As the results have been reported, this chapter will discuss some of the insights from these results. First, we reflect on the choice of the diversity metric. Then, we discuss some of the unexpected insights from comparing the components of the GA.

6.1. Diversity metrics

In the results of testing populations in Section 5.1.1, we have seen that a lot of diversity metrics are inconsistent with the covering radius distribution property. As the property comes from our intuition regarding even space coverage, we lack proof that the property holds for any two populations following the condition on the distribution of mean and standard deviation of the minimum distances.

To illustrate this point, we give the following example. Considering the populations in Figure 6.1, the covering radius distribution property states that $D(P_1) > D(P_3)$ while giving no preference between P_1 and P_2 caused by the insensitivity of using minimum distances. However, one could argue that intuitively $D(P_1) < D(P_3)$ as the tightest convex hull containing the individuals of P' has a larger volume than the one containing individuals of P. This situation highlights the trade-off between filling a larger volume of space and achieving even distribution within the covered space. The monotonicity property states that $D(P_2) > D(P_1)$ and $D(P_2) > D(P_3)$, showing more sensitivity when a population covers a larger space. Therefore, the covering radius distribution property is treated as an indicator mainly used when the convex hulls of the compared populations are close to the same volume. However, as almost none of the considered diversity metrics agree with the property when comparing the Sobol sequence with modified concentration populations — as can be seen in Table A.2 — it shows this property is not to be followed too strictly.



(a) Population P_1 (b) Population P_2 (c) Population P_3

Figure 6.1: Example of when we are uncertain about the conclusion of the covering radius distribution property.

As different diversity metrics result in different preferences in the comparison of P_1 and P_3, the choice of diversity metric then also implicitly defines the preferences in the trade-off between filling a larger volume of space and achieving even distribution within the covered space.

6.1.1. Choice of metric

As discussed in Section 5.1.3, the harmonic mean distance diversity metric was chosen to be used in the other experiments and the final GA.

6.2. GA components

The results of comparing the components suggest that all components should be used in the GA to achieve the best results. However, as we have seen that adding the mutation component greatly improves the exploration speed of the GA, it could be beneficial to disable this component, to allow finding more feasible alternatives close to the minimum cost investment plan. In essence, this would favour exploitation in the first part of the optimisation process and could reduce the bias towards the border of the near-optimal space for ESOMs with a relatively high number of investment variables.

6.2.1. Diversity at convergence

The little difference in diversity convergence criteria after the normalisation component is added, as explained in Section 5.2.2, raises the question whether the most lenient criterion is most useful in practice as an early stopping criterion.

In practice, convergence criteria are useful as indicators of how well the execution of the GA converged. However, it is in general better to give the GA as much time as possible, leading often to the usage of a maximum computational budget as a stopping criterion.

6.3. Performance evaluation experiments

6.3.1. Lower diversity at convergence without knowledge of boundary at initialisation

In the results of the performance evaluation, Sections 5.3.1 and 5.3.2 highlighted the diversity of the 'blind GA' approach to be lower than the 'border GA' approach. A possible explanation would be that increasing the dimension would make it harder for the 'blind GA' approach to explore all regions. As the initialisation operator produces individuals where the investment decisions only increase compared to the minimum cost investment plan, the other regions of the near-optimal space S_ε, where some capacities decrease, could get relatively larger when the number of investment variables increases. Therefore, further experiments are needed to determine whether this is truly the case or the GA only needs more time to converge to a similar diversity.

6.3.2. Computational advantage of GA approach with relatively high temporal resolution

Section 5.3.3 shows how the computational cost of the GA approach does not exceed the cost of a weighted sum MGA method. This is a result of the lower solving time for solving the ESOM for unit commitment, compared to also solving it for the investment variables, which compensates for the higher number of solves required by the GA.

However, when the temporal resolution increases, the number of operational variables increases, which could reduce the benefit of only solving for unit commitment. Further experiments would be necessary to clarify how increasing the temporal resolution impacts this trade-off. Ideally, some relation between the number of investment variables and the number of operational variables could be found for which the benefit of only solving for unit commitment is sufficiently large, such that the computational burden of the GA method does not exceed the cost of a weighted sum MGA method.

6.4. Bias towards the border of the near-optimal space for a high number of investment variables

In the results of the experiments, we noticed several indications of a bias towards the boundary of the near-optimal space S_ε. This Section will analyse those situations and relate them to the expected relation between population size and number of investment variables of the ESOM discussed in Section 4.4.

6.4.1. KL-divergence in component comparison

The results of comparing the KL-divergence in the component comparison experiment indicated a possible bias towards the border of S_ε (see Section 5.2.3). As expected relation between the number of investment

variables v and population size $|P|$ states that we expect no bias towards the boundary when $2^v \leq |P|$, we would not expect the bias to occur when $v = 3$ and $|P| = 100$.

This observation raises questions about whether comparing KL-divergences is a useful indicator of when the bias towards the boundary occurs.

6.4.2. Scalability performance evaluation

The results of the scalability experiment of the performance evaluation of the GA in Section 5.3.3 indicated a possible bias towards the border of S_ε. When considering the smallest population $|P| = 50$ and the largest number of investment variables $v = 21$, we notice that $|P|$ is close to $2v$, where we would be confident that the GA is biased towards the boundary when $|P| \leq 2v|$. However, since $2v < |P| \leq 2^v$ still holds for all the population sizes considered in the experiment, we are uncertain whether the GA is biased towards the boundary of the space.

To conclude, further experiments would be needed to better grasp the impact of a bias towards the boundary on the quality of the output, and in what situations the GA is biased. We also suggest identifying better indicators of when the bias occurs, to be more certain of its presence.

6.5. Hyperparameter tuning

6.5.1. Population size

6.5.2. Component parameters

6.6. Further testing on the European power system

Within this work, no actual analysis of the alternatives generated for the European power system was done in terms of secondary objectives. Such an analysis would be a useful example to highlight how alternatives that uniformly cover the space of alternatives can improve the decision-making process when doing a similar analysis using alternatives generated by other MGA methods.

Also, applying the GA to different sizes of the European power system by including a different number of countries can verify the conclusions on the ESOM showcase for a more realistic ESOM.

Finally, analysing the distribution of the costs of the produced alternatives would show how well the GA method is able to find points inside the near-optimal space.

6.7. Future work

Besides investigating whether the 'blind GA' approach can reach the same diversity as the 'border GA' approach (Section 6.3.1), the GA has worse computational performance compared to other MGA methods for an increased temporal resolution of the ESOM (Section 6.3.2), when the GA is biased towards the border of the near-optimal space due to the curse of dimensionality (Section 6.4) and doing futher analysis on a realistic model like the European power system (Section 6.6), we propose the following potential improvements of the GA.

6.7.1. Handling discrete variables

Since the computational burden of the GA lies in solving the individuals for unit commitment, the investment variables could be defined as discrete variables without greatly reducing the computational performance. However, the current operators need to be modified to support doing so, where the crossover operator loses the ability to produce offspring, which is certain to be feasible without needing to solve for unit commitment.

Potentially, it can be assumed that the convex combinations of feasible parents that are rounded to the nearest discrete solution are feasible. This could potentially 'lose' a few alternatives when post-processing them for unit commitment before analysis of secondary objectives.

6.7.2. Hyperparameter tuning

This work only used different population sizes in some of the experiments. The performance in terms of convergence speed and computational cost could be impacted by tuning parameters stated in Table 4.4 to increase or decrease the number of unit commitment solves needed, exploration speed of the GA.

6.7.3. Computational improvements

The main computational bottleneck of the GA lies in solving individuals for unit commitment. Reducing the cost of such solves would improve the GA's performance, allowing the GA to run more generations for the same computational cost.

Surrogate modelling offers a promising way to reduce the computational cost of individual solves. Alternatively, solving only those individuals that cover unexplored regions of the near-optimal space could lower the number of unit commitment solves required.

7

Conclusion

In this work, we have suggested an alternative MGA that uses diversity optimisation to find alternatives that evenly cover the near-optimal space.

Firstly, we defined how the quality of the MGA method's output can be measured in terms of how well it covers the near-optimal space S_ε. Several diversity metrics have been compared and evaluated to pick the most suitable one to be used in an optimisation setting. This resulted in the usage of the 'harmonic mean of pairwise distances' diversity metric (see A.1) as it is not inherently biased towards the border of S_ε, satisfies the properties we want the metric to have and is most in line with our ordering of example populations from least to most diverse.

Then, a GA was designed to optimise the diversity metric. The algorithm's components are specifically tailored to generate alternatives for ESOMs efficiently. Each component's improvement of the algorithm's performance has been iteratively tested on an ESOM showcase that can easily be modified to change the number of investment decisions. The tests generally showed that adding the components improved the rate of convergence of the GA, without limiting the diversity of the output.

After designing the GA, it was compared to state-of-the-art methods from the literature in terms of result quality and scalability on an ESOM showcase. This has shown us that the proposed GA method can generate more diverse alternatives without increasing the computational burden per generated alternative. It also showed that the GA could benefit from obtaining more knowledge about the boundary of S_ε and more time to converge.

Finally, the GA method was applied to a scaled-down version of a model of the European power system, giving an example output of all the alternatives.

Analysing the output indicates a potential benefit of generating alternatives evenly throughout the near-optimal space. Using these alternatives in the decision-making process can accelerate the extraction of stakeholder requirements and the identification of more agreeable compromises. However, further tests are required to verify the potential benefit by comparing the resulting alternatives with alternatives generated by other state-of-the-art MGA methods.

Applying the GA to the European power system has shown promising results considering a larger number of investment variables, encouraging further testing on larger-scale ESOMs.

7.1. Acknowledgements

First and foremost, I wish to express my gratitude to my daily supervisor, Périne Cunat, for her guidance in helping me prioritise my work, her critical questions, and our many fruitful discussions. I also thank my academic supervisors, Neil Yorke-Smith and Jochen Cremer. The expertise and feedback of all the supervisors greatly helped shape the direction and quality of this work.

I am thankful to the Austrian Institute of Technology for allowing me to conduct this research at their facilities in Vienna, and to my colleagues at AIT for their interest in my research, stimulating questions and enjoyable informal moments.

Finally, I wish to thank Anton Bouter, from CWI, for an insightful discussion of the problem for which we employ a GA to solve and for posing critical questions regarding the GA's design.

Appendices

A

Diversity measures

A.1. Overview of considered diversity measures

Table A.1 gives an overview of all diversity metrics considered in this work. Note that more metrics were found in literature, but were not considered due to computational infeasibility or being ill-defined for measuring diversity of a population in some convex space for which the boundaries are considered unknown.

Table A.1: Overview of diversity metrics considered in the experiments. The metrics in bold are discussed in the report.

Name	$D(P) =$	Type	Reference
Mean distance	$\frac{2}{\lvert P\rvert(\lvert P\rvert-1)}\sum_{\mathbf{k},\mathbf{l}\in P} d(\mathbf{k},\mathbf{l})$	Sum of distances	[7]
Harmonic energy	$\sum_{\mathbf{k},\mathbf{l}\in P} d(\mathbf{k},\mathbf{l})^2$	Sum of distances	[3]
Minimum spacing	$\min_{\mathbf{k},\mathbf{l}\in P:\ \mathbf{k}\neq\mathbf{l}} d(\mathbf{k},\mathbf{l})$	Sum of distances	[3]
Population radius	$\max_{\mathbf{i}\in P} d(\mathbf{i},\bar{\mathbf{i}})$	Sum of distances	[7]
Distance to average point	$\frac{1}{N}\sum_{\mathbf{i}\in P} d(\mathbf{i},\bar{\mathbf{i}})$	Sum of distances	[7]
Harmonic mean distance	$\frac{\lvert P\rvert(\lvert P\rvert-1)/2}{\sum_{\mathbf{k},\mathbf{l}\in P:\ \mathbf{k}\neq\mathbf{l}} \frac{1}{d(\mathbf{k},\mathbf{l})}}$	Sum of fractional distances	[2]
Harmonic mean minimal distances	$\frac{\lvert P\rvert}{\sum_{\mathbf{k}\in P}\max_{\mathbf{l}\in P:\ \mathbf{l}\neq\mathbf{k}} \frac{1}{d(\mathbf{k},\mathbf{l})}}$	Sum of fractional distances	
Coulomb potential	$\sum_{\mathbf{k},\mathbf{l}\in P:\ \mathbf{k}\neq\mathbf{l}} \frac{1}{d(\mathbf{k},\mathbf{l})}$	Sum of fractional distances	[3]
Riesz s-energy	$\sum_{\mathbf{k},\mathbf{l}\in P:\ \mathbf{k}\neq\mathbf{l}} \frac{1}{d(\mathbf{k},\mathbf{l})^s}$	Sum of fractional distances	[3]
Logarithmic energy	$\sum_{\mathbf{k},\mathbf{l}\in P:\ \mathbf{k}\neq\mathbf{l}} \log\frac{1}{d(\mathbf{k},\mathbf{l})}$	Sum of fractional distances	[3]
Wrap-around L_2-discrepancy	$-\left(\frac{4}{3}\right)^{\nu} - \frac{1}{\lvert P\rvert^2}\sum_{\mathbf{k},\mathbf{l}\in P}\prod_{j=1}^{\nu}\left(\frac{3}{2}\lvert k_j-l_j\rvert+\lvert k_j-l_j\rvert^2\right)$	Discrepancy	[48]

nVol	$\sqrt{V_{pop}/V_{lim}}$ $V_{pop} = \sqrt{\prod_{j=1}^{v} 2 \cdot iqr(\{\mathbf{i}_j \mid \mathbf{i} \in P\})}$ $V_{lim} = \sqrt{\prod_{j=1}^{v} \lvert u_j - l_j \rvert}$	Other	[26]
Moment of inertia	$\sum_{\mathbf{i} \in P} \sum_{j=1}^{v} (\mathbf{i}_j - \bar{\mathbf{i}}_j)^2$	Other	[7]
True diversity	$\frac{1}{\lvert P \rvert} \sqrt{\sum_{j=1}^{v} \left(\bar{\mathbf{i}_j^2} - (\bar{\mathbf{i}}_j)^2 \right)}$ $\bar{\mathbf{i}_k^2} = \frac{1}{\lvert P \rvert} \sum_{\mathbf{i} \in P} \mathbf{i}_j^2$	Other	[7]
Mean normalised standard deviation	$\frac{1}{v} \sum_{j=1}^{v} \frac{\sigma_j(P)}{\lvert \bar{\mathbf{i}}_j \rvert}$ $\sigma_j(P) = \frac{1}{\lvert P \rvert} \sum_{\mathbf{i} \in P} \left(\mathbf{i}_j - \bar{\mathbf{i}}_j \right)^2$	Other	[8]

Other methods use some measure of empty space [17]

“The best known algorithm for the star discrepancy computation has a running time of order n1+d/2 [7], which is exponential in the dimension d.” ([Neumann et al., 2018, p. 5])

A.2. Populations used to compare diversity measures

A.2.1. Populations in unit square space

Table A.2: How our expectations of how the diversity of each pair of populations in unit square space are ordered relative to each other are correct. '>' indicates $D(P_{row}) > D(P_{column})$, and '<' indicates the reverse. Entries marked with '(>)' or '(<)' follow from the covering radius distribution property. The green coloured cells are correct, the red ones are incorrect, and the yellow ones are the relations defined using the output diversities. The bold diversity value indicates the highest value.

Type	**Population**	$D(P)$	P_s	P_g	$P_{mv.u}$	$P_s.b$	$P_s.c$	$P_{s.sep}$	$P_{s.cen}$	$P_{mv.n}$
Sum of distances – Mean distance										
Uniform	P_s	0.523	=	(<)	(>)	>	>	>	>	>
	P_g	0.557	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.524	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.528	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	0.514	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.605**	<	(<)	>	>	>	=	>	(>)
	$P_{s.cen}$	0.262	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.369	<	<	<	(<)	(<)	(<)	>	=
Sum of distances – Harmonic energy										
Uniform	P_s	10923	=	(<)	(>)	>	>	>	>	>
	P_g	12379	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	11001	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	11135	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	10568	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**15019**	<	(<)	>	>	>	=	>	(>)
	$P_{s.cen}$	2731	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	5585	<	<	<	(<)	(<)	(<)	>	=
Sum of distances – Population radius										
Uniform	P_s	0.498	=	(<)	(>)	>	>	>	>	>
	P_g	0.500	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	**0.532**	(<)	(<)	=	(<)	(<)	>	>	>
Concentration	$P_{s.b}$	0.499	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	0.499	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	0.499	<	(<)	<	>	>	=	>	(>)
	$P_{s.cen}$	0.249	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.503	<	<	<	(<)	(<)	(<)	>	=
Sum of distances – Distance to average point										
Uniform	P_s	0.383	=	(<)	(>)	>	>	>	>	>
	P_g	0.408	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.382	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.386	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	0.377	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.469**	<	(<)	>	>	>	=	>	(>)
	$P_{s.cen}$	0.191	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.261	<	<	<	(<)	(<)	(<)	>	=
Sum of fractional distances – Harmonic mean distance										
Uniform	P_s	0.360	=	(<)	(>)	>	>	>	>	>
	P_g	**0.388**	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.339	(<)	(<)	=	(<)	(<)	>	>	>
Concentration	$P_{s.b}$	0.358	<	(<)	(>)	=	(<)	>	>	(>)
	$P_{s.c}$	0.331	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	0.335	<	(<)	<	<	>	=	>	(>)
	$P_{s.cen}$	0.180	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.234	<	<	<	(<)	(<)	(<)	>	=

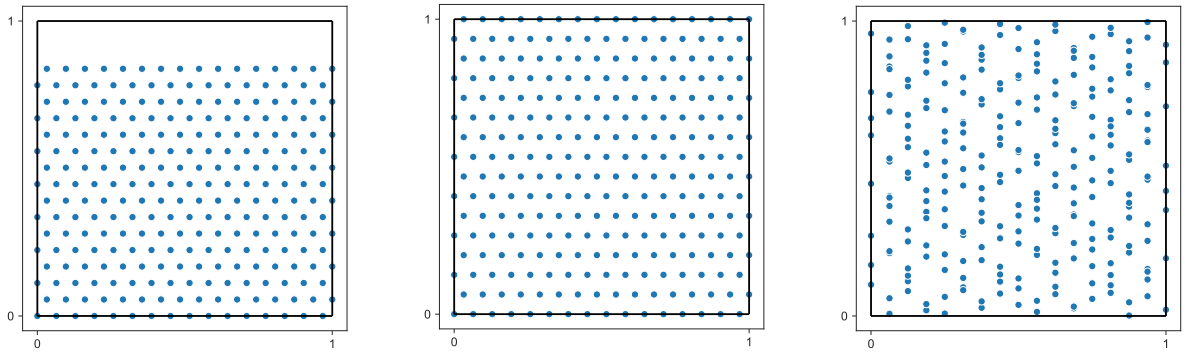
Table A.2: Whether our expectations of how each pair of test populations perform against each other in terms of diversity in unit square space are correct. '>' indicates $D(P_{row}) > D(P_{column})$, and '<' indicates the reverse. Entries marked with '(>)' or '(<)' follow from the covering radius distribution property. The green coloured cells are correct, the red ones are incorrect, and the yellow ones are the relations defined using the output diversities. The bold diversity value indicates the highest value.

Type	**Population**	$D(P)$	P_s	P_g	$P_{mv.u}$	$P_s.b$	$P_s.c$	$P_{s.sep}$	$P_{s.cen}$	$P_{mv.n}$
Sum of fractional distances – Harmonic mean minimal distances										
Uniform	P_s	0.040	=	(<)	(>)	>	>	>	>	>
	P_g	**0.067**	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.021	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.036	<	(<)	(>)	=	(<)	>	>	(>)
	$P_{s.c}$	0.031	<	(<)	(>)	(>)	=	>	>	(>)
Bias to border	$P_{s.sep}$	0.028	<	(<)	>	<	<	=	>	(>)
	$P_{s.cen}$	0.020	<	(<)	<	<	<	<	=	>
Suboptimal	$P_{mv.n}$	0.014	<	<	<	(<)	(<)	(<)	<	=
Sum of fractional distances – Coulomb potential[1]										
Uniform	P_s	90559	=	(<)	(>)	>	>	>	>	>
	P_g	**84166**	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	96418	(<)	(<)	=	(<)	(<)	>	>	>
Concentration	$P_{s.b}$	91081	<	(<)	(>)	=	(<)	>	>	(>)
	$P_{s.c}$	98658	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	97511	<	(<)	<	<	>	=	>	(>)
	$P_{s.cen}$	181118	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	139556	<	<	<	(<)	(<)	(<)	>	=
Sum of fractional distances – Riesz 2-energy[2]										
Uniform	P_s	490254	=	(<)	(>)	>	>	>	>	>
	P_g	**385999**	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	1117532	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	534624	<	(<)	(>)	=	(<)	>	>	(>)
	$P_{s.c}$	750360	<	(<)	(>)	(>)	=	>	>	(>)
Bias to border	$P_{s.sep}$	798071	<	(<)	>	<	<	=	>	(>)
	$P_{s.cen}$	1961014	<	(<)	<	<	<	<	=	>
Suboptimal	$P_{mv.n}$	3258653	<	<	<	(<)	(<)	(<)	<	=
Sum of fractional distances – Logarithmic energy[2]										
Uniform	P_s	25897	=	(<)	(>)	>	>	>	>	>
	P_g	23819	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	26159	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	25666	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	26923	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**23283**	<	(<)	>	>	>	=	>	(>)
	$P_{s.cen}$	48521	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	37892	<	<	<	(<)	(<)	(<)	>	=
Discrepancy – Wrap-around discrepancy[2]										
Uniform	P_s	**0.000**	=	(<)	(>)	>	>	>	>	>
	P_g	0.004	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.002	(<)	(<)	=	(<)	(<)	>	>	>
Concentration	$P_{s.b}$	0.001	<	(<)	(>)	=	(<)	>	>	(>)
	$P_{s.c}$	0.002	<	(<)	(>)	(>)	=	>	>	(>)
Bias to border	$P_{s.sep}$	0.056	<	(<)	<	<	<	=	>	(>)
	$P_{s.cen}$	0.113	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.045	<	<	<	(<)	(<)	(<)	>	=

[1] A lower value of this metric signifies a more diverse population

Table A.2: Whether our expectations of how each pair of test populations perform against each other in terms of diversity in unit square space are correct. '>' indicates $D(P_{row}) > D(P_{column})$, and '<' indicates the reverse. Entries marked with '(>)' or '(<)' follow from the covering radius distribution property. The green coloured cells are correct, the red ones are incorrect, and the yellow ones are the relations defined using the output diversities. The bold diversity value indicates the highest value.

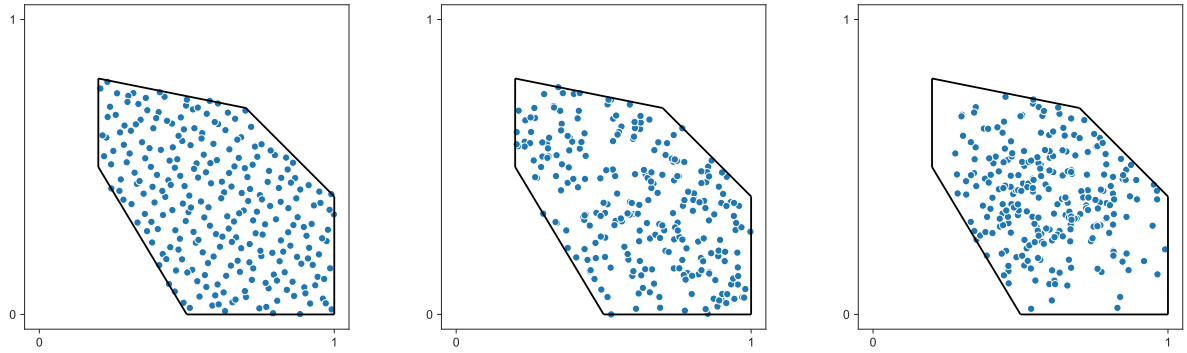
Type	**Population**	$D(P)$	P_s	P_g	$P_{mv.u}$	$P_s.b$	$P_s.c$	$P_{s.sep}$	$P_{s.cen}$	$P_{mv.n}$
Other – nVol										
Uniform	P_s	0.994	=	(<)	(>)	>	>	>	>	>
	P_g	1.000	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.991	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	1.017	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	0.938	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**1.493**	<	(<)	>	>	>	=	>	(>)
	$P_{s.cen}$	0.248	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.327	<	<	<	(<)	(<)	(<)	>	=
Other – Moment of inertia										
Uniform	P_s	42.7	=	(<)	(>)	>	>	>	>	>
	P_g	48.4	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	43.0	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	43.5	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	41.3	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**58.7**	<	(<)	>	>	>	=	>	(>)
	$P_{s.cen}$	10.7	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	21.8	<	<	<	(<)	(<)	(<)	>	=
Other – True diversity										
Uniform	P_s	0.083	=	(<)	(>)	>	>	>	>	>
	P_g	0.094	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.084	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.085	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	0.081	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.115**	<	(<)	>	>	>	=	>	(>)
	$P_{s.cen}$	0.021	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.043	<	<	<	(<)	(<)	(<)	>	=
Other – Mean normalised standard deviation										
Uniform	P_s	0.577	=	(<)	(>)	>	>	>	>	>
	P_g	0.615	(>)	=	(>)	(>)	(>)	(>)	(>)	>
	$P_{mv.u}$	0.545	(<)	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.583	<	(<)	(>)	=	(<)	<	>	(>)
	$P_{s.c}$	0.568	<	(<)	(>)	(>)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.671**	<	(<)	>	>	>	=	>	(>)
	$P_{s.cen}$	0.289	<	(<)	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.420	<	<	<	(<)	(<)	(<)	>	=



(a) Cropped hexagonal tiling $P_{hex.c}$ (b) Filled hexagonal tiling $P_{hex.f}$ (c) Discretised Sobol sequence $P_{s.d}$

Figure A.1: Two hexagonal populations created by fitting a hexagonal tiling in unit square space and one by discretising a single dimension of the Sobol sequence population P_s. All populations consist of 256 individuals.

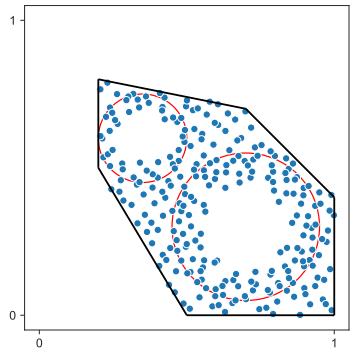
A.2.2. Populations in convex space



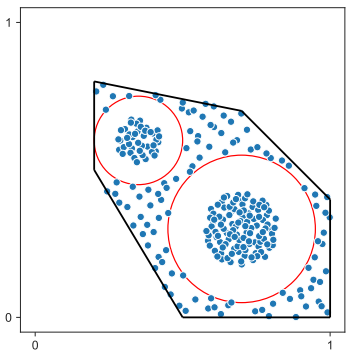
(a) Sobol sequence in convex space P_s^c

(b) Multivariate uniform sample in convex space $P_{mv.u}^c$

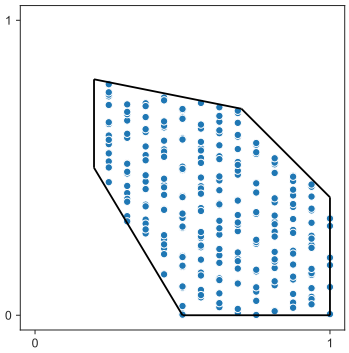
(c) Multivariate normal sample in convex space $P_{mv.n}^c$



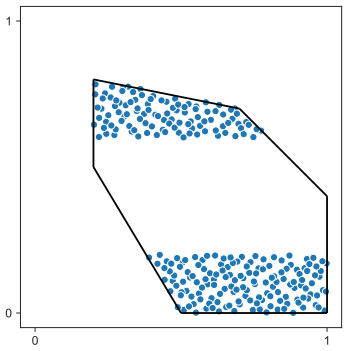
(d) Sobol sequence with blind spots in convex space $P_{s.b}^c$



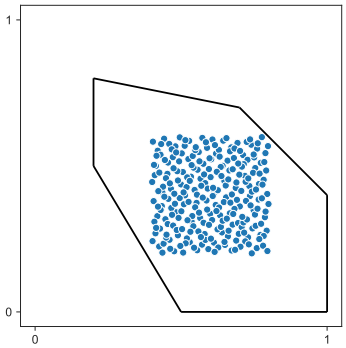
(e) Sobol sequence with concentrated spots in convex space $P_{s.c}^c$



(f) Discretised Sobol sequence in convex space $P_{s.d}^c$



(g) Sobol sequence with vertical separation in convex space $P_{s.sep}^c$



(h) Sobol sequence in centre square in convex space $P_{s.cen}^c$

Figure A.2: The variants of the test populations defined in convex space consisting of 256 individuals.

Table A.3: Whether our expectations of how each pair of test populations perform against each other in terms of diversity in convex space are correct. '>' indicates $D(P_{row}) > D(P_{column})$, and '<' indicates the reverse. Entries marked with '(>)' or '(<)' follow from the covering radius distribution property. The green coloured cells are correct, the red ones are incorrect, and the yellow ones are the relations defined using the output diversities. The bold diversity value indicates the highest value.

Type	**Population**	$D(P)$	P_s	$P_{mv.u}$	$P_{s.b}$	$P_{s.c}$	$P_{s.sep}$	$P_{s.cen}$	$P_{mv.n}$
Sum of distances – Mean distance									
	P_s	0.370	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	0.366	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.384	<	(>)	=	(>)	<	>	(>)
	$P_{s.c}$	0.337	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.432**	<	>	>	>	=	>	>
	$P_{s.cen}$	0.209	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.286	<	<	(<)	(<)	<	>	=
Sum of distances – Harmonic energy									
	P_s	5614	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	5549	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	6026	<	(>)	=	(>)	<	>	(>)
	$P_{s.c}$	4870	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**8274**	<	>	>	>	=	>	>
	$P_{s.cen}$	1748	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	3324	<	<	(<)	(<)	<	>	=
Sum of distances – Population radius									
	P_s	0.412	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	0.417	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.411	<	(>)	=	(>)	<	>	(>)
	$P_{s.c}$	0.412	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.441**	<	>	>	>	=	>	>
	$P_{s.cen}$	0.199	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.393	<	<	(<)	(<)	<	>	=
Sum of distances – Distance to average point									
	P_s	0.271	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	0.268	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.285	<	(>)	=	(>)	<	>	(>)
	$P_{s.c}$	0.245	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.343**	<	>	>	>	=	>	>
	$P_{s.cen}$	0.153	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.207	<	<	(<)	(<)	<	>	=
Sum of fractional distances – Harmonic mean distance									
	P_s	**0.248**	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	0.228	(<)	=	(<)	(<)	>	>	>
Concentration	$P_{s.b}$	0.245	<	(>)	=	(>)	>	>	(>)
	$P_{s.c}$	0.191	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	0.209	<	<	<	>	=	>	>
	$P_{s.cen}$	0.144	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.186	<	<	(<)	(<)	<	>	=
Sum of fractional distances – Harmonic mean minimal distances									
	P_s	**0.027**	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	0.014	(<)	=	(<)	(<)	<	<	>
Concentration	$P_{s.b}$	0.022	<	(>)	=	(>)	>	>	(>)
	$P_{s.c}$	0.017	<	(>)	(<)	=	>	>	(>)
Bias to border	$P_{s.sep}$	0.017	<	>	<	<	=	>	>
	$P_{s.cen}$	0.016	<	>	<	<	<	=	>
Suboptimal	$P_{mv.n}$	0.012	<	<	(<)	(<)	<	<	=

Table A.3: Whether our expectations of how each pair of test populations perform against each other in terms of diversity in convex space are correct. '>' indicates $D(P_{row}) > D(P_{column})$, and '<' indicates the reverse. Entries marked with '(>)' or '(<)' follow from the covering radius distribution property. The green coloured cells are correct, the red ones are incorrect, and the yellow ones are the relations defined using the output diversities. The bold diversity value indicates the highest value.

Type	**Population**	$D(P)$	P_s	$P_{mv.u}$	$P_{s.b}$	$P_{s.c}$	$P_{s.sep}$	$P_{s.cen}$	$P_{mv.n}$
Sum of fractional distances – Coulomb potential[2]									
Uniform	P_s	**131804**	=	(>)	>	>	>	>	>
	$P_{mv.u}$	143219	(<)	=	(<)	(<)	>	>	>
Concentration	$P_{s.b}$	133080	<	(>)	=	(>)	>	>	(>)
	$P_{s.c}$	170617	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	155857	<	<	<	>	=	>	>
	$P_{s.cen}$	226398	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	175378	<	<	(<)	(<)	<	>	=
Sum of fractional distances – Riesz 2-energy [2]									
Uniform	P_s	**1065661**	=	(>)	>	>	>	>	>
	$P_{mv.u}$	2454885	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	1282503	<	(>)	=	(>)	>	>	(>)
	$P_{s.c}$	2407077	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	2229101	<	>	<	>	=	>	>
	$P_{s.cen}$	3064085	<	<	<	<	<	=	>
Suboptimal	$P_{mv.n}$	3570866	<	<	(<)	(<)	<	<	=
Sum of fractional distances – Logarithmic energy[2]									
Uniform	P_s	37688	=	(>)	>	>	>	>	>
	$P_{mv.u}$	38399	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	36737	<	(>)	=	(>)	<	>	(>)
	$P_{s.c}$	42357	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**36406**	<	>	>	>	=	>	>
	$P_{s.cen}$	55805	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	45916	<	<	(<)	(<)	<	>	=
Discrepancy – Wrap-around discrepancy[2]									
Uniform	P_s	0.007	=	(>)	>	>	>	>	>
	$P_{mv.u}$	0.008	(<)	=	(<)	(<)	>	>	>
Concentration	$P_{s.b}$	**0.005**	<	(>)	=	(>)	>	>	(>)
	$P_{s.c}$	0.025	<	(>)	(<)	=	>	>	(>)
Bias to border	$P_{s.sep}$	0.053	<	<	<	<	=	>	<
	$P_{s.cen}$	0.113	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.050	<	<	(<)	(<)	>	>	=

[2] A lower value of this metric signifies a more diverse population

Table A.3: Whether our expectations of how each pair of test populations perform against each other in terms of diversity in convex space are correct. '>' indicates $D(P_{row}) > D(P_{column})$, and '<' indicates the reverse. Entries marked with '(>)' or '(<)' follow from the covering radius distribution property. The green coloured cells are correct, the red ones are incorrect, and the yellow ones are the relations defined using the output diversities. The bold diversity value indicates the highest value.

Type	**Population**	$D(P)$	P_s	$P_{mv.u}$	$P_{s.b}$	$P_{s.c}$	$P_{s.sep}$	$P_{s.cen}$	$P_{mv.n}$
Other – nVol									
	P_s	0.480	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	0.522	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.531	<	(>)	=	(>)	<	>	(>)
	$P_{s.c}$	0.448	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.770**	<	>	>	>	=	>	>
	$P_{s.cen}$	0.178	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.253	<	<	(<)	(<)	<	>	=
Other – Moment of inertia									
	P_s	21.9	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	21.7	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	23.5	<	(>)	=	(>)	<	>	(>)
	$P_{s.c}$	19.0	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**32.3**	<	>	>	>	=	>	>
	$P_{s.cen}$	6.8	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	13.0	<	<	(<)	(<)	<	>	=
Other – True diversity									
	P_s	0.043	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	0.042	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.046	<	(>)	=	(>)	<	>	(>)
	$P_{s.c}$	0.037	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.063**	<	>	>	>	=	>	>
	$P_{s.cen}$	0.013	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.025	<	<	(<)	(<)	<	>	=
Other – Mean normalised standard deviation									
	P_s	0.445	=	(>)	>	>	>	>	>
Uniform	$P_{mv.u}$	0.437	(<)	=	(<)	(<)	<	>	>
Concentration	$P_{s.b}$	0.461	<	(>)	=	(>)	<	>	(>)
	$P_{s.c}$	0.414	<	(>)	(<)	=	<	>	(>)
Bias to border	$P_{s.sep}$	**0.593**	<	>	>	>	=	>	>
	$P_{s.cen}$	0.241	<	<	<	<	<	=	<
Suboptimal	$P_{mv.n}$	0.332	<	<	(<)	(<)	<	>	=

B

GA component comparison results

Setup	Total number of individuals checked for near-optimality per generation at convergence generation						
	Investment space		*Normalised space*				
	MAA diversity	MGA diversity	MAA norm. diversity	MGA norm. diversity	Lenient convergence	Moderate convergence	Strict convergence
Base	33740 ± 695	46580 ± 801	25560 ± 658	46680 ± 650	46900 ± 15500	80700 ± 0	100000 ± 0
Crossover	2065 ± 120	2260 ± 138	2069 ± 93	3589 ± 1723	73 ± 6	5225 ± 1111	10687 ± 2069
Normalisation	2179 ± 101	2431 ± 76	2197 ± 92	2745 ± 82	81 ± 4	8007 ± 1666	11976 ± 1034
Initialisation	241 ± 27	454 ± 80	12 ± 24	390 ± 35	2004 ± 336	6073 ± 1687	9270 ± 1585
Mutation	170 ± 39	339 ± 30	13 ± 24	309 ± 45	2191 ± 256	7613 ± 1439	11053 ± 2871

Table B.1: Number of the offspring needing to be checked for near-optmality on average at each generation for different points of measurement and setups.

Setup	Total number of non-near-optimal individuals per generation at convergence generation						
	Investment space		*Normalised space*				
	MAA diversity	MGA diversity	MAA norm. diversity	MGA norm. diversity	Lenient convergence	Moderate convergence	Strict convergence
Base	4366 ± 203	6640 ± 218	3240 ± 149	6662 ± 127	7463 ± 2952	18462 ± 0	27067 ± 385
Crossover	413 ± 94	487 ± 109	413 ± 83	1251 ± 1037	48 ± 5	2604 ± 912	7234 ± 1658
Normalisation	474 ± 55	590 ± 43	482 ± 48	773 ± 25	49 ± 10	5333 ± 1475	9036 ± 869
Initialisation	94 ± 21	193 ± 50	21 ± 7	160 ± 30	1365 ± 284	5080 ± 1593	8059 ± 1508
Mutation	83 ± 23	165 ± 21	24 ± 9	149 ± 31	1638 ± 222	6631 ± 1344	9866 ± 2686

Table B.2: Test

Bibliography

[1] C. Bradford Barber, David P. Dobkin, and Hannu Huhdanpaa. Qhull: Quickhull algorithm for computing the convex hull. *Astrophysics Source Code Library*, page ascl:1304.016, April 2013.

[2] Philip B. Berntsen and Evelina Trutnevyte. Ensuring diversity of national energy scenarios: Bottom-up energy system model with Modeling to Generate Alternatives. *Energy*, 126:886–898, May 2017. ISSN 0360-5442. doi: 10.1016/j.energy.2017.03.043.

[3] Luca Maria Del Bono, Flavio Nicoletti, and Federico Ricci-Tersenghi. The most uniform distribution of points on the sphere, July 2024.

[4] Miguel Chang, Jakob Zink Thellufsen, Behnam Zakeri, Bryn Pickering, Stefan Pfenninger, Henrik Lund, and Poul Alberg Østergaard. Trends in tools and approaches for modelling the energy transition. *Applied Energy*, 290:116731, May 2021. ISSN 0306-2619. doi: 10.1016/j.apenergy.2021.116731.

[5] Shoou-Yuh Chang, E. Downey Brill Jr., and Lewis D. Hopkins. Efficient Random Generation of Feasible Alternatives: A Land Use Example. *Journal of Regional Science*, 22(3):303–314, 1982. ISSN 1467-9787. doi: 10.1111/j.1467-9787.1982.tb00754.x.

[6] Seolhee Cho, Can Li, and Ignacio E. Grossmann. Recent advances and challenges in optimization models for expansion planning of power systems and reliability optimization. *Computers & Chemical Engineering*, 165:107924, September 2022. ISSN 0098-1354. doi: 10.1016/j.compchemeng.2022.107924.

[7] G. Corriveau, R. Guilbault, A. Tahan, and R. Sabourin. Review and study of genotypic diversity measures for real-coded representations. *IEEE Transactions on Evolutionary Computation*, 16(5):695–710, 2012. doi: 10.1109/TEVC.2011.2170075.

[8] Emilie Danna and David L. Woodruff. How to select a small set of diverse solutions to mixed integer programming problems. *Operations Research Letters*, 37(4):255–260, July 2009. ISSN 0167-6377. doi: 10.1016/j.orl.2009.03.004.

[9] Edwin D. de Jong, Richard A. Watson, and Jordan B. Pollack. Reducing bloat and promoting diversity using multi-objective methods. In *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, GECCO'01, pages 11–18, San Francisco, CA, USA, July 2001. Morgan Kaufmann Publishers Inc. ISBN 978-1-55860-774-3.

[10] J. F. DeCarolis, S. Babaee, B. Li, and S. Kanungo. Modelling to generate alternatives with an energy system optimization model. *Environmental Modelling & Software*, 79:300–310, May 2016. ISSN 1364-8152. doi: 10.1016/j.envsoft.2015.11.019.

[11] Wanru Gao, Samadhi Nallaperuma, and Frank Neumann. Feature-Based Diversity Optimization for Problem Instance Classification, May 2020.

[12] Aleksander Grochowicz, Koen van Greevenbroek, Fred Espen Benth, and Marianne Zeyringer. Intersecting near-optimal spaces: European power systems with more resilience to weather variability. *Energy Economics*, 118:106496, February 2023. ISSN 0140-9883. doi: 10.1016/j.eneco.2022.106496.

[13] Reza Hemmati, Rahmat-Allah Hooshmand, and Amin Khodabakhshian. Comprehensive review of generation and transmission expansion planning. *IET Generation, Transmission & Distribution*, 7(9):955–964, 2013. ISSN 1751-8695. doi: 10.1049/iet-gtd.2013.0031.

[14] Maike Hennen, Matthias Lampe, Philip Voll, and André Bardow. SPREAD – Exploring the decision space in energy systems synthesis. *Computers & Chemical Engineering*, 106:297–308, November 2017. ISSN 0098-1354. doi: 10.1016/j.compchemeng.2017.06.002.

[15] Gordon H. Huang, Jonathan D. Linton, Julian Scott Yeomans, and Reena Yoogalingam. Policy planning under uncertainty: Efficient starting populations for simulation-optimization methods applied to municipal solid waste management. *Journal of Environmental Management*, 77(1):22–34, October 2005. ISSN 0301-4797. doi: 10.1016/j.jenvman.2005.02.008.

[16] Raha Imanirad, Xin-She Yang, and J. Yeomans. Modelling-to-generate-alternatives via the firefly algorithm. 2014.

[17] R Keller and Wolfgang Banzhaf. Explicit maintenance of genetic diversity on genospaces. *Unpublished manuscript. Available online at Citeseer*, 1994.

[18] Joel Lehman and Kenneth O. Stanley. Evolving a diversity of virtual creatures through novelty search and local competition. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, GECCO '11, pages 211–218, New York, NY, USA, July 2011. Association for Computing Machinery. ISBN 978-1-4503-0557-0. doi: 10.1145/2001576.2001606.

[19] F. Lombardi, B. Pickering, E. Colombo, and S. Pfenninger. Policy Decision Support for Renewables Deployment through Spatially Explicit Practically Optimal Alternatives. *Joule*, 4(10):2185–2207, 2020. doi: 10.1016/j.joule.2020.08.002.

[20] F. Lombardi, B. Pickering, and S. Pfenninger. What is redundant and what is not? Computational trade-offs in modelling to generate alternatives for energy infrastructure deployment. *Applied Energy*, 339, 2023. doi: 10.1016/j.apenergy.2023.121002.

[21] Francesco Lombardi and Stefan Pfenninger. Human-in-the-loop MGA to generate energy system design options matching stakeholder needs. *PLOS Climate*, 4(2):e0000560, February 2025. ISSN 2767-3200. doi: 10.1371/journal.pclm.0000560.

[22] Mikhail Mironov and Liudmila Prokhorenkova. Measuring Diversity: Axioms and Challenges. October 2024.

[23] Aneta Neumann, Wanru Gao, Carola Doerr, Frank Neumann, and Markus Wagner. Discrepancy-based evolutionary diversity optimization. In *GECCO '18 - Genetic and Evolutionary Computation Conference*, pages 991–998, Kyoto, France, July 2018. ACM Press. doi: 10.1145/3205455.3205532.

[24] Aneta Neumann, Wanru Gao, Markus Wagner, and Frank Neumann. Evolutionary diversity optimization using multi-objective indicators. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '19, pages 837–845, New York, NY, USA, July 2019. Association for Computing Machinery. ISBN 978-1-4503-6111-8. doi: 10.1145/3321707.3321796.

[25] Fabian Neumann and Tom Brown. The near-optimal feasible space of a renewable power system model. *Electric Power Systems Research*, 190:106690, January 2021. ISSN 0378-7796. doi: 10.1016/j.epsr.2020.106690.

[26] Valentín Osuna-Enciso, Erik Cuevas, and Bernardo Morales Castañeda. A diversity metric for population-based metaheuristic algorithms. *Information Sciences*, 586:192–208, March 2022. ISSN 0020-0255. doi: 10.1016/j.ins.2021.11.073.

[27] Tim T. Pedersen, Marta Victoria, Morten G. Rasmussen, and Gorm B. Andresen. Modeling all alternative solutions for highly renewable energy systems. *Energy*, 234:121294, November 2021. ISSN 0360-5442. doi: 10.1016/j.energy.2021.121294.

[28] Fernando Perez-Cruz. Kullback-Leibler divergence estimation of continuous distributions. In *2008 IEEE International Symposium on Information Theory*, pages 1666–1670, July 2008. doi: 10.1109/ISIT.2008.4595271.

[29] James Price and Ilkka Keppo. Modelling to generate alternatives: A technique to explore uncertainty in energy-environment-economy models. *Applied Energy*, 195:356–369, June 2017. ISSN 0306-2619. doi: 10.1016/j.apenergy.2017.03.065.

[30] Matthias Rainer. A Genetic Algorithm for Mixed-integer Multicriteria Optimization Problems and its Application to Engines in order to Optimize Fuel Consumption and Driving Performance. 2012.

[31] Pedro Ramaciotti Morales, Robin Lamarche-Perrin, Raphaël Fournier-S'niehotta, Rémy Poulain, Lionel Tabourier, and Fabien Tarissan. Measuring diversity in heterogeneous information networks. *Theoretical Computer Science*, 859:80–115, March 2021. ISSN 0304-3975. doi: 10.1016/j.tcs.2021.01.013.

[32] Sebastian Risi, Sandy D. Vanderbleek, Charles E. Hughes, and Kenneth O. Stanley. How novelty search escapes the deceptive trap of learning to learn. In *Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, GECCO '09, pages 153–160, New York, NY, USA, July 2009. Association for Computing Machinery. ISBN 978-1-60558-325-9. doi: 10.1145/1569901.1569923.

[33] Hendrik Schricker, Benedikt Schuler, Christiane Reinert, and Niklas von der Aßen. Gotta catch 'em all: Modeling All Discrete Alternatives for Industrial Energy System Transitions, July 2023.

[34] Bruno U. Schyska, Alexander Kies, Markus Schlott, Lueder von Bremen, and Wided Medjroubi. The sensitivity of power system expansion models. *Joule*, 5(10):2606–2624, October 2021. ISSN 2542-4351. doi: 10.1016/j.joule.2021.07.017.

[35] Andrew R. Solow and Stephen Polasky. Measuring biological diversity. *Environmental and Ecological Statistics*, 1(2):95–103, June 1994. ISSN 1573-3009. doi: 10.1007/BF02426650.

[36] Giovanni Squillero and Alberto Tonda. Divergence of character and premature convergence: A survey of methodologies for promoting diversity in evolutionary optimization. *Information Sciences*, 329:782–799, February 2016. ISSN 0020-0255. doi: 10.1016/j.ins.2015.09.056.

[37] Andy Stirling. A general framework for analysing diversity in science, technology and society. *Journal of The Royal Society Interface*, 4(15):707–719, February 2007. doi: 10.1098/rsif.2007.0213.

[38] Stefan Strömer and Klara Maggauer. IESopt: A Modular Framework for High-Performance Energy System Optimization. In *2024 Open Source Modelling and Simulation of Energy Systems (OSMSES)*, pages 1–6, September 2024. doi: 10.1109/OSMSES62085.2024.10668965.

[39] Demet Suna, Gustav Resch, Franziska Schniger, Florian Hasengst, Gerhard Totschnig, Peter Widhalm, Herbert Formayer, Philipp Maier, David Leidinger, and Imran Nadeem. Securing Austria's Electricity Supply in Times of Climate Change. *Climate.Changes.Security.*, 2, June 2024.

[40] Tamara Ulrich and Lothar Thiele. Maximizing population diversity in single-objective optimization. In *Proceedings of the 13th Annual Conference on Genetic and Evolutionary Computation*, GECCO '11, pages 641–648, New York, NY, USA, July 2011. Association for Computing Machinery. ISBN 978-1-4503-0557-0. doi: 10.1145/2001576.2001665.

[41] Tamara Ulrich, Johannes Bader, and Lothar Thiele. Defining and Optimizing Indicator-Based Diversity Measures in Multiobjective Search. In *Parallel Problem Solving from Nature, PPSN XI*, pages 707–717. Springer, Berlin, Heidelberg, 2010. ISBN 978-3-642-15844-5. doi: 10.1007/978-3-642-15844-5_71.

[42] Yirui Wang, Shangce Gao, Mengchu Zhou, and Yang Yu. A multi-layered gravitational search algorithm for function optimization and real-world problems. *IEEE/CAA Journal of Automatica Sinica*, 8(1):94–109, January 2021. ISSN 2329-9274. doi: 10.1109/JAS.2020.1003462.

[43] Keigo Watanabe and M. M. A. Hashem. *Evolutionary Computations*, volume 147 of *Studies in Fuzziness and Soft Computing*. Springer, Berlin, Heidelberg, 2004. ISBN 978-3-642-05887-5 978-3-540-39883-7. doi: 10.1007/978-3-540-39883-7.

[44] Yutong Xie, Ziqiao Xu, Jiaqi Ma, and Qiaozhu Mei. How Much Space Has Been Explored? Measuring the Chemical Space Covered by Databases and Machine-Generated Molecules, March 2023.

[45] Julian Scott Yeomans. Efficient generation of alternative perspectives in public environmental policy formulation: Applying co-evolutionary simulation–optimization to municipal solid waste management. *Central European Journal of Operations Research*, 19(4):391–413, December 2011. ISSN 1435-246X, 1613-9178. doi: 10.1007/s10100-011-0190-y.

[46] Emily M. Zechman and S. Ranji Ranjithan. An evolutionary algorithm to generate alternatives (EAGA) for engineering optimization problems. *Engineering Optimization*, 36(5):539–553, October 2004. ISSN 0305-215X. doi: 10.1080/03052150410001704863.

[47] Emily M Zechman, S Ranji Ranjithan, and Li Liu. Niched Co-Evolution Strategies to Address Non-uniqueness in Engineering Design. 2006.

[48] Yong-Dao Zhou, Kai-Tai Fang, and Jian-Hui Ning. Mixture discrepancy for quasi-random point sets. *Journal of Complexity*, 29(3):283–301, June 2013. ISSN 0885-064X. doi: 10.1016/j.jco.2012.11.006.