

Article

A Genetic Algorithm for Site-Specific Management Zone Delineation

Francisco Huguet ^{1,2} , Lluís M. Plà-Aragónés ^{1,3,*} , Víctor M. Albornoz ⁴  and Mauricio Pohl ² ¹ Department de Matemàtica, Universitat de Lleida, c/ Jaume II, 73, 25003 Lleida, Spain; fhuguet@uca.edu.sv² Department of Electronics and Informatics, Universidad Centroamericana UCA, Bulevar Los Próceres, Antiguo Cuscatlán, La Libertad 01-168, El Salvador; mapohl@uca.edu.sv³ Agrotecnio Center, Universitat de Lleida, 25198 Lleida, Spain⁴ Departamento de Industrias, Campus Santiago Vitacura, Universidad Técnica Federico Santa María, Av. Santa María 6400, Santiago 7650568, Chile; victor.albornoz@usm.cl

* Correspondence: limpla@matematica.udl.cat; Tel.: +34-973703318

Abstract: This paper presents a genetic algorithm-based methodology to address the Site-Specific Management Zone (SSMZ) delineation problem. A SSMZ is a subregion of a field that is homogeneous with respect to a soil or crop property, enabling farmers to apply customized management strategies for optimizing resource use. The algorithm generates optimized field partitions using rectangular zones, applicable to both regular and irregularly shaped fields. To the best of our knowledge, the Genetic Algorithm for Zone Delineation (GAZD) is the first approach to handle the rectangular SSMZ delineation problem in irregular-shaped lands without introducing non-real data. The algorithm's performance is compared with an exact solution based on integer linear programming. Experimental tests conducted on real-field and generated irregular-shaped instances show that while the GAZD requires longer execution times than the exact approach, it proves to be functional and robust in solving the SSMZ problem. Furthermore, the GAZD offers a set of “good enough” solutions that can be evaluated for feasibility and practical convenience, making it a valuable tool for decision-making processes. Moreover, strategies such as implementation in a compiled language and parallel processing can be used to improve the execution time performance of the algorithm.



Academic Editors: Igor Litvinchev,
Tetyana Romanova and
Socorro Rangel

Received: 11 February 2025

Revised: 13 March 2025

Accepted: 21 March 2025

Published: 25 March 2025

Citation: Huguet, F.; Plà-Aragónés, L.M.; Albornoz, V.M.; Pohl, M. A Genetic Algorithm for Site-Specific Management Zone Delineation. *Mathematics* **2025**, *13*, 1064. <https://doi.org/10.3390/math13071064>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: genetic algorithms; management zone delineation; operations research; agriculture

MSC: 90-08; 90C10; 90B80; 90C59; 68W50

1. Introduction

In modern agriculture, crop and soil properties are collected, processed, and analyzed both temporally and spatially, integrating additional information to support decision-making. By addressing variability within agricultural systems, it is possible to improve resource efficiency, productivity, profitability, and the sustainability of agricultural production. Data retrieved by soil sampling, drone and satellite imagery, remote sensors, raster data, and other information technologies are used to gather information about issues such as heterogeneity of chemical and physical soil properties, crop development, and climate variability. This information can be used as inputs to decision support systems to assist farmers in decisions such as targeted specific water and nutrient needs, yield prediction, and crop harvest management.

Site-Specific Management Zones (SSMZs) are defined as areas within a field that share similar characteristics, identified using data from sources such as soil sampling, remote sensing, or yield monitoring. By dividing fields into these zones, farmers can implement customized management strategies, optimizing the application of resources like fertilizers, water, and seeds. This targeted approach reduces waste, enhances productivity, and supports sustainable farming practices by addressing the unique needs of each zone. Several methods have been proposed to address the management zone delineation problem. Clustering techniques based on information such as soil properties, yield maps, historical seasonal data, and combined factors have been widely studied [1–12]. These approaches often result in irregularly shaped zones, which can hinder their practical implementation by farmers.

To overcome this limitation, several models for delineating rectangular and orthogonal management zones have been proposed using mathematical programming, heuristic, and metaheuristic techniques. Rectangular and orthogonal partitions are generally more compatible with agricultural machinery, improving their functionality in practice. Such models typically define field partitions based on agricultural field data, an objective function, e.g., minimizing the number of zones or minimizing the sum of variances within the zones, subject to a required homogeneity index.

For instance, in [13], a binary integer linear programming (BILP) model was proposed for delineating rectangular and homogeneous management zones, minimizing the sum of variances for a soil property. Albornoz et al. [14] introduced a bi-objective mixed-integer linear programming model to simultaneously minimize the number of management zones and maximize the relative homogeneity of the partition. Later, Albornoz et al. [15] presented a robust mixed-integer optimization model that considers the spatial and temporal variability of vegetation or soil indices, combining it with a column generation algorithm for solving the model. Additionally, more recently Albornoz et al. [16] developed a linear binary integer program for integrated zoning and crop planning with adjacency constraints, solved using a decomposition-based heuristic. Integrated approaches for the delineation of rectangular management zones in crop planning problems under both deterministic and uncertain conditions were proposed in [17] and [18], respectively. Finally, orthogonal management zone delineation is approached using a greedy heuristics algorithm in [19] and estimation of distribution algorithms in [20,21].

Initially, the aforementioned models require as input a dataset that describes the variability of a property across a rectangular field, i.e., soil or crop property values obtained from equidistant samples. Each of these equidistant samples characterize small square area units, defining a perfect grid within the rectangular fields. When fields are not rectangular (i.e., when samples and their respective area units do not form a perfect grid), equidistant dummy samples are introduced to complete the rectangular shape of the field. The handling of property values for these dummy samples varies among authors. For instance, authors in [13,14] assigned to dummy samples high property values, relative to real samples. This approach ensures that dummy samples are grouped separately from real samples, allowing them to be excluded from the final partition afterwards. Conversely, Velasco et al. [20] assigned dummy samples the value of their nearest neighboring sample. This method prevents the formation of management zones composed solely of dummy samples, which are also removed from the final field partition. The risk with these approaches for handling irregularly shaped fields is that the insertion of arbitrary values to inexistent samples alter the agricultural field description and has an impact in global and local descriptors, such as the total variance of the field and the internal variance of zones containing dummy samples involved in the optimization process. In these cases, the distortion in the field description becomes accentuated as the number of dummy samples increases. The objective

of this paper is to present a new methodology based on genetic algorithms to approach the rectangular management zone delineation problem, in both rectangular and irregularly shaped agricultural field, using only real sample data.

The remainder of this article is organized as follows: Section 2 presents Materials and Methods, including a general description of the genetic algorithm, and the benchmark algorithm and instances used to evaluate the performance. Section 3 describes and discusses the experimental results. Conclusions and recommendations are presented in Section 4.

2. Materials and Methods

2.1. Mathematical Description

The SSMZ delineation problem requires as input the description of the variability of a soil or crop property across the agricultural field. This information may be provided by procedures such as soil sampling or analysis of drone or satellite imagery. These data are used to produce field partitions according to an optimization objective and constraints. The goal of the proposed Genetic Algorithm for Zone Delineation (GAZD) is to find partitions having the minimum number of exclusively rectangular zones while achieving an homogeneity requirement. In this study, relative variance is used as the homogeneity criterion, as it has been demonstrated to be a reliable index for evaluating the efficiency of zoning methods [10]. Relative variance ranges between 0 and 1, with values exceeding 0.5 ensuring homogeneous partitions [13].

For the GAZD, the description of the heterogeneity of a property is provided by a set of equidistant samples defining a rectangular instance bounded by the leftmost, rightmost, uppermost, and bottommost samples. This rectangular instance is divided into rectangular area units (AUs), which may or may not contain a sample. Each AU is identified by its position (row, column) within a rectangular grid. The entire instance is represented as a matrix, where sampled AUs are represented by the value of the measured property and non-sampled AU by zero. The former are considered available for the SSMZ delineation problem and the latter excluded. Figure 1a shows an example of an irregularly shaped agricultural field characterized by 26 samples. The value of each sample is represented by the letter “S” with a subindex. Figure 1b shows the field divided by a grid of 35 AUs, and Figure 1c shows the corresponding field matrix.

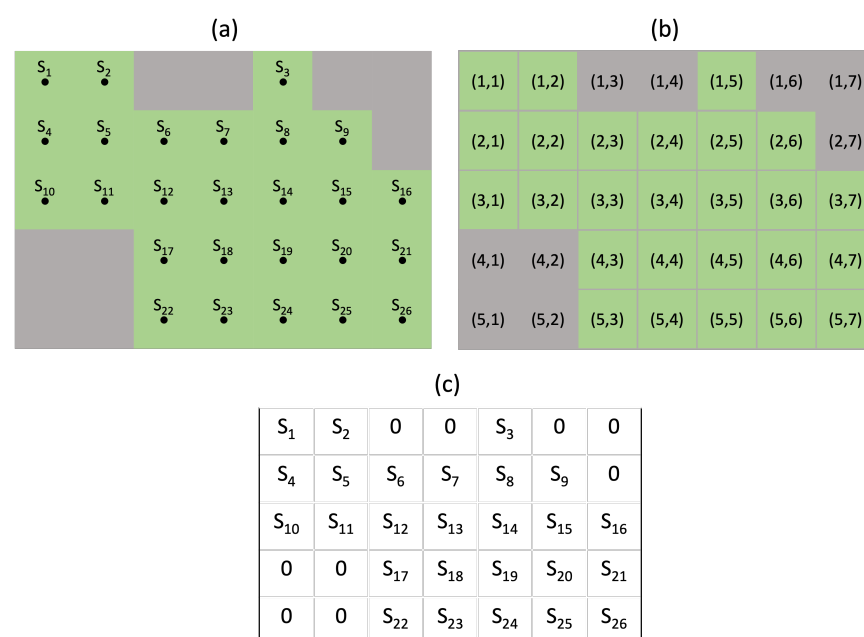


Figure 1. Example of irregularly shaped field. (a) Samples, (b) Division in AUs, (c) Field matrix.

We let R be the set of AUs dividing a land instance according to

$$R = \{(i, j) | 1 \leq i \leq m, 1 \leq j \leq n\} \quad (1)$$

where m and n are the number of rows and columns of the grid, respectively, and (i, j) is the position of the AU. The AUs characterized by the value of a sample are comprised in the subset Q ($Q \subseteq R$).

A candidate solution for the SSMZ delineation problem having N zones is represented by x in (2):

$$x = \{z_1, z_2, \dots, z_N\} \quad (2)$$

where all zones in the solution are rectangular:

$$z_k = \{(i, j) | i_{\min,k} \leq i \leq i_{\max,k}, j_{\min,k} \leq j \leq j_{\max,k}\} \quad \text{for } k = 1, 2, \dots, N \quad (3)$$

The objective function of the combinatorial optimization problem is defined by

$$\min f(x) = |x| = N \quad (4)$$

$$\text{s.t.} \quad \bigcup_{k=1}^N z_k = Q \quad (5)$$

$$z_k \cap z_l = \emptyset \quad \forall k \neq l \quad (6)$$

$$\left(1 - \frac{\sum_{k \in N} (n_k - 1) \sigma_k^2}{\sigma_T^2 (|Q| - N)}\right) \geq \alpha \quad (7)$$

Constraint (5) corresponds to complete coverage of uniquely sampled area units, (6) corresponds to zone disjunction. Finally, Constraint (7) refers to the homogeneity requirement: the relative variance of the partition (left hand of the equation) must be equal or superior to a required value (right hand of the equation). The relative variance is calculated using the total variance of the set of sampled AUs σ_T^2 , the variance of each zone of the partition σ_k^2 , the number of sampled AUs in each zone n_k , the number of sampled AUs $|Q|$, and the number of zones N .

2.2. Genetic Algorithms

Genetic algorithms, introduced in [22], are optimization and search techniques inspired by the principles of natural selection and evolution. They belong to the broader category of evolutionary algorithms particularly effective for solving complex problems, such as non-convex problems, problems with discontinuities, problems having huge search space, and problems with multiple and conflicting objectives. In genetic algorithms, potential solutions to a problem are encoded and represented as individuals in a population. These individuals evolve over successive generations. The evolution is guided by genetic operators such as selection, crossover, and mutation, which mimic biological processes. The goal of the evolution process is to iteratively improve the population to find the best or near-optimal solution to the given problem. The process begins with the generation of an initial population of solutions, typically created randomly. Each solution is evaluated using a fitness function, which measures how well it satisfies the objective of the problem. Solutions with higher fitness have a greater chance of being selected as “parents” to create a new generation by means of genetic operators. Through crossover, parents exchange information to create offspring, while mutation introduces random changes to maintain diversity and avoid premature convergence. This iterative process continues until a termination condition

is met, such as reaching a predefined number of generations or achieving a satisfactory fitness level.

2.3. Genetic Algorithm for Rectangular Management Zone Delineation

2.3.1. General Description of GAZD

The algorithm's main inputs include the field matrix, the maximum and minimum number of zones a solution can have, the minimum zone size in terms of area units (AUs), the minimum relative variance required for a solution, the size of the initial population, the number of crossovers and mutations per generation, and the total number of generations in the evolutionary process.

Figure 2 provides an overview of the key stages of the GAZD. The process begins with the generation of an initial population of diverse field partitions, each containing a number of zones within the specified bounds and with a minimum zone size meeting or exceeding the defined threshold. Each solution is initially evaluated based on its relative variance. Solutions meeting or exceeding the required relative variance are selected for ranking, where they are ordered based on the number of zones. Solutions with fewer zones receive higher rankings and are more likely to be chosen during parent selection.

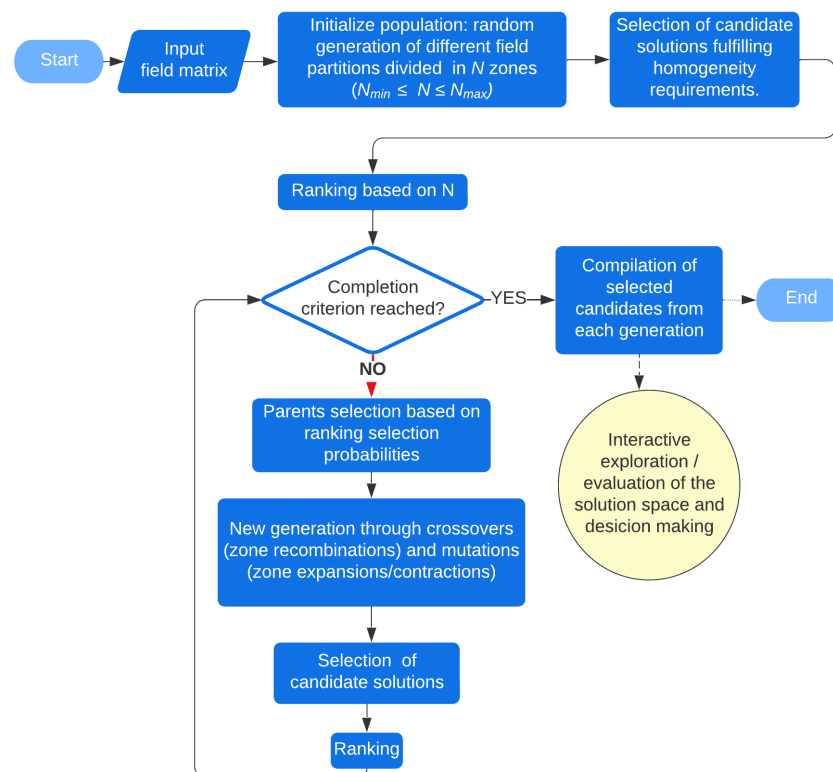


Figure 2. General scheme of the Genetic Algorithm for Zone Delineation (GAZD).

The selected parents then undergo genetic operations to create a new generation of solutions. These operations include crossover (information exchange between two solutions) and mutation (modifications to a single solution), producing new offspring. The resulting generation is evaluated following the same steps: pre-selection based on relative variance, ranking, and parent selection. The selected parents generate the next generation through further genetic operations. This iterative process continues until the specified number of generations is reached. Throughout the evolutionary process, a record of selected solutions is maintained. The final output consists of all selected solutions generated during the process.

2.3.2. Genotype and Phenotype

Solutions are encoded as lists of sub-lists. Each sub-list corresponds to a rectangular zone and contains information about the area units (AUs) that comprise it. The top-level list represents the partition of the agricultural field, with its length indicating the number of zones in the partition. In the genotype representation, the elements of the sub-lists are the positions (i, j) of the area units included in each zone. In the phenotype representation, the elements are the corresponding sample values. Figure 3 illustrates an example of a field partition into 8 zones. Both the genotype and phenotype are represented as 8 sub-lists, each corresponding to a single zone. The genotype lists contain the positions of the AUs in a compact index notation for matrix positions, while the phenotype lists contain the sample values, denoted by S with a subscript.

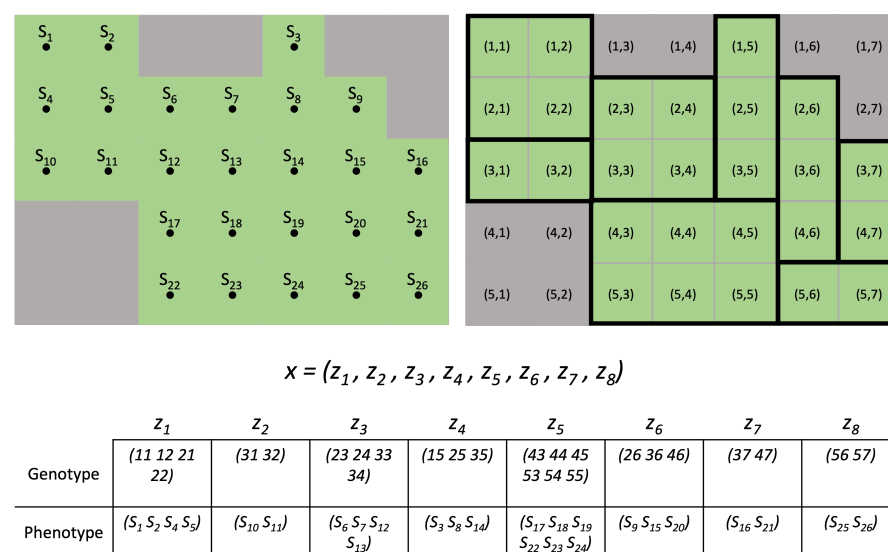


Figure 3. Genotype and phenotype representation.

2.3.3. Initial Population

The initial population consists of a list of randomly generated solutions. The procedure to construct an aleatory solution begins by defining the number of zones in which the agricultural field is divided. This number is randomly chosen within the predetermined minimum and maximum bounds. Next, for each zone, a randomly selected AU is used to initialize it. The initialization is followed by an iterative zone expansion routine in which a search and selection process of available neighbor AUs for potential horizontal or vertical rectangular expansion takes place. This process occurs alternately for the set of zones and ends when the available AUs run out. The construction process of aleatory solutions is repeated as many times as the number of individuals required by the initial population. The size of the initial population and the bounds for the minimum and maximum number of zones are closely tied to the diversity of the initial population. High diversity is particularly beneficial when the problem demands elevated levels of homogeneity, such as achieving a high value of relative variance. The initial population size must ensure the inclusion of viable solutions that can serve as a foundation for the evolutionary process.

2.3.4. Fitness Function, Ranking, and Parent Selection

Solutions are evaluated according to two criteria: their relative variance and the number of zones. Only solutions with a relative variance equal to or greater than the required threshold are selected as potential parents for generating the next generation. Once selected, these candidate parents are ranked based on their number of zones, which serves as the fitness function, with solutions having fewer zones ranked higher than

those with more zones. A rank-based selection probability method is employed to choose solutions for the genetic operations stage, with higher-ranked solutions having a greater probability of being selected.

2.3.5. Genetic Operations

Crossovers and mutations of the selected parents are used to generate a new generation of solutions. In crossovers, a new solution is created by recombining a “donor” solution with a “recipient” solution. A randomly selected zone from the donor is inserted into a copy of the recipient solution. Overlapping zones in the recipient copy may reconfigure into smaller zones. If they cannot reconfigure while maintaining a rectangular shape, they disappear, leaving their AUs available to be absorbed by neighboring zones or incorporated into new zones.

Mutations are conducted by expanding or contracting a randomly selected zone of a parent solution. The contraction or expansion is made while assuring a final rectangular shape. Both operation may lead to a reconfiguration process where some AUs are left available to be absorbed by neighboring zones or to create a new zone. Expanding a zone leads to a similar reconfiguration problem as in the crossovers. Overlapped existent zones by the expanded one may disappear if they cannot be adapted to a smaller size and rectangular shape.

2.3.6. Solution Space

When the model reaches the predetermined number of iterations, it outputs a compilation of the selected solutions—those that satisfied the homogeneity constraint in each generation—over the entire evolutionary process. Each solution is documented with its genotype, phenotype, relative variance, and additional details such as the generation it belongs to and its ranking position. Data files containing this information are generated and subsequently imported into a software application specifically designed for the interactive exploration of the solution space. This software provides interactive visualizations of the solutions and the evolution process, enabling users to evaluate the suitability of different solutions and analyze the development of the evolutionary process.

2.4. Test Materials

2.4.1. Benchmark Algorithm

GAZD’s performance is compared to a BILP model based on [13], using two real-field case studies and one involving irregularly shaped generated instances.

2.4.2. Real-Field Instances

Two real case studies are used to evaluate the algorithm performance. The first is a real-field instance presented in [13], called “Quilaco”. This is used to make a first comparison between the BILP and the GAZD performances. This instance represents an agricultural field divided into 42 AUs, 40 of which are characterized by soil samples. These soil samples provide data on chemical properties such as organic matter, pH, phosphorus, and the sum of bases. The two unsampled AUs are incorporated into the BILP analysis using dummy samples, while they are excluded from the GAZD analysis. The performance data for the BILP model on this instance is sourced from [20]. The GAZD experiments were conducted on a 2020 MacBook Pro with 16 GB of 3733 MHz RAM and a 2 GHz quad-core Intel Core i5 processor, running macOS 13.6. The GAZD was implemented on Python 3.12.1 with tcl/tk 8.6.13.

The second real case corresponds to a set of 12 rectangular-shaped instances used in [15]. The instances were extracted from a set of NDVI vegetation index samples from a table grape field during the 2014–2015 season in Los Andes, a commune in Valparaíso,

Chile. This index can be used to predict crop yields and delineate harvest management zones. Instances 1 to 11 are subsets of Instance 12. The size of the instances ranges from 42 to 380 samples. Instances 1–6 are used to complement the comparative with the BILP, and the larger instances (7–12) are used to extend the scalability analysis of the GAZD in terms of execution time. The experiments were conducted by running both algorithms on the aforementioned system. For the BILP model, the AMPL algebraic modeling language was used, with CPLEX 22.1.1 as the optimization solver.

2.4.3. Irregularly Shaped Generated Instances

To evaluate GAZD's performance in the SSMZ delineation problem for irregularly shaped instances, modified versions of real Instances 4, 5, and 6 were used as inputs. The irregularly shaped versions were created by excluding samples from the originals. Figure 4 shows the modified versions, where sampled AUs are represented by green squares and unsampled AUs (corresponding to excluded samples) by gray squares. Notice that original and modified instances contain the same amount of AUs and their difference strive in the number of sampled AUs. The modified instances contain 67, 87, and 103 sampled AUs (Figure 4a–c). The shape irregularities in all modified instances can be described as a diagonal border on the upper right side and an orthogonal border on the lower left side. Additionally, modified Instance 6 features internal irregularities, forming a square zone of non-disposable AUs for agricultural use. The tests were conducted on the same computational system mentioned in the previous section.

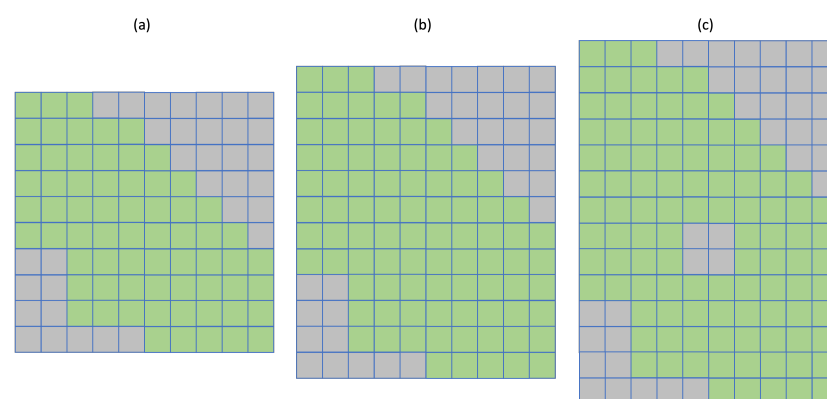


Figure 4. Irregularly shaped instances. (a) Modified instance 4, (b) Modified instance 5, (c) Modified instance 6. Sampled AUs are in green.

3. Experimental Results and Discussion

3.1. Algorithm Parameter Tuning

Several tests were conducted to assess the algorithm's sensitivity to input parameters, such as initial population size, the share of genetic operations, and the number of evolutionary iterations. Based on these tests, reference values were established and later refined for each optimization case through additional experiments to enhance the algorithm's tuning. The minimum and maximum number of zones allowed in a field partition— N_{min} and N_{max} in Figure 2—were set to one and the number of sampled AUs, respectively, for each optimization case.

3.1.1. Initial Population Size

The evolutionary process can only occur if the initial population includes solutions that meet the homogeneity requirements. Therefore, the population size must be large enough to ensure sufficient diversity, including individuals with a relative variance equal to or greater than the required threshold. The appropriate size of the initial population

is influenced by instance characteristics such as heterogeneity and size. To evaluate the impact of these factors, tests were conducted for the different study cases. These tests involved varying the initial population size, assessing the percentage of successful iteration starts over a fixed number of trials, and analyzing the effects of changing homogeneity requirements and instance sizes.

Table 1 shows how the initial population size required for a successful start of the evolutionary process depends on the relative variance (RV) requirement for organic matter and pH properties in the “Quilaco” instance. For organic matter, the required population size increases as the RV requirement grows, with a more pronounced rise when the RV meets 0.9. In contrast, for pH, a small population size is sufficient to initiate the optimization process. This difference is related to the heterogeneity of the properties within the instance: the total variance of organic matter (4.45) is significantly higher than that of pH (0.012).

Table 1. Initial population for “Quilaco” organic matter (OM) and soil acidity (pH) properties.

Relative Variance	0.1	0.3	0.5	0.7	0.9
OM	10	10	30	30	230
pH	10	10	10	10	20

Table 2 shows how the instance size affects the initial population size in the group of Instance 1 to 12 for a relative variance requirement of 0.9. The figure shows a non-linear relationship between both parameters consisting of a first phase (Instances 1 to 5), where the size of the initial population increases as the number of sample increases, and a second phase (Instances 6 to 12), where the population size decreases and stabilizes as the number of samples grows. In the first phase (smaller instances), as the instance size increases, the size of the population increases. This is because in smaller instances, the random solutions tend to create more smaller zones—zones that group fewer samples—resulting in greater heterogeneity between zones and lower relative variance. To achieve a high relative variance, more aleatory solutions are necessary due to this increased variability. In the second phase (for larger instances), the required number of solutions decreases as the instance size grows. This is because in larger instances, the random solutions tend to create larger zones that group more samples, leading to greater homogeneity and a higher relative variance. As the number of zones decreases, fewer solutions are needed to achieve a high relative variance. The transition between these two phases suggests a critical point, where spatial averaging starts to dominate, making homogeneous solutions easier to obtain.

Table 2. Initial population vs. Instance size. Instances 1 to 12.

Instance	1	2	3	4	5	6	7	8	9	10	11	12
Number of Samples	42	60	80	100	120	140	180	220	260	300	340	380
Size of Initial population	40	90	90	110	140	120	50	30	10	10	10	10

3.1.2. Mutation and Crossover Share

The influence of genetic operations was investigated by applying different proportions of mutations and crossovers to various optimization cases. Over a fixed number of genetic operations—defined as the size of the new generations in the evolutionary process—seven mutation–crossover shares (0–100%, 20–80%, 40–60%, 50–50%, 60–40%, 80–20%, and 100–0%) were tested. Each optimization test was repeated a fixed number of times, allowing for the collection of the minimal, maximal, and average best solutions from

the set of optimizations. Additionally, optimal solutions were calculated using the BILP model for each case.

Figure 5 presents the GAZD results for Instance 1 and Figure 6 presents results for Instance 4, both under a relative variance requirement of 0.9. For each mutation–crossover share, the figures display the minimal, average, and maximal values from the set of executions. The BILP optimal solution is 13 zones for Instance 1 and 31 zones for Instance 4.

As shown, the absence of mutations leads to convergence toward local optima, resulting in significant deviations of the GAZD results from the global optimum and a wider range of variation around it. A mutation share of 40% or higher leads to solutions that approach the global optimum and reduces variability around it.

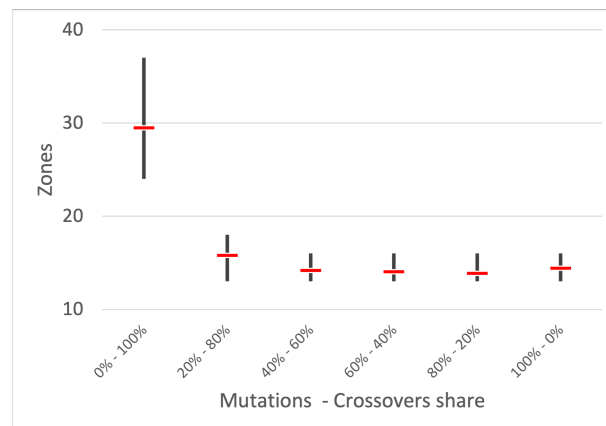


Figure 5. Influence of mutation and crossover rates on optimization performance. Instance 1. $R_v = 0.9$. The red lines represent the average best solution, while the black lines indicate its range of variation across 25 executions.

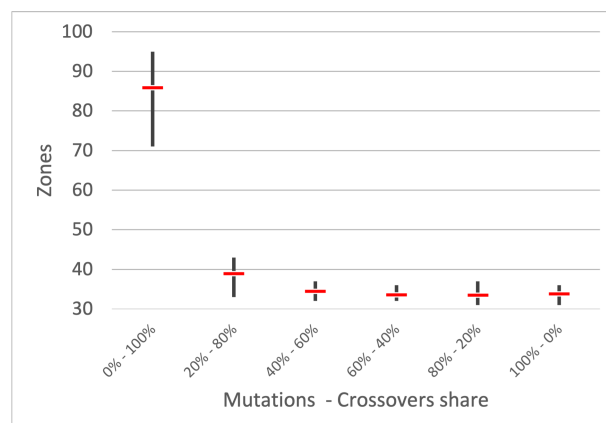


Figure 6. Influence of mutation and crossover rates on optimization performance. Instance 4. $R_v = 0.9$. The red lines represent the average best solution, while the black lines indicate its range of variation across 25 executions.

3.1.3. Finalization Criterion

The end of the optimization process is determined by the number of iterations of the evolution process. This directly affects the possibilities to converge to a local or global optimal. As in the initial population size case, the appropriate setting of this parameter may be affected by characteristics of the study case such as the size of the instance and heterogeneity. To determine the appropriate number of generations for each optimization case, different values of the finalization criterion were tested. Figures 7 and 8 shows the evolution profile for Instances 3 and 6 under a relative variance requirement of 0.7. The figures display the best solutions—measured by the number of zones—across generations. Reaching solutions with fewer zones requires more iterations for Instance 6 (140 sam-

ples) than for Instance 3 (89 samples). As instance size increases, so does the number of generations needed to achieve solutions with fewer zones.

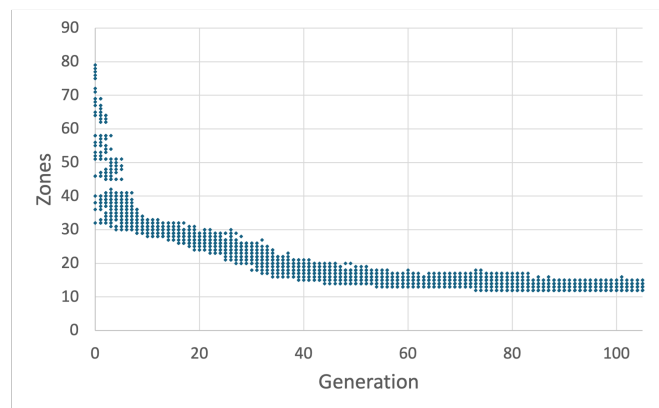


Figure 7. Number of generations. Instance 3. $R_v = 0.7$.

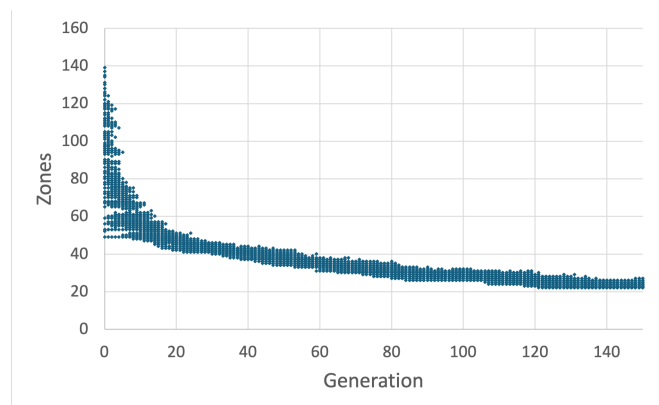


Figure 8. Number of generations. Instance 6. $R_v = 0.7$.

3.2. “Quilaco” Instance

The genetic algorithm was evaluated using the real-field instance “Quilaco” described in Section 2.4. Table 3 presents the experimental results comparing the GAZD approach with the BILP model. For each property, relative variance values ranging from 0.5 to 1 were used to test the algorithms, resulting in a total of 24 optimization cases. The first column lists the chemical soil properties: organic matter (OM), pH, phosphorus (P), and sum of bases (SB). The second column specifies the homogeneity level of the management zones, represented by the required relative variance. Column 3 shows the optimal solution, in terms of the number of zones, obtained by the BILP model. Columns 4–6 present the results for the genetic algorithm, reporting the minimum, average, and maximum number of zones among the best solutions found across 50 independent runs of the algorithm. The color-highlighted results indicate cases where the GAZD achieved outcomes equal to or better than those of the BILP. As the two algorithms were executed on different computers for this test, execution times are not included in the table, nor is a direct comparison provided. However, time performance is briefly discussed at the end of the section to provide an overview of the differences.

Within each property optimization set, the GAZD exhibited a consistent relationship between the homogeneity constraint and the number of zones: as the relative variance requirement increases, the number of zones in the partitions also tends to increase. The range, i.e., the difference between the maximum and minimum number of zones in the best solutions, varies between zero and five.

Table 3. Comparison of BILP and AG-SSMZ Results for “Quilaco” instance. Highlighted results show cases where GAZD matched or outperformed BILP.

Soil Property	Relative Variance	BILP	AG-SSMZ		
		Zones	Min.	Avg.	Max.
OM	1	40	40	40	40
	0.9	20	20	21.44	23
	0.8	17	17	18.34	21
	0.7	14	14	15.1	17
	0.6	11	11	12.28	15
	0.5	9	8	9.86	12
pH	1	24	22	23.02	25
	0.9	17	16	18.02	21
	0.8	10	12	13.56	16
	0.7	7	9	10.14	12
	0.6	5	7	7.74	10
	0.5	4	5	6.28	8
P	1	33	33	33	33
	0.9	9	11	12.1	14
	0.8	5	7	7.8	10
	0.7	3	5	5.66	8
	0.6	3	4	4	4
	0.5	3	4	4	4
SB	1	40	40	40	40
	0.9	20	19	20.84	23
	0.8	16	16	17.48	20
	0.7	12	12	13.34	15
	0.6	9	8	9.94	12
	0.5	7	7	7.74	9

Nevertheless, the average range across all optimization cases is 2.71 zones, demonstrating the robustness of the GAZD, as the best solutions do not vary significantly between executions. In more than 50% of the optimizations (15 out of 24), the GAZD found solutions equal to or better than those of the BILP, particularly for cases involving organic matter and sum of bases properties. For the pH and phosphorus optimization sets, the genetic algorithm generally produced solutions that were slightly inferior but still close to those of the BILP. The fact that in several cases the GAZD “outperformed” the optimal solutions produced by the BILP can be attributed to differences in how the models handle the field description. While the BILP requires the addition of two dummy samples to complete a rectangular field shape—thereby assigning non-real property values to the field description—the genetic algorithm excludes these areas from the optimization process. As a result, the two models are effectively dealing with different field representations.

BILP execution times (see [20] for details on software and hardware) range from 0.01 to 0.2 s, while GAZD execution times range from 5.57 to 21.25 s. Higher execution times in the GAZD are associated with cases requiring high homogeneity levels, where larger initial population sizes are necessary to ensure sufficient diversity and the presence of well-suited solutions to support the following stages of the evolutionary process.

3.3. Instances 1–6

Table 4 presents the experimental results comparing the BILP and GAZD approaches for the set of Instances 1–6. For each instance, three different relative variance values (0.5, 0.7, 0.9) were used to test the algorithms, resulting in a total of 18 optimization cases. The first column lists the instance index, while the second column shows the instance size in terms of the number of samples. The third column specifies the required homogeneity level

of the partition. Columns 4–5 present the results and execution times for the BILP, while Columns 6–9 display the results for the GAZD across 50 independent runs, along with their average execution time. Cases where the GAZD achieved results equal to or better than those of the BILP are highlighted in color.

Table 4. Comparison of BILP and AG-SSMZ Results for Instances 1–6. Highlighted results show cases where GAZD matched BILP.

Generated Instance	Samples	Relative Variance	BILP		AG-SSMZ			
			Zones	Time (Seconds)	Min.	Avg.	Max.	Time (Seconds)
Instance 1	42	0.9	13	0.078212	13	13.98	15	8.94824
		0.7	5	0.091649	5	5.4	7	11.58368
		0.5	3	0.460792	3	3	3	10.37918
Instance 2	60	0.9	18	0.108554	19	20.26	23	22.63273
		0.7	8	0.159806	8	9.92	12	17.99779
		0.5	4	0.589699	4	4.3	6	17.61462
Instance 3	80	0.9	23	0.193209	24	25.9	28	38.39443
		0.7	9	0.229263	9	11.94	15	53.1683
		0.5	5	0.601512	5	6.38	9	46.433
Instance 4	100	0.9	30	0.315366	30	33.8	36	117.599
		0.7	13	0.347363	15	17.38	21	115.431
		0.5	6	0.880585	6	8.04	10	104.3865
Instance 5	120	0.9	35	0.409061	38	40.68	45	184.6348
		0.7	15	0.673061	16	19.1	21	252.6944
		0.5	7	1.7285	7	8.98	11	191.8161
Instance 6	140	0.9	38	0.772159	41	44.36	47	261.6986
		0.7	15	0.953882	18	21.8	26	249.4993
		0.5	7	1.69942	7	8.48	12	375.4883

Since no dummy samples were required, the field descriptions used by both models were identical, allowing for a direct comparison of their results. In this set of tests, GAZD results showed a slight increase in the difference between the best minimal and maximal solutions per case, ranging from zero to eight zones. Nevertheless, GAZD found the global optimum in 60% of the optimizations (11 out of 18), and for the remaining cases, the deviation of GAZD's best minimal solution from the BILP results ranged from one to three zones, equivalent to a relative percentage error range of 4.3% to 15.4%. This demonstrates that GAZD is capable of finding global or near-optimal solutions, effectively avoiding premature convergence to local optima.

However, the BILP approach significantly outperformed GAZD in terms of execution time: BILP execution times ranged from 0.078 to 1.75 s, while GAZD's average times varied between 8.94 and 375.5 s. The execution times of the genetic algorithm increased with the combinatorial size of the problem (i.e., the number of samples in the instance) and with stricter homogeneity requirements. The execution time ratios (GAZD/BILP) varied widely, from 22.5 to 451 across all optimizations. While GAZD is slower, its execution times remain within acceptable limits for most agricultural decision-making processes, where real-time performance is not critical. Furthermore, strategies such as implementing the model in C and parallelizing processes during execution could significantly improve performance.

3.4. Scalability Analysis

Complementary experiments using a relative variance requirement of 0.9 were conducted with Instances 7–12 to evaluate the GAZD's performance on larger-scale optimization problems. These results, combined with those from Instances 1–6, provide a

comprehensive overview of GAZD's performance. Figures 9 and 10 summarize the results from 50 independent runs for each optimization case. The first figure presents the average execution time, while the second illustrates the trends in the minimum, average, and maximum of the best solutions across all instances. The execution time behavior exhibits two distinct phases. The first phase, covering Instances 1 to 6, shows an increasing trend as the instance size grows. This can be explained by the simultaneous increase in both the initial population size required to start the evolutionary process and the number of iterations needed to achieve solutions with fewer zones. The second phase, from Instances 7 to 12, marks a halt in this growth trend, with execution times stabilizing at values slightly lower than the maximum observed in Phase 1. This can be attributed to two compensating behaviors: while the initial population size requires decreases for this set of instances, the number of iterations increases. Overall, it can be inferred that beyond a certain instance size, execution time ceases to be a critical factor. In such cases, the improvement strategies proposed in previous sections could be sufficient to enhance GAZD's performance. In relation to the number of zones in the solutions, as shown in the second figure, the gap between the minimum and maximum number of zones in the best solutions widens as instance size increases. This indicates that for larger instances, GAZD's results exhibit increased variability in the solutions obtained. As a result, the algorithm's robustness and precision decrease in larger-scale problems, potentially leading to deviations from the optimal solution. This problem should be addressed in future research. Potential solutions could involve exploring how the maximum allowable partition sizes (i.e., the maximum number of zones) can be regulated, either statically or dynamically, within the evolutionary process. In the present study, these values were kept static, fixed to the maximum possible partitioning—equivalent to the number of sampled AUs. By varying these parameters, it may be possible to enhance the robustness and precision of the algorithm for larger-scale problems.

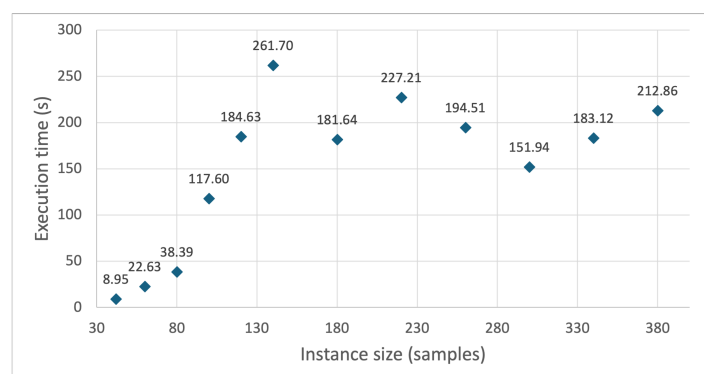


Figure 9. Execution time vs. instance size. Instances 1–12. RV = 0.9.

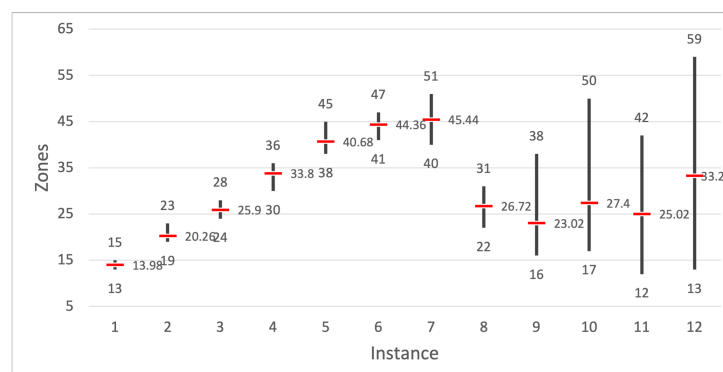


Figure 10. Number of zones vs. instance size. Instances 1–12. RV = 0.9. The red lines represent the average best solution, while the black lines indicate its range of variation across 50 executions.

3.5. Irregularly Shaped Instances

The results of GAZD for the three irregularly shaped instances are presented in Table 5. Three relative variance values (0.5, 0.7, and 0.9) were used to evaluate GAZD, resulting in a total of nine optimization cases. The first column lists the modified instance, while the second column indicates the number of sampled AUs. The third column specifies the required homogeneity level of the partition. Columns 4–7 present the results obtained across 50 independent runs of GAZD, along with the average execution time.

For this test, a performance comparison between the BILP and the GAZD was not conducted due to the number of dummy samples required by the BILP to handle the non-rectangular shapes. For the modified Instances 4 and 5, the BILP requires 33 dummy samples, and for modified Instance 6, 37 dummy samples are required. This extra information, added to the “real” field description, distorts the values of parameters such as the total variance and the relative variance of the partitions, which are used in the optimization process. In contrast, the GAZD uses only the information from the sampled AUs, preserving the “real” agricultural field description. Compared to the real-instance experiment, where only two dummy samples were involved, in this case the differences between the instances descriptions would be emphasized, resulting in the optimization of different problems by the algorithms. In these circumstances, the comparison between model was deemed not pertinent. Table 5 shows that the relationship between the homogeneity requirement and the number of zones in the partition remains consistent: as the relative variance requirement increases, the number of zones also increases. Figure 11 presents examples of solutions for all modified instances under various relative variance requirements. The figure illustrates that, in all cases, the GAZD is able to find solutions that respect the rectangular zone shape constraint.

Table 5. AG-SSMZ Results for Irregular-Shaped Generated Instances.

Modified Instance	Sampled AUs	Relative Variance	AG-SSMZ			
			Min.	Avg.	Max.	Time (s)
Instance 4	67	0.9	23	24.3	26	104.5069
		0.7	13	14.42	17	91.88751
		0.5	11	11.4	13	83.83991
Instance 5	87	0.9	27	29.88	32	182.679
		0.7	15	17.2	20	159.1199
		0.5	11	12.18	14	162.5808
Instance 6	103	0.9	35	37.24	39	233.6485
		0.7	18	20.86	23	214.5141
		0.5	12	13.12	16	347.3826

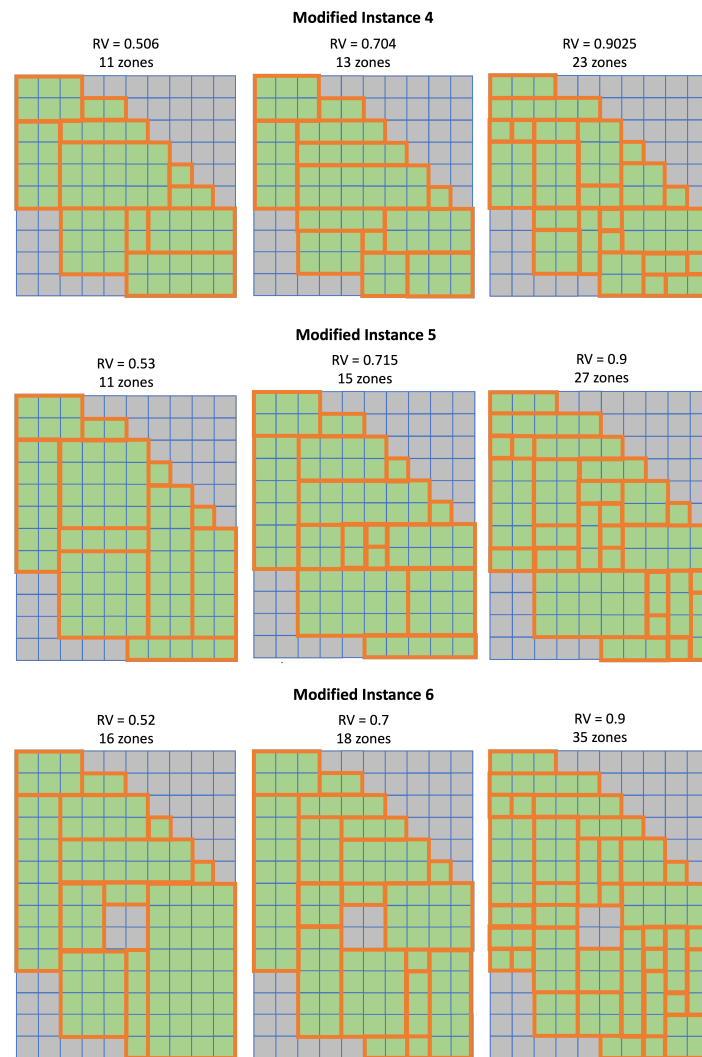


Figure 11. SSMZ delineation in irregularly shaped instances. Zone boundaries are shown in orange.

4. Conclusions

This paper presents a methodology based on genetic algorithms to address the site-specific management zone (SSMZ) delineation problem. The GAZD generates optimized field partitions, using rectangular zones, for both regular and irregularly shaped agricultural fields. While other methodologies add dummy samples to handle the SSMZ problem in non-rectangular fields, the GAZD uses only real sample data, thereby not altering the field description during the optimization process. The possibility to deal with irregularly shaped instances without adding non-real data is the major contribution of this methodology.

To evaluate the algorithm, its performance was compared with that of an exact approach based on integer linear programming (BILP). Experimental tests were conducted using real-field instances and a set of generated irregularly shaped instances of various sizes. Although the GAZD requires longer execution times compared to the exact BILP approach, the algorithm demonstrates functionality, flexibility, and robustness in addressing the SSMZ problem for both rectangular and irregularly shaped agricultural instances, especially for smaller problem sizes. However, for larger instance sizes, a loss of precision and robustness is observed, with the GAZD exhibiting increased variability in the solutions. This issue will be addressed in future work, where potential solutions may involve dynamically adjusting the maximum allowable partition sizes (i.e., the maximum number of zones).

during the evolutionary process to enhance the algorithm's robustness and precision for larger-scale problems.

Moreover, GAZD's time performance is not critical for the planning decisions related to the SSMZ problem, and there is significant potential for improving its performance. Strategies such as implementing the model in a compiled language and parallel processing could substantially reduce execution times. Additionally, the GAZD provides a set of "good enough" solutions that users can evaluate in terms of feasibility and practical convenience, offering a potential advantage in supporting decision-making processes.

Author Contributions: Conceptualization, F.H., L.M.P.-A. and V.M.A.; methodology, F.H. and L.M.P.-A.; software, F.H.; validation, F.H., L.M.P.-A., V.M.A. and M.P.; formal analysis, F.H.; investigation, F.H.; resources, V.M.A.; writing—original draft preparation, F.H.; writing—review and editing, F.H., V.M.A., L.M.P.-A. and M.P.; visualization, F.H.; supervision, L.M.P.-A., V.M.A. and M.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by CYTED program, grant number 524RT0158.

Data Availability Statement: Restrictions apply to the availability of these data. Data is available from the corresponding author.

Acknowledgments: The authors wish to acknowledge the support of the CYTED network Artificial Intelligence in Agriculture (AI4AGRIB).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Fraisse, C.; Sudduth, K.; Kitchen, N. Delineation of Site-Specific Management Zones by Unsupervised Classification of Topographic Attributes and Soil Electrical Conductivity. *Trans. Am. Soc. Agric. Eng.* **2001**, *44*, 155–166. [\[CrossRef\]](#)
2. Ortega, J.A.; Foster, W.; Ortega, R. Definition of sub-stands for Precision Forestry: An application of the fuzzy k-means method. *Cienc. E Investig. Agrar.* **2002**, *29*, 35–44. [\[CrossRef\]](#)
3. Johnson, C.K.; Mortensen, D.A.; Wienhold, B.J.; Shanahan, J.F.; Doran, J.W. Site-Specific Management Zones Based on Soil Electrical Conductivity in a Semiarid Cropping System. *Agron. J.* **2003**, *95*, 303–315. [\[CrossRef\]](#)
4. Whelan, B.; Cupitt, J.; McBratney, A.B.; Robert, P.C. Practical definition and interpretation of potential management zones in Australian dryland cropping. In Proceedings of the 6th International Conference on Precision Agriculture and Other Precision Resources Management, American Society of Agronomy, Minneapolis, MN, USA, 2003; pp. 395–409.
5. Franzen, D.; Nanna, T. Management zone delineation methods. In Proceedings of the 6th International Conference on Precision Agriculture and Other Precision Resources Management, American Society of Agronomy, Minneapolis, MN, USA, 2003; pp. 443–457.
6. Schepers, A.; Shanahan, J.; Liebig, M.; Schepers, J.; Johnson, S.H.; Luchiari, A., Jr. Appropriateness of Management Zones for Characterizing Spatial Variability of Soil Properties and Irrigated Corn Yields across Years. *Agron. J.* **2004**, *96*, 195–203. [\[CrossRef\]](#)
7. Diker, K.; Heermann, D.; Brodahl, M. Frequency Analysis of Yield for Delineating Yield Response Zones. *Precis. Agric.* **2004**, *5*, 435–444. [\[CrossRef\]](#)
8. Jaynes, D.; Colvin, T.; Kaspar, T. Identifying potential soybean management zones from multi-year yield data. *Comput. Electr. Agric.* **2005**, *46*, 309–327. [\[CrossRef\]](#)
9. Hornung, A.; Khosla, R.; Reich, R.; Inman, D.; Westfall, D.G. Comparison of Site-Specific Management Zones: Soil-Color-Based and Yield-Based. *Agron. J.* **2006**, *98*, 407–415.
10. Ortega, R.; Santibáñez, O. Determination of management zones in corn (*Zea mays* L.) based on soil fertility. *Comput. Electron. Agric.* **2007**, *58*, 49–59. [\[CrossRef\]](#)
11. Pedrosa Jr, M.; Taylor, J.; Tisseyre, B.; Charnomordic, B.; Guillaume, S. A segmentation algorithm for the delineation of agricultural management zones. *Comput. Electron. Agric.* **2010**, *70*, 199–208. [\[CrossRef\]](#)
12. Jiang, Q.; Fu, Q.; Wang, Z. Study on Delineation of Irrigation Management Zones Based on Management Zone Analyst Software. In Proceedings of the Computer and Computing Technologies in Agriculture IV, Nanchang, China, 22–25 October 2010; Li, D., Liu, Y., Chen, Y., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; pp. 419–427.
13. Cid-García, N.; Albornoz, V.; Rios, Y.; Ortega-Blu, R. Rectangular shape management zone delineation using integer linear programming. *Comput. Electron. Agric.* **2013**, *93*, 1–9. [\[CrossRef\]](#)

14. Albornoz, V.; Cid-Garcia, N.; Ortega-Blu, R.; Rios, Y. A Hierarchical Planning Scheme Based on Precision Agriculture. In *Handbook of Operations Research in Agriculture and the Agri-Food Industry*; Springer: New York, NY, USA, 2015; Volume 224, pp. 129–162. [[CrossRef](#)]
15. Albornoz, V.M.; Nanco, L.J.; Sáez, J.L. Delineating robust rectangular management zones based on column generation algorithm. *Comput. Electron. Agric.* **2019**, *161*, 194–201. [[CrossRef](#)]
16. Albornoz, V.; Zamora, G. Decomposition-based heuristic for the zoning and crop planning problem with adjacency constraints. *Top* **2021**, *29*, 248–265. [[CrossRef](#)]
17. Cid-Garcia, N.M.; Ibarra-Rojas, O.J. An integrated approach for the rectangular delineation of management zones and the crop planning problems. *Comput. Electron. Agric.* **2019**, *164*, 104925. [[CrossRef](#)]
18. Albornoz, V.; Véliz, M.; Ortega-Blu, R.; Ortiz-Araya, V. Integrated versus hierarchical approach for zone delineation and crop planning under uncertainty. *Ann. Oper. Res.* **2020**, *286*, 617–634. [[CrossRef](#)]
19. Rivero, L.; Velasco, J.; Ramirez, J. A Simple Greedy Heuristic for Site Specific Management Zone Problem. *Axioms* **2022**, *11*, 318. [[CrossRef](#)]
20. Velasco, J.; Vicencio, S.; Lozano, J.A.; Cid-Garcia, N.M. Delineation of site-specific management zones using estimation of distribution algorithms. *Int. Trans. Oper. Res.* **2023**, *30*, 1703–1729.
21. Vicencio-Medina, S.J.; Rios-Solis, Y.A.; Cid-Garcia, N.M. A bio-inspired optimization algorithm with disjoint sets to delineate orthogonal site-specific management zones. *Precis. Agric.* **2024**, *26*, 14. [[CrossRef](#)]
22. Holland, J. *Adaptation in Natural and Artificial Systems*; University of Michigan Press: Ann Arbor, MI, USA, 1975.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.