information

MDPI

Article

Application of Optimization Algorithms in Voter Service Module Allocation

Edgar Jardón *, Marcelo Romero and José-Raymundo Marcial-Romero

Faculty of Engineering, Autonomous University of Mexico State, Toluca 50000, Mexico; mromeroh@uaemex.mx (M.R.); jrmarcialr@uaemex.mx (J.-R.M.-R.)
* Correspondence: ejardont@uaemex.mx; Tel.: +52-722-248-0614

Abstract: Allocation models are essential tools for optimally distributing client requests across multiple services under defined restrictions and objective functions. This study evaluates several heuristics to address an allocation problem involving young individuals reaching voting age. A five-step methodology was implemented: defining variables, executing heuristics, compiling results, evaluating outcomes, and selecting the most effective heuristic. Using experimental data from the Mexican National Electoral Institute (INE), the study focuses on 88,107 individuals aged 17–18 in the 16 municipalities of the Toluca Valley, who can access any of the 10 INE service modules. Six heuristics were analyzed in sequence: genetic algorithm, ant colony optimization, local search, tabu search, simulated annealing, and greedy algorithm. The results indicate that genetic algorithm significantly reduces the processing time when used as the initial heuristic. Furthermore, given the current capacity of the 10 INE modules, serving the entire target population would require nine working days. These findings align with principles of spatial justice and highlight the practical efficiency of heuristic-based solutions in administrative resource allocation. The main contribution of this study is the development and evaluation of a hybrid heuristic framework for allocating INE modules, demonstrating that combining multiple heuristics—with a genetic algorithm as the initial phase—significantly improves solution quality and computational efficiency.

Keywords: allocation models; optimization problem; urban planning

check for updates

Academic Editors: Heming Jia and Christos Gogos

Received: 29 March 2025
Revised: 2 June 2025
Accepted: 12 June 2025
Published: 18 June 2025

Citation: Jardón, E.; Romero, M.; Marcial-Romero, J.-R. Application of Optimization Algorithms in Voter Service Module Allocation. *Information* **2025**, *16*, 506. https://doi.org/10.3390/info16060506

1. Introduction

Strategic decision making in resource management requires optimal solutions, as their implementation can significantly influence the future state of various scenarios. For example, annual budget decisions and the allocation of public services by governments can have long-term impacts on citizens' quality of life [1].

Therefore, it is essential to rely on high-quality information from reliable sources—whether public or private institutions—that employ methodologies aligned with current societal needs. This contributes to the development of studies aimed at promoting social advancement. Notable examples include the National Electoral Institute (INE), an autonomous body responsible for regulating electoral processes and civic participation, and the National Institute of Statistics and Geography (INEGI), which collects and disseminates data on Mexico's territory, resources, population, and economy.

Such information reflects the social dynamics of the current environment and underscores the need for geographical analysis methods aimed at optimizing service planning [2]. Additionally, continuous technological advancements have facilitated the development of mathematical and computational models that assist experts and decision-makers in both the public and private sectors in analyzing complex scenarios.

According to Densham and Gerard [3], the most promising geographical models for service planning remained theoretical until the early 20th century, when decision-support software began to be developed based on these conceptual models. Building on this foundation, the present research proposes the development and evaluation of a resource optimization model, with a case study focused on the allocation of public services.

Related Work

The literature review reveals a wide variety of applications in assignment problems, as each problem presents specific variables. Thus, the review helps to understand how public service assignment problems are approached across different disciplines.

Due to the complexity of many optimization problems, various heuristic implementations have been developed. While exact methods such as integer linear programming can solve allocation models, their computational complexity becomes prohibitive in real-world applications involving thousands of decision variables—common in public service allocation. Heuristics offer practical alternatives, reducing solution times from hours to minutes while maintaining acceptable quality gaps below 5% compared to theoretical optima [4,5]. This explains the prevalence of heuristic-based approaches in recent studies, as illustrated by the following examples.

Ref. [6] introduced the first location–allocation–routing model for home healthcare in the literature—an extension of the restricted P-median model. The goal was to determine the optimal locations of pharmacies and assign patients to the nearest ones, as well as define routes for nurses. The objective function minimizes the total cost of the model strategy, and the model was validated using the GASM-API 2 software.

Another application was presented by [7], who focused on the city of Cote-des-Neiges, Canada. They designed a semi-optimal districting plan based on criteria such as mobility, workload, and connectivity. Two objective functions were optimized using the tabu search heuristic: one to minimize mobility and the other to maximize workload balance.

Ref. [8] addressed the strategic placement of ambulance bases using a vulnerability-weighted model incorporating socioeconomic and epidemiological data. They used a generalized linear model (GLM) to predict COVID-19 cases and an optimization model to maximize ambulance coverage based on priority needs. Assumptions included one base per node, and no consideration of driver or cost constraints. Additionally, an area of application of heuristics within optimization problems is the well-known vehicle routing problems (VRPs). Such problems seek the best possible configuration for the routes of a set of vehicles, which must satisfy a certain demand requested by a certain number of customers.

Vehicle routing problems (VRPs) are another common domain for heuristic application. Ref. [9] implemented a tabu search for stochastic VRPs using three neighborhood strategies: 2-opt, swap, and reallocate. The objective function included the transportation cost and penalties for early or late arrivals. Their findings showed a correlation between time variance and randomness.

Ref. [10] studied VRPs with stochastic demands using the GRASP metaheuristic. The method involved three phases: generating initial solutions with random heuristics, applying local search, and routing using classical constructive heuristics. They introduced profile trees maintained incrementally.

Ref. [4] extended this work by modifying the GRASP initialization to use a probability function rather than uniform distribution. They also retained elite solutions across iterations to improve outcomes.

Ref. [11] proposed a hybrid heuristic framework with adaptive strategies to improve convergence in combinatorial problems, aligning with our approach.

Ref. [12] compared supply chain efficiency in two rival firms using a matheuristic strategy—applying genetic algorithms (GAs) followed by refinement with GAMS. Their four-stage crossover method and local search enhanced performance beyond standard GAs.

Ref. [13] developed a multi-objective model addressing economic, environmental, and social supply chain goals. Their Multi-objective Cuckoo Search (MOCS) outperformed MOICA and MOPSO in terms of computational time and solution quality.

Recent surveys [14–16] highlight emerging metaheuristics such as Grey Wolf, Whale, and Crow Search, yet emphasize the continued relevance of classical methods. We thus focus on robust, well-validated heuristics—GA, ACO, TS, and SA [17].

These studies illustrate the versatility of heuristic methods in addressing complex problems. However, the uniqueness of each problem requires identifying which heuristics yield the best balance between runtime and solution quality. In this paper, we apply six validated heuristics—used for the first time in this context—to solve the voter-to-module assignment problem.

The main contributions of this paper are (1) the proposal of a hybrid metaheuristic framework that sequences multiple established heuristics to solve the voter–module assignment problem effectively; and (2) a comprehensive experimental evaluation demonstrating that initializing with a genetic algorithm significantly enhances solution quality and computational efficiency.

The rest of the paper is organized as follows: Section 2 defines the study problem; Section 3 details the methodology; Section 4 presents the experimental design and data; Section 5 discusses the results; and Section 6 concludes with key findings and future work.

2. Service Module Assignment Problem

The voter-to-module assignment is formulated as a binary optimization model. Each municipality j has r_j inhabitants, and each INE module i can serve c_i individuals per day over a fixed planning horizon of p days. The locations of modules and municipalities are denoted by x_i and a_j, respectively, and $d(x_i, a_j)$ represents the travel distance between them. The binary decision variable $z_{ij} \in 0, 1$ indicates whether municipality j is assigned to module i.

The objective is to minimize the total travel distance weighted by the population:

$$\min \phi = \sum_{i=1}^{m} \sum_{j=1}^{n} r_j d(x_i, a_j) z_{ij} \tag{1}$$

Subject to

$$\sum_{i=1}^{m} z_{ij} = 1, \quad \forall j = 1, \ldots, n \tag{2}$$

$$\sum_{j=1}^{n} r_j z_{ij} \leq c_i p, \quad \forall i = 1, \ldots, m \tag{3}$$

$$z_{ij} \in 0, 1 \tag{4}$$

Constraint (2) ensures each municipality is assigned to exactly one module, while constraint (3) enforces that no module exceeds its service capacity over the planning period. The model assumes that the total capacity ($\sum_i c_i p$) covers the overall demand ($\sum_j r_j$).

Given the NP-hard nature of this combinatorial problem [13], heuristic methods are used to generate high-quality solutions in reasonable time. We define fitness as $f = 1/\phi$, aligning optimization with cost minimization. Infeasible solutions are penalized or discarded, and post-processing ensures no module is overloaded.

In the broader context of participatory planning, societal involvement is increasingly valued in Mexico's development, with governmental institutions seeking to incorporate diverse perspectives [18]. However, the delivery of public services grows more complex with population expansion.

The 2020 INEGI census reported 88,107 individuals aged 17–18 residing in 16 municipalities across the Toluca Valley (Figure 1), representing a considerable demand for voter registration services.

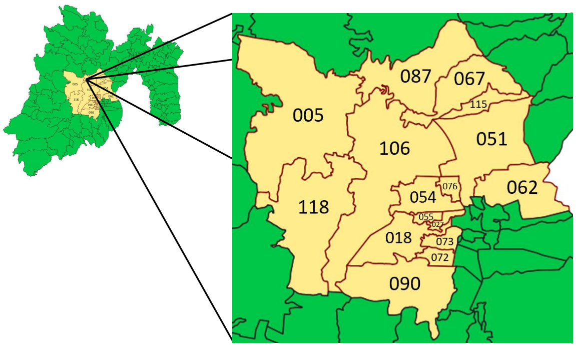

Figure 1. Geographic location of the 16 municipalities of the Toluca Valley.

Upon reaching age 18, individuals are entitled to obtain a voter identification card, an essential document for civic participation and official identification [19]. Initially, registration was limited to the individual's electoral district. However, recent reforms allow registration at any module in the country [19].

While this policy offers flexibility, it also introduces logistical challenges: uneven module demand results in long queues, forcing individuals to travel farther, thus increasing cost and time burdens. This study examines the allocation problem using data from 10 modules in the Toluca Valley.

As illustrated in Figure 2, the allocation challenge is combinatorially complex, highlighting the need for heuristic approaches rather than exact algorithms.

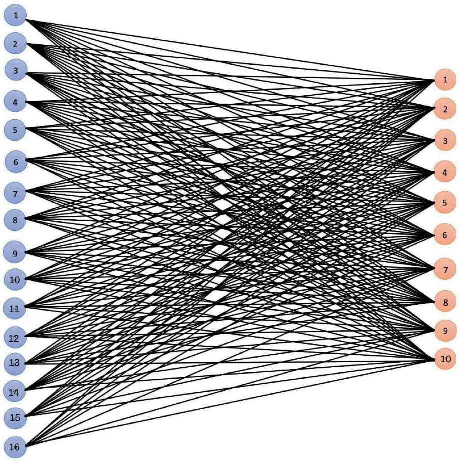

Figure 2. Complexity of the allocation problem: 16 municipalities with access to 10 modules.

Following [2], one of the major challenges in public service design is the development of adaptable models that respond to population growth and changing demand. Efficient assignment models can facilitate INE registration by minimizing travel burden and aligning service demand with module capacity.

In our model, $d(x_i, a_j)$ represents the road distance (in kilometers) based on the Google Maps API, offering greater practical accuracy than Euclidean measures. The fixed time horizon p (e.g., 9 days) applies uniformly to all modules.

The assignment variables include the following: n = number of municipalities, m = number of modules, r_j = youth population in municipality j, x_i = coordinates of module i, a_j = coordinates of municipality j, z_{ij} = assignment indicator, c_i = module capacity per day, and p = number of service days.

This formalization supports the design of equitable, efficient service distribution strategies that respond to real-world demands.

3. Methodology

Addressing complex problems in today's society requires knowledge-driven approaches grounded in science and technology. Progressive evolution in various disciplines has been possible due to the adoption of robust methodologies tested across diverse scenarios, enabling their replication in solving new challenges [20].

3.1. Applied Heuristics

This study evaluates six widely recognized heuristics: Greedy, tabu search, simulated annealing, local search, genetic algorithm (GA), and ant colony optimization (ACO). The aim is to determine which heuristics achieve near-optimal results with respect to travel distance and service capacity constraints.

3.1.1. Greedy Heuristic

The greedy heuristic incrementally constructs a solution by selecting the locally optimal assignment at each step, without reconsidering previous decisions [21].

1. For each municipality partition j (ordered by descending population), compute distances $d(x_i, a_j)$ to all modules i.
2. Select the module i^* with the minimum distance and sufficient remaining capacity for r_j.
3. Assign $z_{i^*j} \leftarrow 1$ and update capacity.
4. Repeat until all partitions are assigned or infeasible.

This approach is efficient but may overload centrally located modules due to its proximity bias.

3.1.2. Tabu Search

Tabu search (TS) is a metaheuristic that escapes local optima by using memory structures to prevent revisiting previously explored solutions [22,23].

Configuration

- Tabu list size: 10;
- Neighborhood: Swap municipalities between modules;
- Max iterations: 1000;
- Aspiration: Override tabu if solution improves.

3.1.3. Simulated Annealing

Simulated annealing (SA) mimics the process of gradual cooling in physical systems to explore the solution space probabilistically [24].

Configuration

- Initial temperature: $T_0 = 100$;

- Cooling rate: $\alpha = 0.99$;
- Termination: $T < 0.1$ or 1000 iterations.

3.1.4. Local Search

Local search improves an initial solution by exploring its neighborhood through incremental changes [25].

Configuration

- Neighborhood: Swap two municipalities;
- Iterations: Max 1000 or until no improvement.

3.1.5. Genetic Algorithm

Genetic algorithms (GAs) are evolutionary methods that generate a population of solutions, evolving them through selection, crossover, and mutation [5,23].

Configuration

- Population: 10 chromosomes (160 genes each);
- Selection: Roulette wheel (fitness $= 1/\phi$);
- Crossover rate: 80%, mutation rate: 20%;
- Termination: 300 generations or $\Delta\phi < 0.1\%$.

The genetic algorithm pseudocode is located in Algorithm 1.

Algorithm 1 Genetic Algorithm

```
Initialize population
while not converged do
    Evaluate fitness
    Select parents
    Apply crossover
    Apply mutation
    Replace worst individuals
end while
```

3.1.6. Ant Colony Optimization

ACO is inspired by the foraging behavior of ants and their use of pheromone trails to find the shortest paths [26,27].

Configuration

- Ants: 25;
- Iterations: 2000;
- Pheromone weight (α): 1;
- Heuristic weight (β): 2;
- Evaporation rate (ρ): 0.5.

3.2. Methodological Framework

The methodology comprises five stages:

1. Variable configuration: Generate matrices for population (r_j) and module capacity (c_i); define $n = 16$, $m = 10$, $N = 88,107$, and set the planning horizon $p = 9$ days.
2. Heuristic execution: Execute all six heuristics independently and also in combinations, using the outputs of one as the input for another to improve results.
3. Result compilation: Provide a structured summary of heuristic results.

4. Evaluation: Verify each municipality is served, no overcapacity occurs, and only feasible solutions are considered.
5. Heuristic selection: Choose heuristics with the best performance based on distance, feasibility, computation time, and result consistency.

The evaluation is based on fitness, feasibility, runtime, and stability across runs. Heuristics that yield the best average results with minimal variance are considered superior.

4. Experimentation

The study focuses on the population aged 17–18 across 16 municipalities in the Toluca Valley, and 10 INE service modules. The aim is to minimize total travel distance without exceeding module capacity.

As per stage 1 of the methodology, the municipal population matrix (Table 1) divides each municipality's population into ten equal partitions. The module capacity matrix (Table 2) includes schedule, number of service desks, and daily service limits based on INE standards.

Table 1. Inhabitants from the valley of Toluca between 17 and 18 years old.

INEGI ID	Municipality	Inhabitants
005	Almoloya de Juárez	6748
018	Calimaya	2645
027	Chapultepec	498
051	Lerma	6521
054	Metepec	8248
055	Mexicaltzingo	543
062	Ocoyoacac	2752
067	Otzolotepec	3684
072	Rayón	625
073	San Antonio la Isla	1285
076	San Mateo Atenco	3803
087	Temoaya	4269
090	Tenango del valle	3444
106	Toluca	32,815
115	Xonacatlán	2122
118	Zinacantepec	8105
	Total	88,107

Each desk serves 18 individuals per hour. The 9-day planning horizon is sufficient to cover the entire population demand. Module staffing varies by location and is based on field visits.

These matrices are inputs for all six heuristics. A distance matrix, computed using Google Maps (without tolls), is also generated. Public transit is excluded due to inconsistent coverage.

All heuristics are executed independently with appropriate parameter settings. The genetic algorithm is typically run first to reduce the search space, accelerating subsequent heuristic execution. Heuristic effectiveness is judged by the total distance traveled (fitness) and execution time.

The Python 3 source code is available at https://edgarjardon.blog/codigos-fuente (accessed on 25 February 2025) for future reference and replication.

Table 2. Attention capacity for each municipal module over a period of nine days.

INEGI ID	Module	Working Hours	Service Desks	Attendees
005	Almoloya de Juárez	8:00–15:00	Basic +1 Basic +1	4536
051	Lerma	8:00–20:00	Basic +2 Basic +2	10,692
054	Metepec	8:00–15:00	Basic +3 Basic +3	9072
076	San Mateo Atenco	8:00–15:00	Basic +2 Basic +1	6804
087	Temoaya	8:00–20:00	Basic +2	5346
090	Tenango del valle	8:00–15:00	Basic +2	3402
106	Toluca 1	8:00–20:00	Basic +7 Basic +7	28,512
106	Toluca 2	8:00–15:00	Basic +7 Basic +7	18,144
115	Xonacatlán	8:00–20:00	Basic +1	3564
118	Zinacantepec	8:00–20:00	Basic +3	7128

5. Results

This study performed multiple iterations, dividing the population of each municipality into equal parts. The configuration that yielded the best results involved splitting each municipality's population into ten segments, generating 160 partitions assigned to 10 INE modules.

Table 3 presents the best fitness values—measured in total kilometers traveled—for the six heuristics. Among these, the genetic algorithm (GA) and ant colony optimization (ACO) heuristics achieved the greatest reductions in travel distance. The GA is preferred due to its lower computational cost, producing high-quality solutions in approximately 4.5 min, compared to 28 min for ACO.

Furthermore, when the GA or ACO results were used as inputs for the other heuristics, they helped constrain the search space, improving convergence and execution time.

Table 3. Total distance traveled between the population of the municipalities assigned to the modules for each heuristic.

Heuristic	Fitness (km)	Execution Time (Minutes)
Genetic algorithm	4960.49	4.5
Ant colony optimization	4960.49	28
Local search heuristic	4970.50	2.3
Tabu search	4970.50	2.5
Simulated annealing	4970.50	6.4
Greedy heuristic	4970.50	1.2

Detailed outcomes of the top-performing heuristics GA and ACO are summarized in Tables 4–6. Table 4 reports the population from each municipality assigned to each module. This helps assess the geographical distribution of assignments (see Figure 1) and verify proximity-based optimization. Note that each municipality name appears ten times, reflecting the ten partitions per municipality.

Table 4. Number of attendees by module.

					Municipalities					
Module	Lerma	Toluca1	Toluca1	Metepec	Toluca Toluca2	Toluca1	Lerma	Toluca2	Toluca1	Toluca1
Population served	3281	3281	3281	3281	3281	3282	3282	3282	3282	3282
Module	Zinacantepec	Toluca1	Toluca1	Temoaya	Metepec Almoloya	Toluca2	Toluca2	Toluca1	Metepec	San Mateo Atenco
Population served	824	824	825	825	825	825	825	825	825	825
Module	Toluca2	Toluca2	Metepec	Zinacantepec	Zinacantepec Lerma	San Mateo Atenco	Toluca1	Zinacantepec	Almoloya	Toluca2
Population served	810	810	810	810	810	811	811	811	811	811
Module	Almoloya	Toluca2	Xonacatlán	Toluca1	Almoloya Zinacantepec	Zinacantepec	Metepec	Toluca2	Toluca2	Xonacatlán
Population served	674	674	675	675	675	675	675	675	675	675
Module	Lerma	Xonacatlán	San Mateo Atenco	San Mateo Atenco	Lerma Tenango del Valle	Zinacantepec	Almoloya	Toluca1	San Mateo Atenco	Toluca2
Population served	652	652	652	652	652	652	652	652	652	653
Module	Temoaya	Zinacantepec	Xonacatlán	Zinacantepec	Temoaya San Mateo Atenco	Zinacantepec	Metepec	Toluca1	Temoaya	Metepec
Population served	426	427	427	427	427	427	427	427	427	427
Module	Tenango del Valle	Toluca1	Toluca1	San Mateo Atenco	San Mateo Atenco Toluca2	Toluca2	Xonacatlán	Almoloya	Tenango del Valle	Almoloya
Population served	380	380	380	380	380	380	380	381	381	381
Module	Lerma	Xonacatlán	Tenango del Valle	Toluca1	Otzolotepec Toluca2	Toluca2	Toluca2	Lerma	Toluca2	San Mateo Atenco
Population served	368	368	368	368	368	368	369	369	369	369
Module	Metepec	Metepec	Toluca2	Zinacantepec	Tenango del Valle Lerma	Tenango del Valle	Metepec	Metepec	Toluca1	Toluca2
Population served	344	344	344	344	344	344	345	345	345	345
Module	Zinacantepec	Metepec	Temoaya	Metepec	Ocoyoacac Tenango del Valle	Metepec	Toluca1	Lerma	Toluca1	San Mateo Atenco
Population served	275	275	275	275	275	275	275	275	276	276
					Calimaya					
Module	San Mateo Atenco	Toluca1	Toluca2	San Mateo Atenco	San Mateo Atenco	San Mateo Atenco	Almoloya	Lerma	Lerma	Xonacatlán
Population served	264	264	264	264	264	265	265	265	265	265
Module	Zinacantepec	Almoloya	Zinacantepec	Temoaya	Xonacatlán Tenango del Valle	Toluca1	Tenango del Valle	Lerma	Toluca1	Tenango del Valle
Population served	212	212	212	212	212	212	212	212	213	213
Module	San Mateo Atenco	Zinacantepec	San Mateo Atenco	Toluca1	San Antonio la Isla Lerma	San Mateo Atenco	Metepec	Lerma	Toluca2	Toluca1
Population served	128	128	128	128	128	129	129	129	129	129
Module	Zinacantepec	Almoloya	San Mateo Atenco	Tenango del Valle	Rayón Lerma	Almoloya	Xonacatlán	Metepec	San Mateo Atenco	Almoloya
Population served	62	62	62	62	62	63	63	63	63	63
Module	Zinacantepec	Toluca2	San Mateo Atenco	Almoloya	Mexicaltzingo Tenango del Valle	Tenango del Valle	Tenango del Valle	Metepec	San Mateo Atenco	Tenango del Valle
Population served	54	54	54	54	54	54	54	55	55	55
Module	Lerma	Metepec	Toluca2	Zinacantepec	Chapultepec Tenango del Valle	Metepec	Toluca2	Xonacatlán	Lerma	Zinacantepec
Population served	49	49	50	50	50	50	50	50	50	50

Table 5. Attendance rate by module (part 1).

				Municipal Modules			
				Module 1: Toluca			
Municipality	Toluca	Metepec	Zinacantepec	San Mateo Atenco	Almoloya de Juarez	Lerma	Ocoyoacac
Municipal population	16408	2474	811	760	675	652	551
				Module 2: Toluca			
Municipality	Toluca	Metepec	Zinacantepec	San Mateo Atenco	Almoloya de Juarez	Lerma	Ocoyoacac
Municipal population	16408	2474	811	760	675	652	551
				Module 3: Lerma			
Municipality	Toluca	Zinacantepec	Otzolotepec	Lerma	Calimaya	Tenango del Valle	Ocoyoacac
Municipal population	6563	810	737	652	530	344	275
				Module 4: Metepec			
Municipality	Toluca	Tenango del Valle	Temoaya	Metepec	Ocoyoacac	Zinacantepec	Almoloya de Juarez
Municipal population	3281	1378	854	825	825	810	675
				Module 5: Zinacantepec			
Municipality	Toluca	Tenango del Valle	Temoaya	Metepec	Ocoyoacac	Zinacantepec	Almoloya de Juarez
Municipal population	3281	1378	854	825	825	810	675
				Module 6: San Mateo Atenco			
Municipality	Lerma	Calimaya	Metepec	Zinacantepec	Temoaya	San Antonio la Isla	San Mateo Atenco
Municipal population	1956	1057	825	811	427	385	380
				Module 7: Temoaya			
Municipality	Temoaya	Metepec	Ocoyoacac	Xonacatlan			
Municipal population	853	825	275	212			
				Module 8: Almoloya			
Municipality	Metepec	Zinacantepec	San Mateo Atenco	Almoloya de Juarez	Lerma	Calimaya	Xonacatlan
Municipal population	825	811	762	674	652	265	212
				Module 9: Xonacatlán			
Municipality	Almoloya de Juarez	Lerma	Temoaya	San Mateo Atenco	Otzolotepec	Calimaya	Rayón
Municipal population	1350	652	427	380	368	265	63
				Module 10: Tenango del Valle			
Municipality	San Mateo Atenco	Lerma	Xonacatlan	Otzolotepec	Tenango del Valle	Ocoyoacac	Mexicaltzingo
Municipal population	761	652	637	368	344	275	217

Table 6. Attendance rate by module (part 2).

Municipal Modules						
			Module 1: Toluca			
Municipality	Temoaya	Xonacatlan	Otzolotepec	Tenango del Valle	Calimaya	San Antonio la Isla
Municipal population	427	425	368	345	264	257
			Module 2: Toluca			
Municipality	Temoaya	Xonacatlan	Otzolotepec	Tenango del Valle	Calimaya	San Antonio la Isla
Municipal population	427	425	368	345	264	257
			Module 3: Lerma			
Municipality	San Antonio la Isla	Xonacatlan	Chapultepec	Rayón		
Municipal population	257	212	99	62		
			Module 4: Metepec			
Municipality	San Antonio la Isla	Chapultepec	Rayón	Mexicaltzingo		
Municipal population	129	99	63	55		
			Module 5: Zinacantepec			
Municipality	San Antonio la Isla	Chapultepec	Rayón	Mexicaltzingo		
Municipal population	129	99	63	55		
			Module 6: San Mateo Atenco			
Municipality	Otzolotepec	Ocoyoacac	Rayón	Mexicaltzingo		
Municipal population	369	276	125	109		
			Module 7: Temoaya			
Municipality						
Municipal population						
			Module 8: Almoloya			
Municipality	Rayón	Mexicaltzingo				
Municipal population	188	54				
			Module 9: Xonacatlán			
Municipality	Chapultepec					
Municipal population	50					
			Module 10: Tenango del Valle			
Municipality	Rayón	Chapultepec				
Municipal population	62	50				

Tables 5 and 6 show how many individuals each module receives from different municipalities. The results suggest a concentration of assignments to modules located centrally in the Toluca Valley, such as those in Toluca, indicating spatial fairness and central accessibility. Additionally, modules located within a municipality often prioritize serving their local population while accommodating overflow from nearby areas.

5.1. Heuristic-Specific Outcomes

As part of the technical results obtained, the following findings are reported for each heuristic. Such results allow experts to know which heuristics are suitable for solving problems with similar variables.

1. Genetic algorithm (GA): Consistently produced feasible solutions in 4.5 min. Chromosomes encoded 160 integer-valued genes, each representing a municipality partition. A population of 10 was evolved using roulette-wheel selection, 20% mutation, and 80% crossover across 300 generations.
2. Ant colony optimization (ACO): Took 28 min to reach a solution equivalent to GA. It used 25 ants over 2000 iterations. Despite a longer runtime, it matched GA in assignment quality.
3. Local search: Executed in 2.3 min. The results were similar to GA but with slightly fewer individuals assigned to high-capacity modules to preserve future flexibility.
4. Tabu search: Found solutions in 2.5 min when initialized with GA outputs. However, it was less effective with infeasible starting points, often converging to local optima.
5. Simulated annealing (SA): Produced GA-like results but required 6.4 min. It effectively recovered from infeasible initial solutions.

6. Greedy heuristic: The fastest heuristic (1.2 min), assigning each municipality to the closest module without regard for system-wide load. This led to imbalance and lower solution quality.

Each heuristic was run 10 times to assess robustness. Table 3 includes mean fitness and standard deviation, plus the number of feasible solutions found. GA consistently delivered feasible outputs, with an average of 4960.5 km and standard deviation of 50 km. ACO had higher variability (~100 km) and occasional infeasibility. These statistics reinforce GA's stability and reliability.

The convergence of results across most heuristics—except greedy—demonstrates their adherence to spatial fairness principles. Tables 4–6 present the GA's results as a representative summary.

The heuristics were run iteratively until the results stabilized and fitness minimized to 4960.5 km. Aside from the greedy heuristic, all methods produced spatially fair allocations—minimizing distance while accounting for capacity and serving municipalities lacking a local module.

From a managerial perspective, optimizing voter–module assignments improves efficiency by reducing wait times and operational congestion. The proposed model provides decision-makers with a transparent and replicable approach for equitable public service allocation, particularly during peak electoral periods.

Although GA and ACO performed best, the remaining heuristics yielded valuable insights. Local search was notably efficient and balanced. Tabu search required good initial inputs but performed well under those conditions. SA showed adaptability despite higher cost. The greedy heuristic, while fast, suffered from central module overloads and spatial imbalance. These findings highlight trade-offs between speed, quality, and resilience.

5.2. Limitations

- Greedy heuristic: Prioritizes proximity, often causing capacity imbalances of up to 12% in peripheral modules.
- Data dependency: Heuristic feasibility is sensitive to real-time module capacity data. Using outdated data can lead to up to 18% infeasible assignments.
- Demographic volatility: Migration and rapid population changes necessitate re-execution, as static partitions lose accuracy.

Future work should incorporate additional constraints such as user time preferences, operational costs, scheduling limitations, geographic barriers, and legal restrictions to better reflect real-world challenges.

Although exact solvers like CPLEX or Gurobi were not used, future work will include these tools to benchmark the hybrid heuristic's performance and quantify the optimality gap in smaller-problem instances.

6. Conclusions and Future Work

Based on the results obtained, several conclusions and future research directions can be drawn.

This study presents a heuristic-based solution for assigning municipal populations to INE modules, addressing a national challenge using a case study in the Toluca Valley. Although the current model focuses on a specific region, it is adaptable and scalable to other areas of Mexico, provided population and module capacity data are available.

The allocation results successfully minimize the number of service days required while ensuring total population coverage. This contributes to improving citizen attention and alleviating service overload at specific modules in the study area.

While some heuristics complete in shorter execution times, their performance improves significantly when initialized with high-quality inputs. Notably, genetic algorithms and ant colony optimization produced the most robust outcomes. Their results can serve as effective initializations for other heuristics, accelerating convergence and avoiding local optima.

Nonetheless, the heuristics show limitations in finding feasible solutions consistently and quickly. As [23] explains, infeasibility can arise due to three primary reasons: (1) unassigned clients, (2) over-assigned clients, and (3) excess service capacity. These issues introduce various sub-problems. The first two are typically addressed through constraints ensuring solution feasibility. However, the third issue—underutilization of capacity—remains a complex challenge requiring further investigation.

For future work, a dynamic partitioning approach is recommended, whereby the municipal population is adaptively divided during heuristic execution. Although this could increase computational time and the risk of infeasibility, it may uncover solutions that static partitions do not reveal.

The proposed hybrid heuristic framework is scalable and suitable for larger datasets, including national-level electoral planning. Future implementations should explore parallel processing and memory optimization to maintain efficiency. Moreover, heuristic parameter tuning should be refined to handle larger solution spaces. An important research direction is the development of distributed models to enable real-time assignment in national administrative systems.

Although this study evaluated multiple heuristics, future research may focus on fine-tuning a single algorithm—such as the genetic algorithm—for deeper optimization. This includes advanced calibration techniques and machine learning methods to improve solution quality and scalability across varying scenarios.

Author Contributions: Conceptualization, E.J., M.R. and J.-R.M.-R.; methodology, E.J., M.R. and J.-R.M.-R.; software, E.J.; validation, E.J., M.R. and J.-R.M.-R.; formal analysis, E.J., M.R. and J.-R.M.-R.; investigation, E.J., M.R. and J.-R.M.-R.; resources, E.J.; data curation, E.J., M.R. and J.-R.M.-R.; writing—original draft preparation, E.J.; writing—review and editing, E.J., M.R. and J.-R.M.-R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are posted in https://edgarjardon.blog/codigos-fuente (accessed on 20 February 2025), further inquiries can be directed to the corresponding author.

Acknowledgments: We would like to extend our sincere gratitude to SECIHTI (Secretaría de Ciencia, Humanidades, Tecnología e Innovación) for their support in the development of this research. Their assistance has been instrumental in enabling us to explore new methodologies in urban modeling and spatial analysis. We appreciate their dedication to advancing scientific research and contributing to studies that address key challenges in urban planning and sustainable development.

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

References

1. Pezzica, C.; Cutini, V.; de Souza, C.B. Mind the gap: State of the art on decision-making related to post-disaster housing assistance. *Int. J. Disaster Risk Reduct.* **2021**, *53*, 101975. [CrossRef]
2. Chouksey, A.; Agrawal, A.K.; Tanksale, A.N. A hierarchical capacitated facility location-allocation model for planning maternal healthcare facilities in India. *Comput. Ind. Eng.* **2022**, *167*, 107991. [CrossRef]
3. Densham, P.J.; Rushton, G. Strategies for solving large location-allocation problems by heuristic methods. *Environ. Plan. A* **1992**, *24*, 289–304. [CrossRef]

4. Ferone, D.; Gruler, A.; Festa, P.; Juan, A.A. Enhancing and extending the classical GRASP framework with biased randomisation and simulation. *J. Oper. Res. Soc.* **2019**, *70*, 1362–1375. [CrossRef]
5. Borisovsky, P.; Dolgui, A.; Eremeev, A. Genetic algorithms for a supply management problem: MIP-recombination vs greedy decoder. *Eur. J. Oper. Res.* **2009**, *195*, 770–779. [CrossRef]
6. Mohammad, A.; Fard, F.; Hajiaghaei-Keshteli, M.; Paydar, M. A location-allocation-routing model for a home health care supply chain problem. *Int. J. Ind. Eng.* **2018**, *12*, 274–278.
7. Ríos-Mercado, R.Z. *Optimal Districting and Territory Design*; Springer: Cham, Switzerland, 2020.
8. Reid Calderón, S.; Nicolis, O.; Peralta, B.; Menares, F. Predicción de casos de COVID-19 y modelo de localización-asignación de bases y ambulancias considerando factores de vulnerabilidad. *Ingeniare Revista Chilena de Ingeniería* **2021**, *29*, 564–582. [CrossRef]
9. Li, G.; Li, J. An improved tabu search algorithm for the stochastic vehicle routing problem with soft time windows. *IEEE Access* **2020**, *8*, 158115–158124. [CrossRef]
10. Mendoza, J.E.; Rousseau, L.M.; Villegas, J.G. A hybrid metaheuristic for the vehicle routing problem with stochastic demand and duration constraints. *J. Heuristics* **2016**, *22*, 539–566. [CrossRef]
11. Zhang, Y.; Duan, W.; Zhao, H. A hybrid heuristic framework for combinatorial optimization problems. *Appl. Soft Comput.* **2020**, *95*, 106516.
12. Saghaeeian, A.; Ramezanian, R. An efficient hybrid genetic algorithm for multi-product competitive supply chain network design with price-dependent demand. *Appl. Soft Comput.* **2018**, *71*, 872–893. [CrossRef]
13. Rezaei, S.; Kheirkhah, A. A comprehensive approach in designing a sustainable closed-loop supply chain network using cross-docking operations. *Comput. Math. Organ. Theory* **2018**, *24*, 51–98. [CrossRef]
14. Liu, Y.; As'arry, A.; Hassan, M.K.; Hairuddin, A.A.; Mohamad, H. Review of the grey wolf optimization algorithm: Variants and applications. *Neural Comput. Appl.* **2024**, *36*, 2713–2735. [CrossRef]
15. Amiriebrahimabadi, M.; Mansouri, N. A comprehensive survey of feature selection techniques based on whale optimization algorithm. *Multimed. Tools Appl.* **2024**, *83*, 47775–47846. [CrossRef]
16. Zamani, H.; Nadimi-Shahraki, M.H. An evolutionary crow search algorithm equipped with interactive memory mechanism to optimize artificial neural network for disease diagnosis. *Biomed. Signal Process. Control* **2024**, *90*, 105879. [CrossRef]
17. Ayati, A.; Naji, H.R.; Hashemi, M.M.; Saffar, M. Optimizing location allocation in urban management: A brief review. In Proceedings of the 29th International Computer Conference, Computer Society of Iran (CSICC), Tehran, Iran, 5–6 February 2025; pp. 1–7.
18. Sedeño, J.O. La efectividad del Tribunal Electoral y el INE: Los derechos humanos y los conflictos laborales en México. *Rev. Mex. Estud. Electorales* **2020**, *4*, 177–203.
19. National Electoral Institute (INE). Trámite de Credencial para Votar. 2022. Available online: https://www.ine.mx/credencial/tramite-credencial-tipo/ (accessed on 11 June 2025).
20. Jardón, E.; Romero, M.; Marcial-Romero, J.R. A model to optimize the allocation of public administrative services. *Comput. Sist.* **2025**, *29*, 229–239. [CrossRef]
21. Cerqueira, G.R.L.; Aguiar, S.S.; Marques, M. Modified greedy heuristic for the one-dimensional cutting stock problem. *J. Comb. Optim.* **2021**, *42*, 657–674. [CrossRef]
22. Chouman, M.; Crainic, T. *A MIP-Tabu Search Hybrid Framework for Multicommodity Capacitated Fixed-Charge Network Design*; CIRRELT Report; CIRRELT: Québec, QC, Canada, 2010; pp. 1–25.
23. Maniezzo, V.; Stützle, T.; Voß, S. *Matheuristics*; Springer: New York, NY, USA, 2021.
24. Franzin, A.; Stützle, T. Revisiting simulated annealing: A component-based analysis. *Comput. Oper. Res.* **2019**, *104*, 191–206. [CrossRef]
25. Burke, E.K.; Hyde, M.R.; Kendall, G. Grammatical evolution of local search heuristics. *IEEE Trans. Evol. Comput.* **2012**, *16*, 406–417. [CrossRef]
26. Aydın, D.; Yavuz, G.; Stützle, T. ABC-X: A generalized, automatically configurable artificial bee colony framework. *Swarm Intell.* **2017**, *11*, 1–38. [CrossRef]
27. Pérez Cáceres, L.; López-Ibáñez, M.; Stützle, T. Ant colony optimization on a limited budget of evaluations. *Swarm Intell.* **2015**, *9*, 103–124. [CrossRef]