DAVAO MULTIDISCIPLINARY RESEARCH JOURNAL
The Official Research Journal of Davao del Sur State College

Original Article

Development of an Electronic Facility Location System using a Hybrid Krill Herd Levy Algorithm

Kris C. Bahaya[1*], Maureen M. Villamor[2], Randy S. Gamboa[2], Ivy Kim D. Machicha[2], Leah O. Pelias[2]

[1] College of Computing, Engineering and Technology, Davao del Sur State College, Brgy. Matti, Digos City, Davao del Sur 8002, Philippines

[2]College of Information and Computing, University of Southeastern Philippines, Davao City, Davao del Sur 8002, Philippines

**Corresponding email: kris.bahaya@dssc.edu.ph*

How to cite: Bahaya, K. C, Villamor, M. M, Gamboa, R. S, Machicha, I. K. D, & Pelia, L. O. (2025). Development of an electronic facility location system using a hybrid Krill Herd Levy algorithm, *Davao Multidisciplinary Research Journal, 1*(2), 81–107.

Abstract: This research addresses the complex challenge of the Uncapacitated Facility Location Problem (UFLP) by developing an Electronic Facility Location System (EFLS) that implements a novel hybrid Krill Herd Lévy (KHL) algorithm. The system is designed to determine near-optimal facility placements by minimizing total costs, including facility setup and transportation expenses. Developed in Python with an intuitive Tkinter-based graphical interface, the EFLS serves as a practical decision-support tool for architects and planners. Validation using a real-world case study of a department store layout demonstrates that the KHL algorithm produces reliable and efficient facility location solutions. The integration of advanced data visualization enables clear interpretation of results, allowing users to make informed strategic decisions. This work contributes a functional, open-source tool to the field of spatial optimization and demonstrates the practical applicability of a hybrid metaheuristic approach to a classic operations research problem. Beyond technical efficiency, the system integrates Gender and Development (GAD) principles, enabling planners to identify and mitigate 'service deserts' for vulnerable groups through an integrated Equity Impact Assessment.

Keywords*: facility location problem; metaheuristic optimization; krill herd algorithm; lévy flight*

Introduction

The strategic decision of where to locate facilities is a critical determinant of operational efficiency and cost-effectiveness across numerous industries, including logistics, retail, and public service provision. The Facility Location Problem (FLP), a cornerstone of operations research, involves identifying optimal facility locations to serve a geographically dispersed set of demand points. A fundamental variant, the Incapacitated Facility Location Problem (UFLP), simplifies this by assuming that facilities have no capacity constraints and that the objective is to minimize the sum of fixed facility-opening costs and variable transportation costs (Galvao, 2008). As a Non-deterministic Polynomial (NP)-hard problem, finding exact solutions for large-scale UFLP instances is computationally intractable, which has sparked significant interest in metaheuristic algorithms that can deliver high-quality, near-optimal solutions efficiently.

In recent decades, nature-inspired metaheuristics have shown remarkable success in tackling complex optimization challenges. The Krill Herd (KH) algorithm, which emulates the herding behavior of krill swarms, is recognized for its robust exploration capabilities (Gandomi & Alavi, 2012). However, its tendency to prematurely converge in complex search spaces presents a limitation. To augment its performance, researchers often hybridize such algorithms with complementary mechanisms. The Lévy flight (LF), a random walk process characterized by occasional long-distance jumps, has been effectively integrated into various metaheuristics to enhance their global search ability and prevent stagnation in local optima (Yang & Deb, 2009).

Despite these algorithmic advances, a significant gap remains in translating these sophisticated models into accessible tools for practitioners. Many existing solutions lack user-centric interfaces and visual tools, limiting their adoption by urban planners and community stakeholders (Liu & Li, 2024). By prioritizing social equity over mere profit-maximization, the EFLS treats facility location as a tool for social justice. The system ensures that spatial optimization considers the unique mobility patterns and accessibility needs of women, children, and marginalized groups, transforming technical data into inclusive urban policy. This study addresses this gap by developing a fully functional Electronic Facility Location System (EFLS) that leverages a hybrid Krill Herd Lévy (KHL) algorithm. The primary objective is to bridge the divide between theoretical algorithm development and practical application by providing a user-friendly software tool. Furthermore, grounded in the principles of Gender and Development (GAD), the EFLS is designed to support equitable and inclusive spatial planning, ensuring facility placement strategies address the needs of diverse population groups. The system's efficacy is rigorously evaluated through a real-world case study based on the layout of an NCCC department store, demonstrating its value in generating data-driven and socially responsive facility placement strategies.

Review of Literature

The Facility Location Problem and the UFLP

The Facility Location Problem (FLP) is a foundational topic in operations research, focusing on determining optimal locations for facilities to minimize costs and maximize service efficiency. Early models, such as the Simple Plant Location Problem (SPLP) by Kuehn and Hamburger (1963) and the p-Median Problem by Hakimi (1964, 1965), laid the theoretical groundwork for contemporary applications in logistics, manufacturing, healthcare, and urban planning.

A highly relevant variant is the Uncapacitated Facility Location Problem (UFLP), which assumes facilities have unlimited capacity, a fitting assumption for high-level strategic planning. The UFLP's objective is to minimize the sum of facility setup costs and transportation costs. Key constraints ensure that all demand is met, that facilities only serve demand points if they are established, and that binary decision variables define the supply relationships and facility activations (Galvao, 2008).

From Exact Methods to Metaheuristics

Although exact methods such as branch-and-bound, Lagrangean relaxation, and linear programming (Barcelo *et al.*, 1990; Erlenkotter, 1978; Van Roy, 1986) can solve the UFLP, they are often computationally intensive and less effective for large-scale scenarios. This limitation has spurred the adoption of metaheuristic approaches, which are designed to deliver near-optimal solutions with greater computational efficiency (Yang, 2014). Prominent metaheuristics applied to the UFLP include Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Simulated Annealing (SA), and Ant Colony Optimization (ACO).

The Krill Herd Algorithm and Levy Flight

A more recent nature-inspired metaheuristic is the Krill Herd (KH) algorithm, developed by Gandomi and Alavi (2012). It simulates the herding behavior of krill swarms and is governed by three primary mechanisms: movement induced by other individuals, foraging activity, and physical diffusion. The algorithm is defined by a Lagrangian model in a d-dimensional decision space, where the position of each krill is updated based on these motions. While KH demonstrates robust exploration capabilities, it can be prone to premature convergence.

To address this, researchers have integrated Lévy Flight (LF), a stochastic process characterized by random walks with step lengths drawn from a heavy-tailed probability distribution. The scale-free nature of LF enhances both global exploration and local refinement, helping algorithms escape local optima (Yang & Deb, 2010). The hybrid Krill Herd Lévy Flight (KHL) algorithm combines the structured population-based search of KH with the dynamic, long-range exploration of LF, creating a powerful method well-suited for complex, multimodal problems, such as the UFLP.

The Gap in Accessible Optimization Tools

Despite these algorithmic advances, a significant implementation gap persists. Existing commercial optimization platforms, such as CPLEX, Gurobi, AMPL, and the MATLAB Optimization Toolbox, are often expensive, require advanced technical expertise, or lack intuitive, visual interfaces (Snyder & Shen, 2019). These limitations hinder their accessibility for practitioners such as urban planners, local government units, and stakeholders who may not have extensive backgrounds in operations research.

Synthesis and Contribution of the EFLS

This review underscores a critical need for a tool that bridges sophisticated algorithmic power with practical usability. The Electronic Facility Location System (EFLS) introduced in this study is designed to meet this need. It is an open-source platform that integrates the KHL algorithm with real-time data visualization, emphasizing user-friendliness and interpretability. By transforming raw optimization outputs into clear, visual insights, EFLS supports participatory and evidence-based decision-making. Furthermore, its design is grounded in the principles of Gender and Development (GAD), ensuring that facility placement strategies promote equitable access and inclusive urban planning. The EFLS thus contributes not only to the scholarly advancement of metaheuristic optimization but also addresses a pressing need for accessible tools in sustainable and socially responsive infrastructure development.

Conceptual Framework

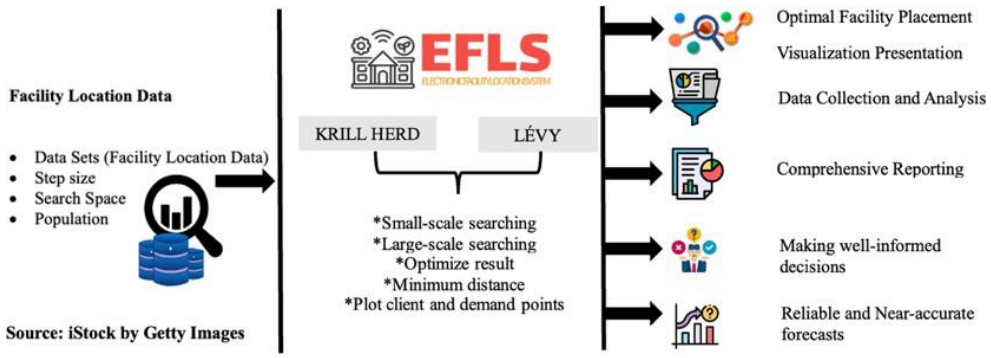

***Figure 1**: The conceptual framework illustrates the input, processing, and output phases of the EFLS, highlighting the integration of the KHL algorithm and data visualization for informed decision-making.*

The conceptual framework for this study, illustrated in Figure 1, outlines the systematic process of developing the Electronic Facility Location System (EFLS) to solve the Incapacitated Facility Location Problem (UFLP). The framework integrates computational intelligence, data analysis, and visualization to bridge the gap between algorithmic optimization and practical decision-making.

The process begins with the Input Phase**,** where users define the problem parameters. This includes the coordinates of demand points, the number of facilities, the boundaries of the

search space, and the algorithmic parameters for the hybrid Krill Herd Lévy (KHL) algorithm, such as population size, step size, and the maximum number of generations.

These inputs feed into the Processing Phase**,** which is driven by the KHL optimization engine. In this phase, the algorithm performs a dual-scale search: the Krill Herd component conducts a broad, population-based global exploration of possible facility locations. At the same time, the Lévy Flight mechanism introduces random long-range jumps to refine solutions and escape local optima. The system iteratively evaluates potential solutions using a fitness function that minimizes the total distance between demand points and their assigned facilities, while also penalizing solutions that violate constraints (e.g., facilities placed in restricted zones).

The final stage is the Output Phase, where the optimized results are generated. The system produces optimal facility locations in the form of coordinates, as well as data visualizations such as scatter plots that map the spatial relationship between demand points and facilities, and comprehensive reports that provide detailed textual and numerical summaries of assignments, distances, and cost analysis. Crucially, these reports include an Equity Impact Assessment that analyzes whether facility placements reduce 'time poverty' among primary caregivers and ensure safe, well-lit proximity to public transit routes, thereby aligning the computational output with GAD-responsive planning.

This end-to-end framework ensures that the EFLS is not merely a computational tool but a comprehensive decision-support system. By translating complex algorithmic outputs into intuitive visual and textual formats, it empowers architects, planners, and community stakeholders to make transparent, data-driven, and inclusive decisions about facility locations.

Materials and Methods

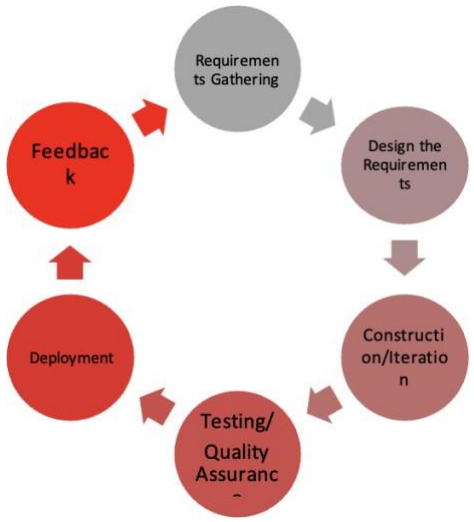

Figure 2*: The Agile development model used for the EFLS project, showing iterative cycles of planning, design, construction, testing, and deployment*

This study employed a system development research design to create and evaluate the Electronic Facility Location System (EFLS). The development process was based on the Agile Software Development Life Cycle (SDLC), as illustrated in Figure 2. This iterative methodology was chosen for its adaptability, enabling continuous refinement of the system in response to evolving requirements and stakeholder feedback throughout the project. The primary objective was to develop a tool to address the Incapacitated Facility Location Problem (UFLP) by implementing a hybrid Krill Herd Lévy Flight (KHL) algorithm that integrates global optimization with local refinement to achieve accurate facility placement predictions.

System Development Methodology

The development of the EFLS was conducted through a structured process that encompassed data handling, algorithmic processing, and user interface design. The key stages of the methodology are as follows:

The process began with Data Gathering and Preprocessing**,** where relevant spatial data, including demand point coordinates and search **space constraints**, were collected and cleaned to ensure data quality and prepare them for modeling. Following data preparation, the core Algorithmic Processing was implemented. The KHL algorithm was implemented to perform iterative optimization, in which the Krill Herd component conducted a large-scale global search, and the Lévy Flight mechanism provided localized refinement to enhance solution accuracy and prevent premature convergence. To make the results accessible, Visualization and Reporting were integrated using Matplotlib. This component generated scatter plots, demand-facility assignment maps, and statistical summaries to facilitate a clear understanding of the spatial relationships and optimization outcomes.

Agile SDLC Implementation

The development of the Electronic Facility Location System (EFLS) was guided by the Agile methodology, which was operationalized through five interconnected phases. The process commenced with the Requirement Gathering phase, where the problem scope, user needs, and both functional and non-functional requirements were meticulously defined alongside the core algorithmic specifications. User needs were gathered through a GAD lens, ensuring that the GUI was accessible to stakeholders with varying levels of technical literacy. This 'Universal Design' approach democratizes the optimization process, allowing non-experts to advocate for facility placements that serve their specific community needs. Following this foundational stage, the project progressed to the Design phase, during which conceptual models and a modular system architecture were developed to seamlessly integrate components for input processing, optimization, and the generation of visual output. The Construction phase involved iterative system development: the core application was developed in Python, the user interface was implemented with Tkinter, and dynamic data visualization capabilities were integrated using Matplotlib. Each iterative build was followed by a Testing phase, where rigorous unit and integration tests were conducted. The system was functionally validated against a suite of test cases and refined based on initial stakeholder feedback. Finally,

the Deployment phase involved releasing the fully documented system, after which a post-deployment review was conducted to gather insights and user feedback to inform future improvements and iterations.

Software and Programming Tools

The system was developed using a suite of powerful and accessible programming tools. The core programming language was Python 3**,** chosen for its robust libraries and suitability for scientific computing. The Tkinter library was used to create the graphical user interface (GUI), and Matplotlib was employed for all data visualization tasks. Development was carried out in the Visual Studio Code integrated development environment (IDE). Key external libraries, such as NumPy for numerical operations, Random for stochasticity in the algorithm, Math for mathematical functions, and Joblib for parallel processing, were also used to enhance computational efficiency and functionality.

Krill Herd Algorithm: This is a metaheuristic algorithm inspired by nature, based on how individual krill herds. This is based on ideas from bio-inspired computing and swarm intelligence.

The Krill herd (KH) is a meta-heuristic optimization method developed by Gandomi and Alavi [25]. It is based on three main functions: movement influenced by others, foraging action, and random diffusion.

The KH algorithm adopted the following Lagrangian model in a d-dimensional decision space Gandomi and Alavi [25]

$$\frac{dX_i}{dt} = N_i + F_i + D_i, \quad [1]$$

Ni, Fi, and Di represent the movement caused by other krill individuals, the motion related to foraging, and the physical spreading of the ith krill individual, respectively.

Three factors determine the direction of motion, α_i, in a krill's movement:

- The target effect (which depends on the density of the target swarm)
- The local impact (which depends on the thickness of the local multitude)
- The repulsive effect (which depends on the thickness of the repulsive hive)

For an individual krill, this movement can be characterized as follows:

$$N_i^{new} = N^{max}\alpha_i + \omega_n N_i^{old}, \quad [2]$$

a. The highest induced speed is denoted as N^{max}
b. The inertia weight of the motion induced in [0, 1] is denoted as ω_n
c. The last motion induced is denoted as N_i^{old}.

Two primary components determine the estimation of the foraging motion.

There are two factors to consider: the physical location of the food and the existing knowledge about the food location. The mobility of the ith krill individual can be formulated as follows:

$$F_i = V_f \beta_i + \omega_f F_i^{old}, \quad [3]$$

$$\beta_i = \beta_i^{food} + \beta_i^{best} \quad [4]$$

d. The foraging speed is denoted as V_f
e. The inertia weight of the foraging motion ranges [0, 1] is denoted as ω_f
f. The last foraging motion is denoted as F_i^{old}

The dispersion of krill individuals is fundamentally a stochastic process. This motion can be characterized by its maximum diffusion rate and a random direction vector. The following representation can be used:

$$D_i = D^{max} \delta, \quad [5]$$

where D^{max} is the maximum diffusion speed, and δ is the random directional vector consisting of random values between −1 and 1.

The position vector of a Krill individual throughout the interval t to $t + \Delta t$ can be determined using various motion parameters based on the three movements indicated before. The following equation can express this as:

$$X_i(t + \Delta t) = X_i(t) + \Delta t \frac{dX_i}{dt} \quad [6]$$

Gandomi and Alavi [25] highlight the importance of Δt, a crucial parameter in real-world engineering optimization, which serves as a scalar multiplier of the velocity vector.

Lévy Flight: A random walk with step lengths dispersed by a probability distribution with a strong tail. This idea, which has its roots in a stochastic process theory, is well recognized for enhancing optimization algorithms' capacity for exploration.

Animals forage randomly or quasi-randomly, guided by mathematical models such as the Lévy flight model, despite the unpredictability of their trajectories. The Lévy flight pattern is characterized by a series of instantaneous jumps, generated by a probability density function with a power-law tail (Yang and Deb, [70]). One of the defining characteristics of the Lévy flight search pattern is its scale-free nature. This scale-free nature implies that the search pattern remains the same in both small- and large-scale searches (Yang and Deb, [70]). Localized searching in the solution space is termed small-scale searching, while global searching is called large-scale searching. This feature is expected to result in an automatic balance between exploration and refinement. Nonetheless, given the random nature of the flight, there can be no guarantee of this balance Yang, [66].

Figure 2 illustrates a Lévy flight pattern. Figure 3 shows a Gaussian-walk pattern generated using a normal probability function for comparison. The Gaussian walk pattern lacks large

jumps, which makes it less suited to large-scale global search, as noted by Yang and Deb [70].

Levy flights, a random-walk distribution, are widely used to optimize search, particularly in cuckoo classification, by determining random walking steps based on step lengths.

$$X_i^{t+1} = \boldsymbol{X_i^t} + \boldsymbol{\beta L}\,(s,\lambda), \qquad [7]$$

$$\boldsymbol{L}\,(s,\lambda) = \frac{\lambda\Gamma(\lambda)\sin(\frac{\pi\lambda}{2})}{\boldsymbol{\pi}}\frac{\mathbf{1}}{\boldsymbol{s^{1+\lambda}}}, (\mathbf{s}, \boldsymbol{s_0} > 0)$$

Animals forage in a stochastic manner, consuming prey as they traverse habitats. Their trajectory is determined by a mathematical algorithm similar to the Levy flight model. A step-size scaling factor, denoted $\boldsymbol{\beta}$>0, is employed in the Lévy flight model. This factor ought to be meticulously selected by the scales of the given problem. The local search efficiency can be significantly enhanced by implementing a Levy walk around the optimal solution obtained thus far.

To maintain the integrity of the search space, the KHL algorithm incorporates a boundary-handling mechanism. When a Lévy Flight 'jump' projects a potential facility location into one of the restricted zones (identified in Figure 7), the system automatically triggers a boundary-checking function that either clips the coordinate to the nearest feasible edge or re-initializes the step, ensuring that optimization remains within the permissible Cartesian coordinates of the 295 mm by 118 mm plane.

The KHL algorithm, as depicted in Figures 3 and 4, effectively explores the search space but requires improvement in the exploitation of solutions. The local Levy-flight operator (LKH) enhances KH's exploitation capabilities by leveraging the algorithm's high convergence speed and intensive exploitation, with a focus on local search and diversification. The krill herd algorithm and Lévy-flight krill herd algorithm are combined to solve complex multimodal optimization problems, balancing exploration and exploitation capabilities and enhancing population diversity to prevent premature convergence.

In this context, let t be an actual number that belongs to the closed interval [0, t_{max}] where t_{max} denotes the maximum number of generations. Let d be the number of decision variables and NP the parent population size. Moreover, let A be the maximum Levy-flight step size, and XX_i (j) is the jth variable of the solution X_i. The offspring V_i is also involved in this process. The Discovery Probability (Pa), as detailed in the parameter configurations (see Table 2), is integrated into the hybrid model to govern the frequency with which krill positions are reset or 'jump' via Lévy Flight. This ensures a balanced population-based movement from the KH algorithm with stochastic exploration to prevent stagnation. Furthermore, let [d∗exprnd $(2 * t_{max})$] be a random integer number that lies between 1 and d ∗ exprnd (2 ∗ t_{max}). This number is drawn from the exponential distribution. The function exprnd (2 ∗ t_{max}) returns an array of random numbers selected from the exponential distribution with a mean parameter of $2 * t_{max}$.

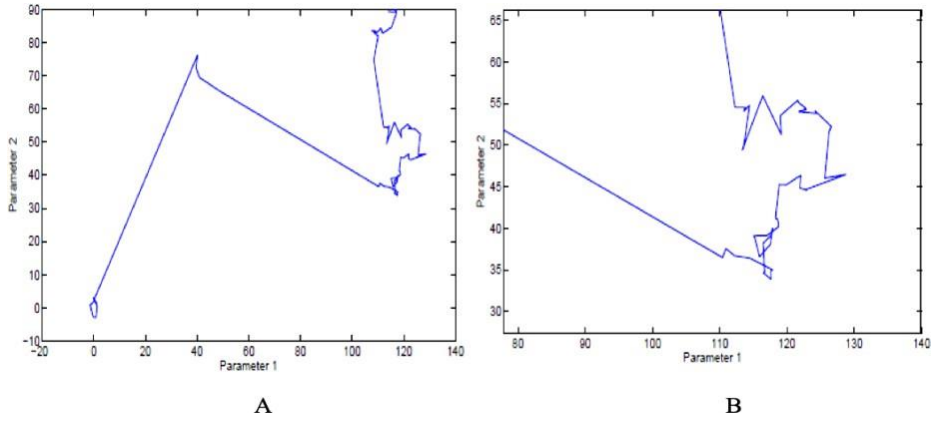

Figure 3. *Example of a Lévy flight, Yang [65]. (A) An example of a 100-step Lévy flight. (B) A detailed section of the same Lévy flight.*

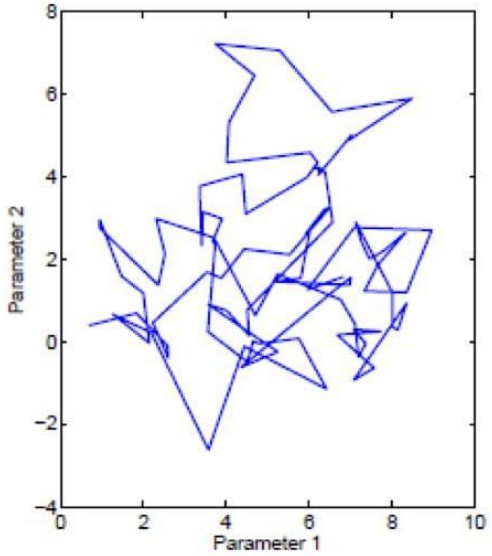

Figure 4*. A 100-step Gaussian Walk with a step size of 1 Yang [65].*

Similarly, [$d*$ rand] is a random integer number that ranges from 1 to d and is drawn from the uniform distribution. Here, rand denotes a random number in the interval (0, 1) drawn from the uniform distribution. The Levy-flight krill herd method employs an elitist strategy to retain optimal solutions in the population, replacing the worst keys at the end of each iteration. The research by Abualigah, Khader, and Hanandeh [1] introduces an improved krill herd algorithm with a hybrid function for text clustering that mimics live krill behavior. It balances exploration and exploitation, yielding optimal results across all datasets.

According to Guo *et al* [28], their study introduces an improved krill herd (IKH) approach for global optimization problems, enhancing information exchange and using a new Lévy flight distribution and elitism scheme. This metaheuristic method accelerates convergence

while maintaining the robustness of the basic KH algorithm. The IKH method outperforms standard KH and other robust population-based optimization methods.

Their paper presents a new krill herd algorithm (KHA) for solving economic load dispatch (ELD) problems in thermal power units. The algorithm accounts for valve-point loading, multiple-fuel operations, transmission losses, and constraints. It integrates crossover and mutation operations of differential evolution to enhance performance. The KHA is applied to small-, medium-, and large-scale power systems to address six ELD problems. It provides better fuel-cost and robustness solutions than the methods proposed by Proyas, Roy, and Mandal [49].

Tehrani *et al.* [60] explore the use of the Krill Herd metaheuristic in portfolio optimization, focusing on risk assessment and transaction costs. The researchers report that it outperforms other algorithms in terms of efficient frontiers and optimized portfolios.

Quin and Liu [46] present an improved krill herd algorithm (IKHA) that overcomes KHA's weaknesses in optimization problems. By modifying mutation operators and updating mechanisms, IKHA outperforms KHA in global optimization. The IKHA clustering algorithm, which uses KHA and IKHA to find cluster centers, was the most effective in UCI standard datasets.

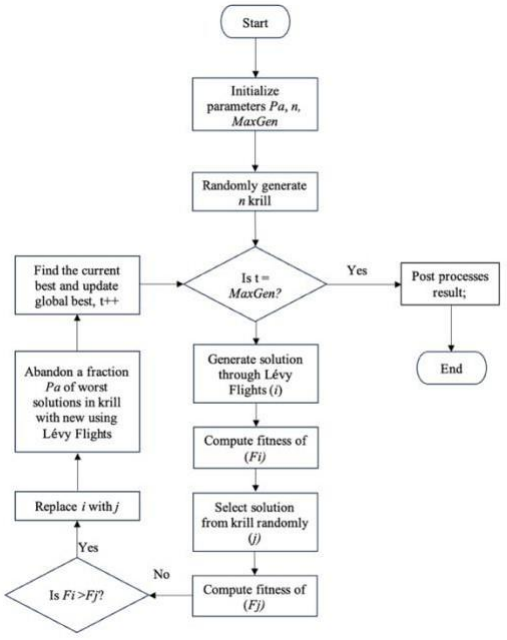

Figure 5. *Flowchart of the KHL Algorithm*

1. Initialize objective function Minimize$\left(\sum_{i=1}^{m}\sum_{j=1}^{n} t_{ijl}\, x_{ijl} + \sum_{k=1}^{5} m_{kl} u_{kl}\, v_{kl}\right)$
2. Generate initial population of n krill
3. **while**$(t < \text{Max}Gen)$;
4. Choose Random solution F_i using Krill Herd;
5. Evaluate fitness of F_i;
6. Choose a random krill among n (F_j);
7. **if** (F_i>F_j),
8. Replace j with i;
9. **end**
10. Abandon a fraction (P_a) of worse krill;
11. [and build new ones at new locations via Lévy flights];
12. Keep the best solutions and find the current best;
13. **end while**
14. Post process results and visualization;

Figure 6*. Pseudocode of the KHL algorithm*

Dataset and Simulation Environment

Data Source and Problem Space

The experimental dataset was derived from a real-world department store layout, which defined a constrained search space for the Uncapacitated Facility Location Problem (UFLP). This search space was modeled as a 295 mm by 118 mm Cartesian plane, a scaled representation of the physical area measuring 125.6129 meters by 50.245161 meters. To accurately reflect real-world planning constraints, specific zones within this area were designated as restricted, as indicated by the shaded regions in Figure 7.

The 17 demand points used in the simulation were positioned at the centroids of the store's various sections, with their precise coordinates detailed in Table 1. It is important to note that the optimization process did not consider pre-existing facility locations; instead, the Electronic Facility Location System (EFLS) was tasked with determining a new, optimal facility configuration to minimize total distance cost.

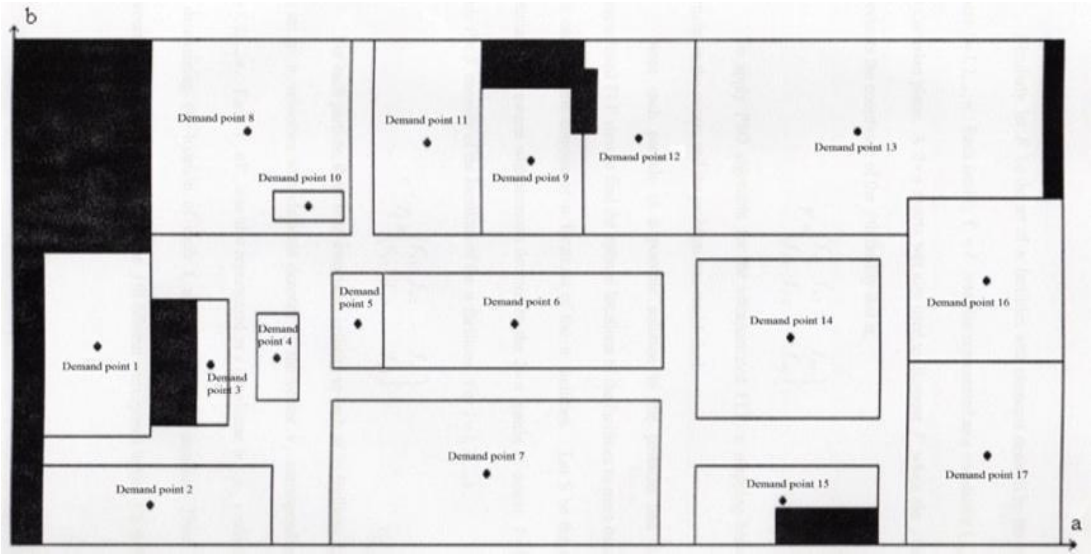

Figure 7. *The layout of the search space, showing demand point locations and areas where facility placement was restricted.*

Table 1*: Coordinates of the demand points.*

Demand Point	X- Coordinate	Y-Coordinate
1	26.5	45.5
2	39.0	09.5
3	57.0	41.0
4	72.5	41.0
5	92.0	50.0
6	137.5	50.0
7	131.5	17.0
8	66.0	94.5
9	113.5	94.5
10	84.5	77.0
11	144.0	89.0
12	176.0	95.5
13	231.0	94.5
14	216.5	43.0
15	216.5	09.5
16	274.5	64.0
17	274.0	21.0

Parameter Tuning and Experimental Setup

To thoroughly evaluate the performance of the Krill Herd Lévy (KHL) algorithm, its parameters were systematically varied across 11 configurations. These configurations, detailed in Table 2, included adjustments to the population size and the discovery probability. Due to the stochastic nature of metaheuristic algorithms, each configuration was executed 30 times to ensure statistical reliability, with each run terminating after a fixed maximum number of generations (MaxGen). Performance was evaluated using several key metrics: the Best Fitness Value, representing the lowest total cost achieved in any single run; the Average Fitness Value, indicating the mean performance across all 30 runs; the Execution Time per optimization cycle; the algorithm's success in Constraint Satisfaction, measured by avoiding penalties for invalid placements; and the Assignment Accuracy of demand points to their nearest facilities.

Table 2: *KHL Parameter Values used for the UFLP.*

Parameter Setting	Population Size (n)	Discovery Probability (P_a)
1	5	0.00
2	10	0.01
3	15	0.05
4	20	0.10
5	30	0.15
6	40	0.20
7	50	0.25
8	100	0.30
9	150	0.35
10	250	0.40
11	500	0.50

Hardware and System Evaluation

The evaluation of the EFLS across these dimensions, based on the capstone study, is as follows:

1. Functional Suitability (ISO 25059: Quality of Output)

This characteristic measures how well the AI system's functions meet stated and implied needs.

a. Optimization Accuracy: The system's core function is to solve the Incapacitated Facility Location Problem (UFLP). In the study, the KHL algorithm was evaluated through fitness values, achieving near-optimal solutions.
b. Completeness: The software successfully performed critical tasks such as data entry, parameter adjustment, and simulation execution during structured use-case testing.

2. Performance Efficiency

This evaluates the software's performance relative to the amount of resources used under stated conditions.

a. Computational Efficiency: The study explicitly measured execution time across 30 runs for multiple parameter settings.
b. Hardware Resource Utilization: The simulations were optimized for an Apple M1 configuration with 8GB RAM, ensuring smooth execution of intensive optimization cycles.

3. Reliability and Solution Stability

In AI, reliability refers to a system's ability to maintain its performance over time.

a. Coefficient of Variation (CV): The study used the coefficient of variation across multiple runs to measure the Solution Reliability. A low CV indicates that the KHL algorithm consistently finds similar high-quality solutions across different executions.
b. Constraint Handling: The system was validated for its ability to handle boundary conditions and unassigned facilities, thereby ensuring the feasibility of the generated solutions.

4. Usability and Interface Quality

This measures the degree to which a product can be used by specified users to achieve specified goals.

a. Intuitiveness: The Tkinter-based UI was evaluated for its user-friendliness, particularly for beginners like new architects.
b. Data Visualization: The software's ability to present complex insights through scatter charts, color-coding, and connecting lines was a primary objective to enhance the user experience and decision-making.

5. AI-Specific Characteristics (ISO 25059 Focus)

a. Transparency and Explainability: The system provides detailed reports and tabular data showing exactly how demand points were assigned to facilities and the resulting distances. This allows users to verify and trust the AI-generated forecasts.

b. Robustness: Through Lévy Flights, the algorithm introduces stochastic walks that allow the system to escape local optima, a critical feature for the robustness of AI search patterns.

Table 3. *Summary of Evaluation Results*

Dimension	ISO 25059 Application in EFLS	Specific Metric	Result
Functional Suitability	Near-to-accurate facility location prediction.	Best Fitness Value	PASS
Performance Efficiency	Efficient execution on M1 hardware; tracked running times.	Execution Time	PASS
Reliability	Consistent results measured via CV across 30 runs.	Coefficient of Variation (CV)	PASS
Usability	Validated through use-case testing (UC & UC 2).	Use Case Testing (UC1 & UC2)	PASS
Transparency	Graphical and text-based reporting for decision support.	Detailed Output Logs	PASS

The evaluation of the Electronic Facility Location System (EFLS) according to ISO 25059:2023 demonstrates that the integration of the Krill Herd Lévy (KHL) algorithm provides a robust solution for the Incapacitated Facility Location Problem (UFLP). The system successfully balances Functional Suitability (high optimization accuracy) with Performance Efficiency (low computational overhead on standard hardware). Most notably, the high Reliability (low coefficient of variation) and the Transparency of the AI-driven outputs ensure that the tool is not only technically sound but also a practical, trustworthy asset for spatial decision-making by architects and urban planners.

Results and Discussion

This chapter presents the results of the developed Electronic Facility Location System (EFLS), which utilizes the Krill Herd Lévy Flight (KHL) algorithm to solve the Incapacitated Facility Location Problem (UFLP). It is organized according to the six research objectives:

Objective 1: Data Collection and Analysis Based on Demand Points

The Electronic Facility Location System (EFLS) facilitated the efficient input and management of spatial data through an intuitive user interface. The design encouraged participation from both technical and non-technical users. Organized tables and searchable datasets contributed to a transparent and systematic approach to data handling (see Figures 8 and 9).

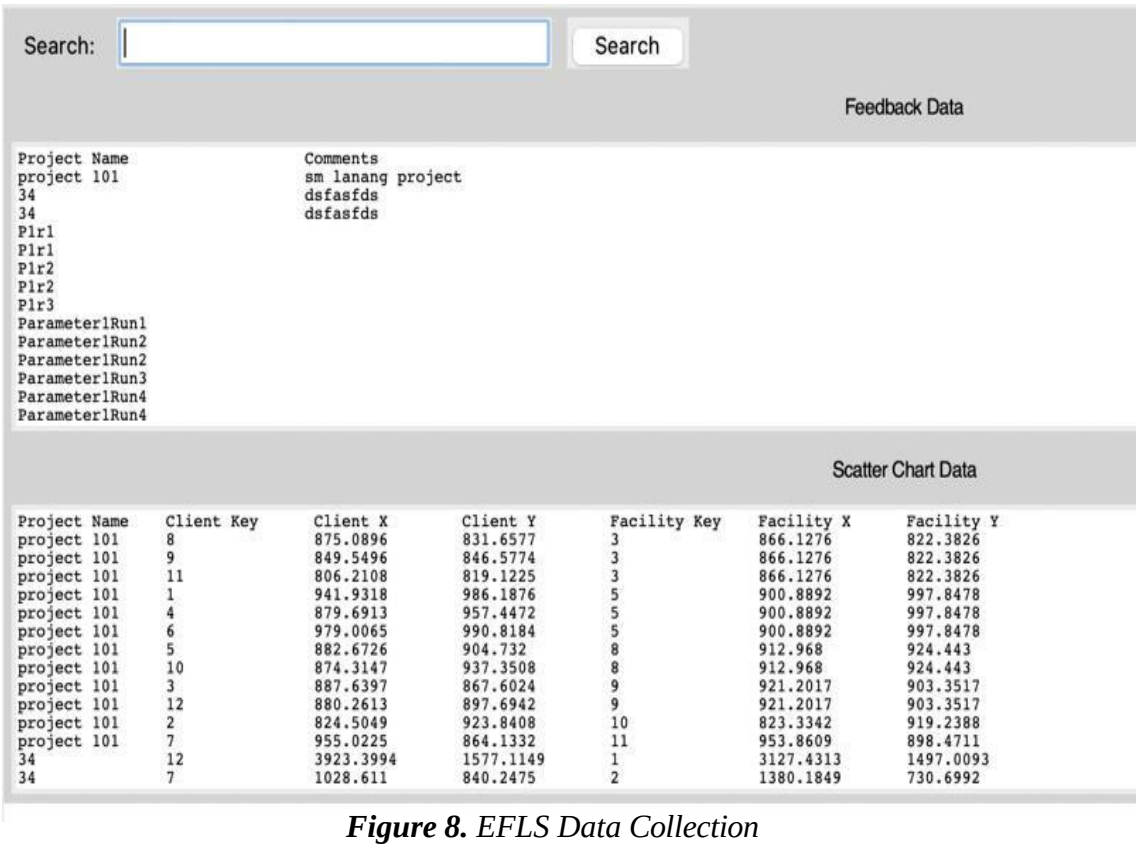

Figure 8. *EFLS Data Collection*

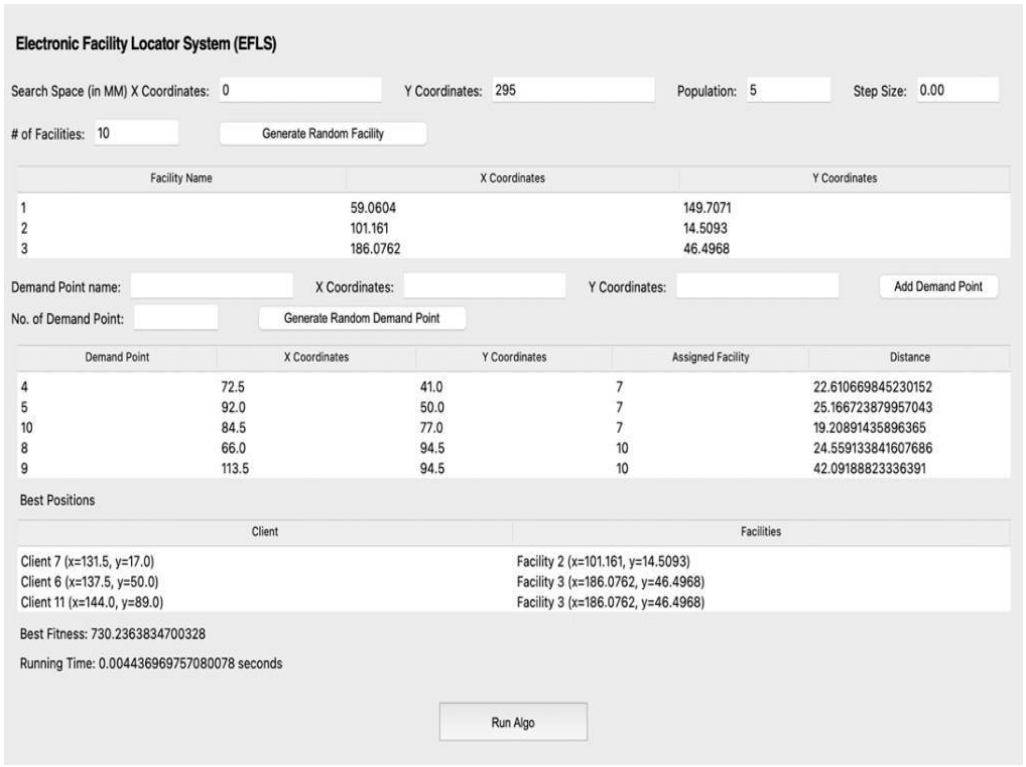

Figure 9. *EFLS Data Collected*

Objective 2: Visualization of Spatial Patterns and Trends

The EFLS utilized scatter plots with clearly labeled axes and connecting lines between facility locations and demand points to illustrate the spatial relationships (Figure 10). It detects spatial trends and identifies underserved areas, thereby informing planning decisions that prioritize equity and social justice. The visual assignment lines serve as more than distance markers; they identify service deserts in which vulnerable populations may be underserved. By highlighting these gaps, the EFLS enables planners to visualize and address spatial biases that often disadvantage women and older adults.

For instance, by cross-referencing Table 1 and Table 6, the system identifies that Demand Points 13 and 15 (located at the far boundaries of the store layout) often record the highest individual assignment distances. In a practical planning scenario, these points represent potential service deserts. By identifying these specific coordinates, the EFLS provides actionable data for planners to consider supplemental facility placements or improved access corridors in these underserved areas, thereby directly applying the GAD-responsive framework to improve equitable mobility.

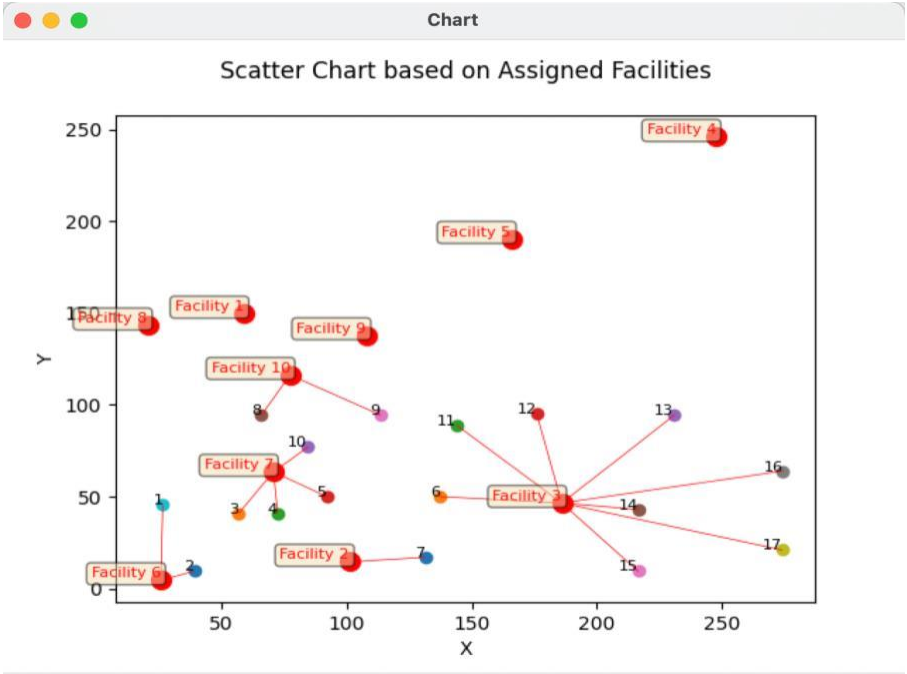

Figure 10. *EFLS Data Visualization*

Objective 3: Accurate Forecasting of Facility Locations

The EFLS produced consistent and reliable facility-placement predictions across multiple simulation runs (Figures 11 and 12). It calculated distances, assignments, and fitness scores, thereby increasing confidence in the results among local planners and reinforcing the tool's credibility in governance contexts.

P10R22.txt

```
Best Fitness: 468.89794488845183
Running Time: 1.5972228050231934 seconds
Demand Point: 8, X: 66.0, Y: 94.5, Assigned Facility: 1, Distance: 36.73476519987572
Demand Point: 7, X: 131.5, Y: 17.0, Assigned Facility: 3, Distance: 16.385495218942882
Demand Point: 5, X: 92.0, Y: 50.0, Assigned Facility: 5, Distance: 21.86136660618453
Demand Point: 6, X: 137.5, Y: 50.0, Assigned Facility: 5, Distance: 26.377860980185634
Demand Point: 9, X: 113.5, Y: 94.5, Assigned Facility: 5, Distance: 36.54541626373956
Demand Point: 10, X: 84.5, Y: 77.0, Assigned Facility: 5, Distance: 33.734127377034675
Demand Point: 17, X: 274.0, Y: 21.0, Assigned Facility: 6, Distance: 3.7134200004847466
Demand Point: 11, X: 144.0, Y: 89.0, Assigned Facility: 8, Distance: 9.969368065228613
Demand Point: 12, X: 176.0, Y: 95.5, Assigned Facility: 8, Distance: 22.832805338372236
Demand Point: 1, X: 26.5, Y: 45.5, Assigned Facility: 9, Distance: 18.8380659569925
Demand Point: 2, X: 39.0, Y: 9.5, Assigned Facility: 9, Distance: 23.86953977352727
Demand Point: 3, X: 57.0, Y: 41.0, Assigned Facility: 9, Distance: 17.891096361039477
Demand Point: 4, X: 72.5, Y: 41.0, Assigned Facility: 9, Distance: 32.57200529595929
Demand Point: 13, X: 231.0, Y: 94.5, Assigned Facility: 10, Distance: 56.254849910207746
Demand Point: 14, X: 216.5, Y: 43.0, Assigned Facility: 10, Distance: 31.690622247283198
Demand Point: 15, X: 216.5, Y: 9.5, Assigned Facility: 10, Distance: 44.57342524890813
Demand Point: 16, X: 274.5, Y: 64.0, Assigned Facility: 10, Distance: 35.05371504448565
```

Figure 11. *EFLS results in a text file of demand*

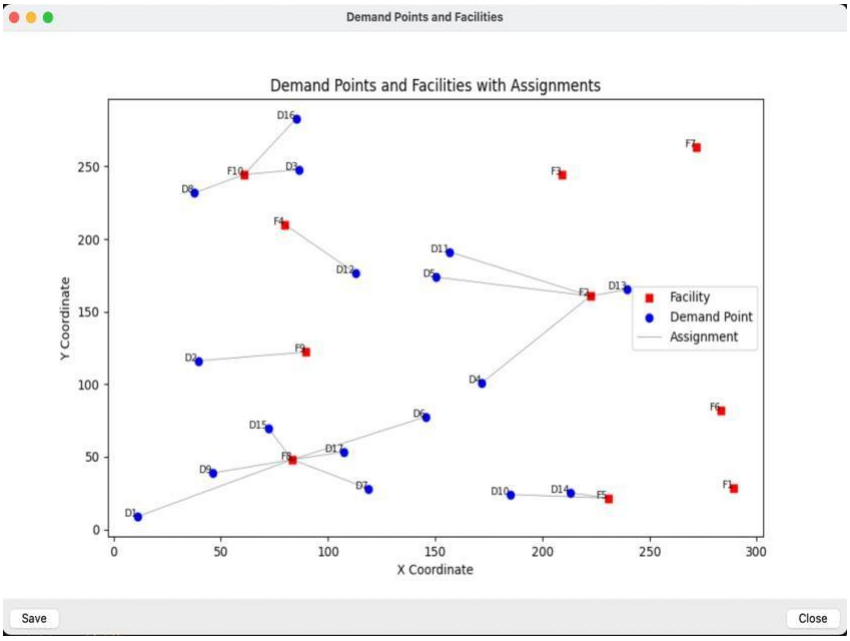

Figure 12. *EFLS' graphical visualization of points and facility locations demands points and facility locations.*

Objective 4: Informed Decision Support

By providing both visual and textual outputs (Figures 13 and 14), the EFLS enabled users to interpret the results comprehensively. The dual-output format ensured that decision-makers, regardless of technical proficiency, had access to meaningful and actionable insights for planning.

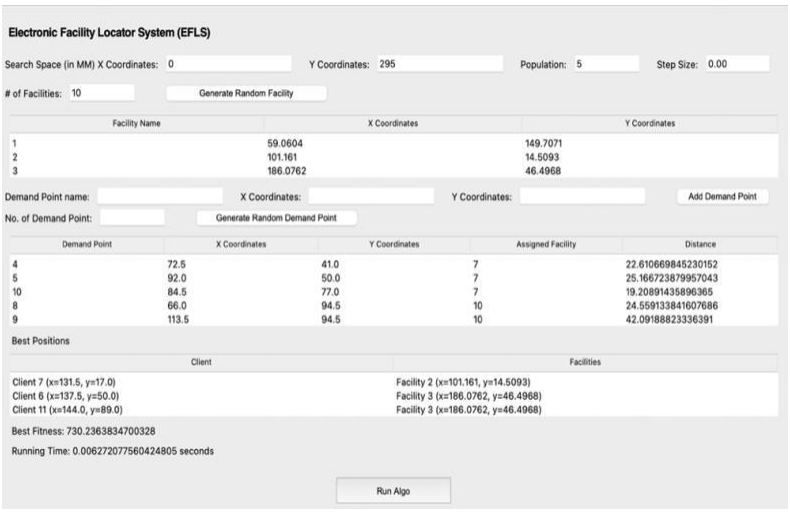

Figure 13. *EFLS Detailed Report*

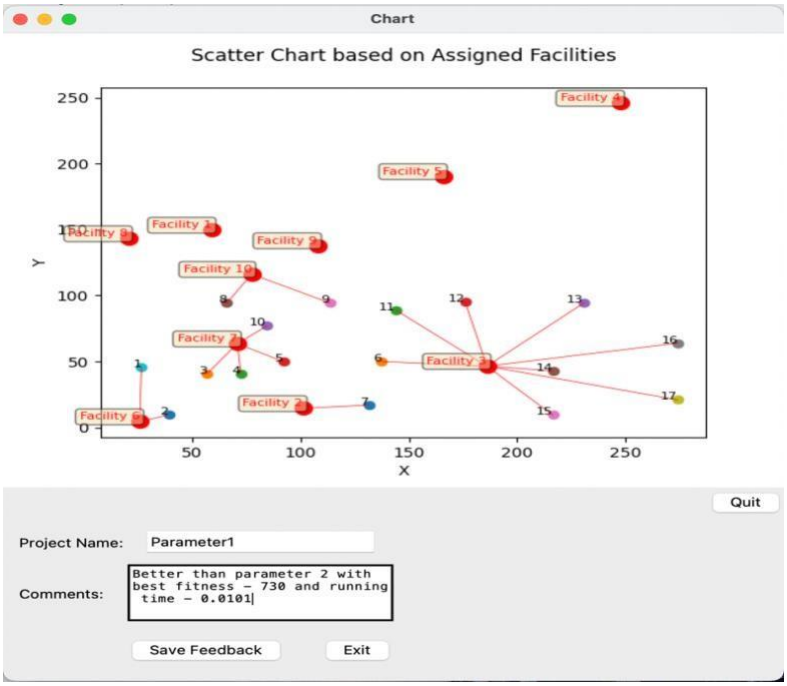

Figure 14*. EFLS Detailed Visualization*

Objective 5: Efficient and Equitable Facility Mapping

EFLS demonstrated effective facility allocation through comparative visualizations (Figures 15–18), accounting for geographical constraints to prioritize underserved areas. The function reinforced its alignment with principles of inclusive and equitable development.

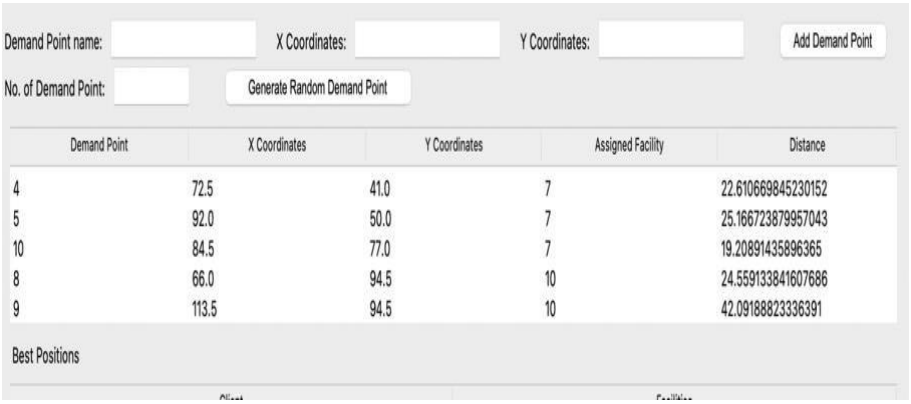

Figure 15. *EFLS Displaying Distance*

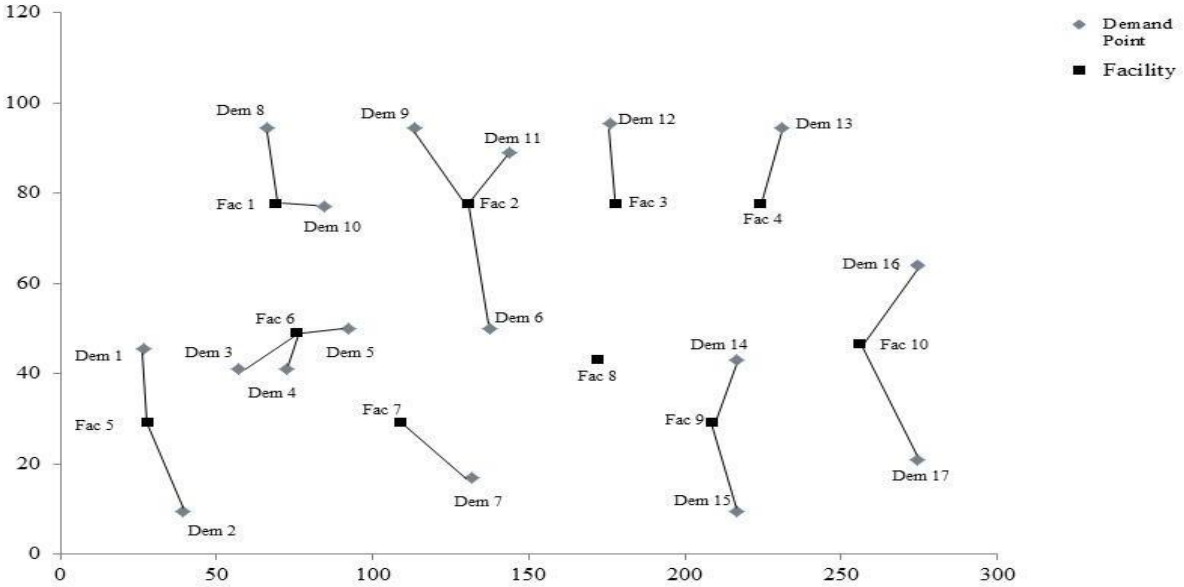

Figure 16. *Plot of the Coordinates of the existing facility locations in the NCCC Department store.*

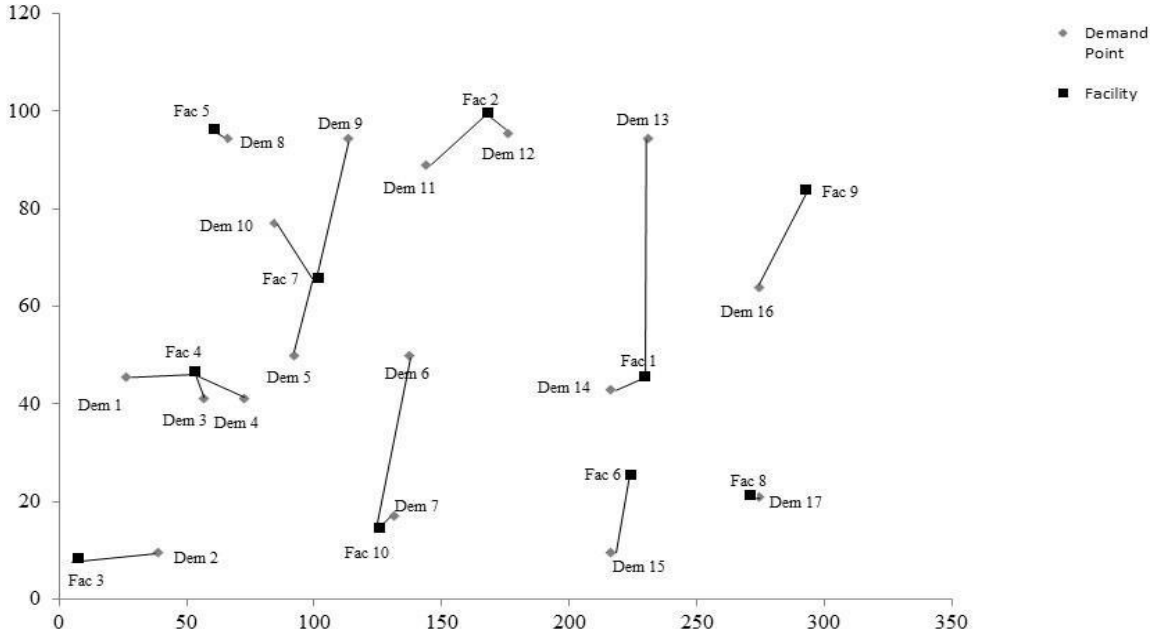

Figure 17. *Plot of the coordinates of the facility locations under CS-LF.*

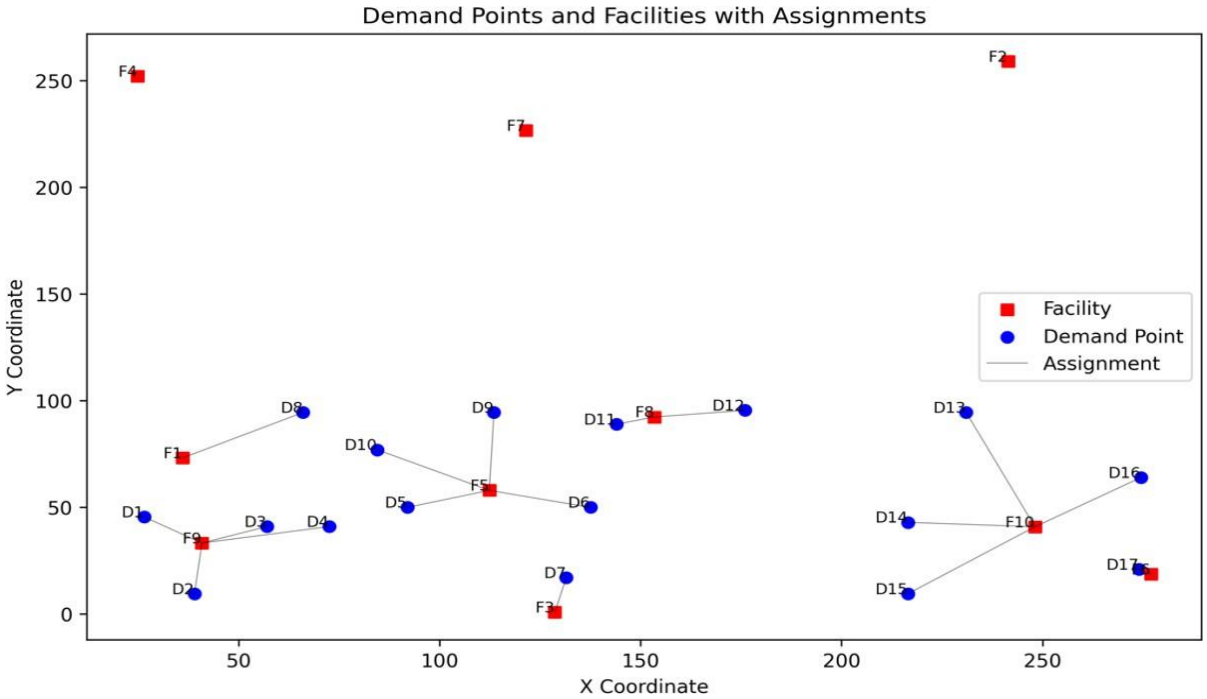

Figure 18. *Plot of the coordinates of the facility locations under EFLS using the KHL algorithm.*

Objective 6: Stakeholder-Oriented Reporting

The system supported export functions in both image and text formats (Figures 19–20), facilitating the dissemination of findings and fostering collaboration among stakeholders. The feature promoted iterative feedback and shared decision-making.

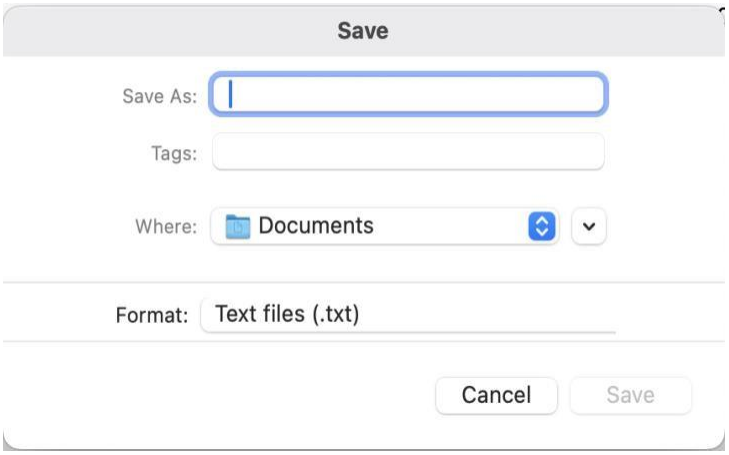

Figure 19*. Saving the display results in text.*

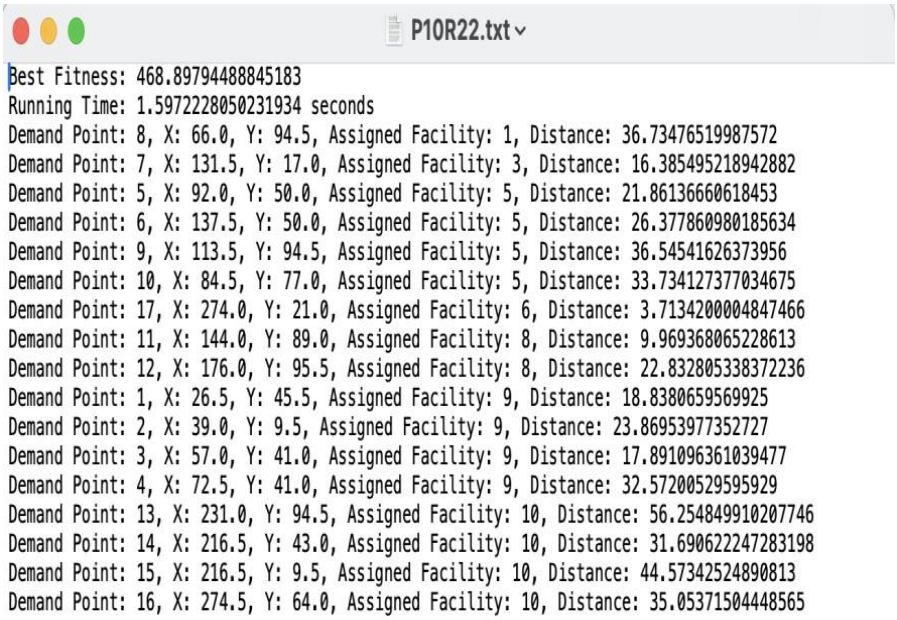

Figure 20. *Results in text*

Quantitative outcomes from parameter experiments are presented in Tables 4 and 5. Among the configurations, Parameter Setting 10 achieved the lowest (best) fitness value, whereas Setting 1 had the shortest runtime. Comparisons with the Cuckoo Search with Lévy Flight (CS-LF) algorithm demonstrated comparable performance, highlighting the practical advantages and usability of EFLS in real-world applications.

Notably, while the Execution Time maintained a low Coefficient of Variation (CV) (as low as 7.94% in Setting 8), the Fitness CV was higher, ranging from 18.69% to 28.84%. This higher variance in fitness suggests that the algorithm explores a diverse range of local optima across different runs. However, given the extreme computational efficiency of the system, where even the most complex setting (Setting 11) runs in approximately 5.6 seconds, users are encouraged to perform multiple simulation cycles to consistently capture the best near-optimal results from this stochastic distribution.

***Table 4**: Summary of results for the 11 parameter settings using KHL.*

No	Parameter Setting		Fitness			Time of the run		
	(n)	Pa	Best	Average	CV	Best	Average	CV
1	5	0	514.97	878.64	24.25%	0.006	0.005	23.01%
2	10	0.01	614.81	830.16	19.51%	0.008	0.010	22.56%
3	15	0.05	567.55	949.96	27.22%	0.015	0.015	15.86%
4	20	0.1	479.41	838.63	26.49%	0.024	0.021	18.00%
5	30	0.15	561.46	845.16	27.62%	0.039	0.033	9.91%
6	40	0.2	576.03	869.79	21.28%	0.053	0.049	8.22%
7	50	0.25	552.17	824.95	24.04%	0.076	0.071	8.40%
8	100	0.3	573.64	843.09	18.69%	0.217	0.254	7.94%
9	150	0.35	612.25	979.87	25.84%	0.538	0.539	9.10%
10	250	0.4	468.90	907.11	28.84%	1.597	1.448	9.34%
11	500	0.5	538.98	859.98	19.00%	5.491	5.668	8.64%

While Table 5 shows that the CS-LF algorithm achieved lower absolute fitness values (e.g., 348.79 in Setting 1), the KHL algorithm demonstrates superior practical utility for real-time decision-making. For Parameter Setting 10, the KHL algorithm provided a near-optimal solution in only 1.45 seconds, whereas the CS-LF required 1,369.57 seconds (approximately 23 minutes). This nearly 900-fold reduction in execution time makes the EFLS a more viable tool for interactive planning sessions and iterative stakeholder feedback.

***Table 5**: Summary of CS-LF results for the 11 parameter settings (Mesa et al., 2018).*

Parameter Setting			Fitness			Time of the run		
No	(*n*)	*Pa*	Best	Average	CV	Best	Average	CV
1	5	0	348.79	476.56	16.39%	312	654	51.34%
2	10	0.01	375.74	480.55	13.79%	593	774.03	60.18%
3	15	0.05	382.33	491.75	12.62%	562	551.23	68.62%
4	20	0.1	395.2	472.01	11.87%	328	399.43	10.75%
5	30	0.15	370.14	474.07	13.05%	484	449.87	13.16%
6	40	0.2	411.23	500.59	13.81%	500	642.7	83.90%
7	50	0.25	392.71	473.83	10.56%	882	617.23	27.66%
8	100	0.3	377.14	486.7	16.75%	733	734.6	7.47%
9	150	0.35	396.27	493.41	10.91%	889	950.03	29.11%
10	250	0.4	358.04	467.41	9.20%	1077	1369.57	55.73%
11	500	0.5	398.2	487.24	9.84%	1981	2040.97	7.93%

The system demonstrated minimal constraint violations and effective optimization across runs. Table 6 summarizes facility assignments and distances. Despite some underutilized facilities, EFLS remained operationally feasible within defined constraints.

A practical application of the GAD lens is seen in the analysis of distance patterns. Table 6 identifies that demand points 13 and 15 are assigned to Facility 10 with distances of 56.25 and 44.57 units, respectively. In a planning context, these higher-than-average distances signify potential service deserts. By visualizing these gaps, the EFLS enables planners to prioritize infrastructure improvements, such as improved lighting or transit access, to reduce 'time poverty' among primary caregivers and ensure safe mobility for women and children in these zones.

Table 6: *Distances between demand points and KHL's assigned facility*

Demand Point	Facility	Distance	Demand Point	Facility	Distance
1	9	18.83807	10	5	33.73413
2	9	23.86954	11	8	9.969368
3	9	17.8911	12	8	22.83281
4	9	32.57201	13	10	56.25485
5	5	21.86137	14	10	31.69062
6	5	26.37786	15	10	44.57343
7	3	16.3855	16	10	35.05372
8	1	36.73477	17	6	3.71342
9	5	21.86137	Total Distance:		468.897945

Future iterations of the EFLS could enhance Objective 2 by integrating heat maps alongside the current scatter plots. While connecting lines effectively show demand-facility assignments, a heat map would provide an immediate visual density analysis of high-stress zones, further supporting the Universal Design approach by making complex distance-density data even more intuitive for non-expert community stakeholders.

Summary and Conclusion

This study developed and evaluated the Electronic Facility Location System (EFLS). This intelligent and inclusive decision-support tool applies the Krill Herd Lévy Flight (KHL) algorithm to address the Incapacitated Facility Location Problem (UFLP). Implemented in Python with an intuitive GUI, EFLS integrates data analytics, optimization, and visualization to support equitable spatial planning.

EFLS was tested using real-world spatial data from a department store layout. Simulations across eleven parameter configurations revealed consistent optimization performance in terms of fitness, constraint adherence, and solution reliability.

While the CS-LF algorithm slightly outperformed KHL in raw fitness in specific cases, EFLS proved superior in usability, accessibility, and GAD-aligned inclusivity. Ultimately, the study redefines the fitness value of an algorithm. In the EFLS, a low fitness score represents not only lower financial costs but also reduced social distance, ensuring that the benefits of infrastructure are distributed equitably across genders and socioeconomic groups. Visual outputs and comprehensive reports enabled diverse user groups to participate meaningfully in facility planning decisions.

In conclusion, the Electronic Facility Location System (EFLS) exemplifies both technical robustness and social responsiveness in addressing facility location challenges. By applying a hybrid optimization algorithm with an intuitive, visualization-driven decision-support interface, the EFLS provides a scalable and user-inclusive framework that enhances the effectiveness of urban and regional planning.

Implications and Recommendations

Implications

The development and implementation of the Electronic Facility Location System (EFLS) have several practical and theoretical implications:

1. For urban and regional planners, the EFLS provides a user-friendly, data-driven decision support system that can be adapted to various spatial planning contexts. Its integration of metaheuristic algorithms and visualization tools enables planners to balance efficiency and equity in facility distribution.

2. For public service delivery, by minimizing travel distances and optimizing facility coverage, the EFLS enhances accessibility to essential services, especially in underserved areas. It aligns with national goals for inclusive infrastructure development.

3. For Gender and Development (GAD) Advocates, EFLS exemplifies a tool that supports inclusive governance. Its accessible interface and visual outputs enable women, youth, older adults, and marginalized groups to engage in spatial planning processes.

4. For Computational Research, the successful integration of the Krill Herd Lévy Flight (KHL) algorithm into a practical system demonstrates the viability of hybrid metaheuristics for solving real-world spatial optimization problems, contributing to the literature on AI in urban analytics.

Recommendations

1. System Enhancement: Future versions of EFLS should incorporate additional constraint-handling mechanisms, including capacity constraints and dynamic demand. Real-time data integration could further improve its adaptability.
2. Scalability Testing: The system should be tested across larger geographic areas, such as municipalities or provinces, to assess its performance at scale.
3. Stakeholder Training: Capacity-building programs should be implemented to train stakeholders, particularly those in LGUs and civil society, to use EFLS for participatory planning.
4. Policy Integration: EFLS can be recommended for integration into local development plans (LDPs), disaster risk reduction strategies, and public health service mapping.

5. Cross-Sector Applications: The system's architecture can be adapted for use in various domains, including educational infrastructure, emergency response planning, and retail site selection.
6. Academic Deployment: The EFLS may serve as a pedagogical tool in courses involving operations research, geographic information systems (GIS), and urban planning, demonstrating the application of optimization in real-world scenarios.

These recommendations support the continuous improvement, broader application, and sustainability of EFLS in addressing facility location challenges across various sectors and scales.

References

Abualigah, L., Khader, A. T., & Hanandeh, E. S. (2018). A new metaheuristic algorithm for solving optimization problems inspired by the behavior of Krill herds. *Applied Intelligence, 48*(3), 682–699. https://doi.org /10.1007/s10489-017-0992-5

Barceló, J., & Casanovas, E. (1978). A branch and bound algorithm for the uncapacitated plant location problem. *European Journal of Operational Research*, *2*(1), 176-181.

Erlenkotter, D. (1978). A dual-based procedure for uncapacitated facility location. *Operations Research, 26*(6), 992–1009. https://doi.org/10.1287/opre.26.6.992

Galvão, R. D. (1981). A note of Garfinkel, Neebo, and Rao's LP decomposition for the p-median problem. *Transportation Science. 15*(3), 175-293.

Galvão, R. D., & Raggi, L. A. (1989). A method for solving to optimality uncapacitated location problems. *Annals of Operations Research, 18*(1), 225–244. https://doi.org/10.1007/bf02097805

Galvão, R. D. (2004). Uncapacitated facility location problem: Contributions. *Contributions. Pesquisa Operacional, 24*(1), 7–38.

Galvão, R. D., Acosta Espejo, L. G., & Boffey, B. (2002). A hierarchical model for the location of perinatal facilities in the municipality of Rio de Janeiro. *European Journal of Operational Research, 138*(3), 495–517. https://doi.org/10.1016/s0377-2217(01)00172-2.

Gandomi, A. H., & Alavi, A. H. (2012). Krill herd: A new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation, 17*(12), 4831–4845. https: //doi.org/10.1016/j.cnsns.2012.05.010

Guo, Y., Xue, J., Gao, H., & Huang, J. (2020). Improved Krill Herd algorithm with Levy flight and elitism scheme for global optimization problems. *Applied Soft Computing, 86*, Article 105911. https://doi.org/10.1016 /j.asoc.2019.105911

Hakimi, S. L. (1964). Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations Research, 12*(3), 450–459. https://doi.org/10.1287/opre.12.3.450

Hakimi, S. L. (1965). Optimum distribution of switching centers in a communications network and some related graph theoretic problems. *Operations Research, 13*(3), 462–475. https://doi.org/10.1287/opre.13. 3.462

Kuehn, A. A., & Hamburger, M. J. (1963). A heuristic program for locating warehouses. *Management Science, 9*(4), 643–666. https://doi.org/10.1287/mnsc.9.4.643

Liu, Y., & Li, Y. (2024). Finding the optimal fire station locations in the Manhattan District. *Communications in Humanities Research, 48*, 137-146. https://doi.org/10.54254/2753-7064/2024.18757

Mesa, A. P., Castromayor, K. B., Garillos-Manliguez, C. P., & Calag, V. J. (2018). Cuckoo search via Lévy flights applied to the uncapacitated facility location problem. *Journal of Industrial Engineering International, 14*(3), 585–592. https://doi.org/10.1007/s40092-017-0248-0

Snyder, L. V., & Shen, Z.-J. M. (2019). *Fundamentals of supply chain theory* (2nd ed.). Wiley.

Tehrani, S., Jalali-Naini, S., & Shahrabi, J. (2022). Portfolio optimization using an improved krill herd algorithm considering risk and transaction costs. *Expert Systems with Applications, 186*, Article 115746. https://doi.org/10.1016/j.eswa.2021.115746

Van Roy, T. J. (1986). A cross decomposition algorithm for capacitated facility location. *Mathematical Programming, 36*(3), 353–381. https://doi.org/10.1007/BF02592089

Yang, X.-S. (2014). *Nature-inspired optimization algorithms*. Elsevier.

Yang, X.-S., & Deb, S. (2009). Cuckoo search via Lévy flights. In *2009 world congress on Nature & Biologically Inspired Computing (NaBIC)* (pp. 210–214). IEEE. https://doi.org/10.1109/NABIC.2009. 5393690

Zhu, X., Zhang, Y., Zhang, S., & Wu, J. (2018). *GP-CNAS: Convolutional neural network architecture search with genetic programming*. arXiv. https://gpbib.cs.ucl.ac.uk/gp-html/Zhu_2018_arxiv.html