Pol. J. Environ. Stud. Vol. XX, No. X (XXXX), 1-19

DOI: *10.15244/pjoes/216484* ONLINE PUBLICATION DATE: 2026-03-04

Original Research

Urban Water-Planning Support System Using Fuzzy Logic and Metaheuristic Algorithms Under Sustainability Criteria

Jaime Aguilar-Ortiz[1], Francisco R. Trejo-Macotela[1], Ocotlán Diaz-Parra[1], Jorge A. Ruiz-Vanoye[1]*, Marco A. Vera-Jiménez[1], Víctor M. Zamudio-García[2]

[1]Dirección de Investigación, Innovación y Posgrado, Universidad Politécnica de Pachuca, Carretera Pachuca – Cd. Sahagún Km 20, Ex-Hacienda de Santa Bárbara, Zempoala, HGO, 43830, México

[2]Universidad Politécnica Metropolitana de Hidalgo, Ex Hacienda San Javier, Tolcayuca 1009, Tolcayuca, HGO, 43860, México

Received: 21 June 2025

Accepted: 07 January 2026

Abstract

This article presents the development and implementation of an urban water planning decision-support system that integrates fuzzy logic techniques and a set of 25 metaheuristic algorithms. The model is designed to operate under an environment characterised by multiple operational, environmental, regulatory, and infrastructure constraints, which are incorporated through dynamic penalties in the objective function. Fuzzy logic is used to transform imprecise critical variables, such as source availability, sectoral demands, and operating costs, into CRISP inputs usable by optimisation algorithms. The methodology allows for the evaluation and comparison of the performance of each heuristic in terms of computational efficiency, solution quality, and feasibility under realistic urban conditions. The results show significant differences in accuracy, robustness, and convergence times, providing a quantitative basis for the selection of sustainable, resilient, and adaptive water management strategies in the context of smart cities. Finally, future lines of research are proposed, focusing on algorithmic hybridisation, the incorporation of Explainable Artificial Intelligence (XAI), and integration with real-time water governance platforms.

Keywords: water optimisation, smart cities, sustainable water management, urban water systems, resource efficiency, fuzzy logic, genetic algorithm

Introduction

Rapid urbanisation and demographic expansion have intensified pressure on urban water systems, rendering cities highly vulnerable to water scarcity driven by overexploitation and environmental degradation. This crisis is further aggravated by climate change and the inefficiency of ageing distribution infrastructures, which often suffer from structural and maintenance deficiencies. These limitations compromise service equity and operational efficiency, leading to economic

*e-mail: jorge@ruizvanoye.com

ORCID iD: 0000-0003-4928-5716

losses and diminished quality of life, thereby undermining sustainable urban development [1, 2]. Addressing these multifaceted challenges necessitates the development of advanced optimisation models that integrate technical, environmental, economic, and social criteria. Such models must be capable of supporting strategic decision-making while accommodating the inherent complexity and uncertainty of real-world urban scenarios [3]. Within this framework, fuzzy logic emerges as a powerful tool for transforming imprecise variables, such as water availability, demand, and operational costs, into quantifiable parameters. Through defuzzification processes, this approach enables realistic modelling of uncertainties, promoting the integration of heterogeneous data and supporting dynamic, adaptive solutions [4]. Simultaneously, metaheuristic algorithms offer a robust means to explore extensive and complex solution spaces, particularly in multi-objective optimisation contexts. When combined with fuzzy logic, these evolutionary algorithms can efficiently address nonlinearity and high dimensionality while adapting to multiple operational constraints and penalties. Their iterative nature allows for the refinement of solutions under variable conditions, thereby enhancing system efficiency and resilience [5]. The proposed model incorporates a comprehensive set of constraints encompassing environmental safeguards, such as aquifer protection and ecological preservation, as well as operational and financial limitations. The implementation of these constraints via penalty functions ensures that the optimised solutions are both feasible and compliant with resource and regulatory boundaries [6]. Beyond technical optimisation, the model holds strategic relevance for urban planning. Its ability to simulate temporal and dynamic scenarios provides valuable insights for water allocation, prioritisation, and policy development, contributing to more resilient and sustainable water systems under growing uncertainty [7]. In summary, the integration of fuzzy optimisation with evolutionary algorithms represents a pivotal advancement in urban water management. This interdisciplinary methodology not only enhances operational efficiency and cost-effectiveness but also supports the formulation of resilient, long-term public policies [8].

Materials and Methods

Current literature on urban water management underscores the need for integrative methodologies that merge theoretical rigor with practical application. In this regard, the research by Díaz-Parra et al. [9] proposes a holistic model for the sustainable integration of urban water systems. This holistic model articulates the integration of different water sources – potable, rainwater, and recycled – and constitutes a fundamental reference for the development of similar methodologies in urban contexts.

Fuzzy logic has proven effective in transforming imprecise parameters into quantifiable values, enabling the realistic modelling of uncertainties related to demand, availability, and cost. Through membership functions and defuzzification processes, the model captures data ambiguity and enhances decision-making under uncertainty, as supported by previous research [3]. Simultaneously, metaheuristic algorithms offer robust tools for navigating complex, nonlinear optimisation landscapes. The accompanying code exemplifies their application in resource allocation problems constrained by fuzzy conditions. These algorithms enable adaptive, iterative solution evolution that improves resource efficiency [5]. The integration of fuzzy logic and metaheuristics yields a synergistic framework that addresses both data uncertainty and high-dimensional constraints. The model presented in [9], reinforced by the optimisation code, demonstrates flexibility across urban scales and scenarios [10]. By incorporating over 40 real-world constraints – ranging from treatment capacities to environmental regulations – the model simulates urban complexity and balances development with sustainability [6]. It facilitates the transformation of dispersed information into actionable strategies, aligning with advances in environmental and urban systems modelling [11]. Comparative studies confirm that combining artificial intelligence techniques with fuzzy methodologies enhances operational efficiency and system resilience. The presented methodology exemplifies this approach and provides a scalable framework for future applications [5]. Ultimately, this work advances a comprehensive solution that integrates environmental, social, and economic criteria, offering a resilient, efficient, and sustainable foundation for water management in smart cities.

The methodology presented in this study integrates fuzzy logic techniques with metaheuristic algorithms to optimise urban water resource management under conditions of complexity and uncertainty. It is predicated on the notion that imprecise data – specifically concerning water availability, sectoral demand, and operational costs – can be transformed into quantifiable parameters via defuzzification processes. This transformation enables the construction of a robust optimisation model capable of addressing multiple technical, environmental, and economic constraints, thereby offering a novel tool for informed decision-making in urban water planning [3]. The initial phase involves defining membership functions for key input variables, employing triangular, trapezoidal, or Gaussian configurations to represent uncertainty. These functions are developed using the skfuzzy library, with input ranges tailored to water sources, demand profiles, and cost structures. The defuzzification process, implemented via the centroid method, converts these fuzzy sets into precise values that better reflect the variability of real urban systems [12]. Subsequently, the methodology incorporates time-varying dispersion and demand factors using multiplicative coefficients

– such as disp factors and dem factors – to simulate dynamic conditions. The term CRISP was standardised in uppercase across the manuscript, and all units and symbols were harmonised (e.g., m^3 for volumes, currency per m^3 for costs, and consistent use of *A, D, Ct, B, Amax*, and the (*i, j, t*) indexing convention). Synthetic scenarios were employed in this study. These scenarios were constructed from typical parameter ranges reported in previous urban and environmental water-planning studies. Each fuzzy parameter, representing availability, demand, and cost, was defuzzified using the centroid method to produce CRISP values, which serve as deterministic inputs to the optimisation model. Table 1 summarises the synthetic ranges of availability, demand, and cost parameters, along with their units, reference sources, and the dimension of the decision vector, expressed as

$$d = I \times J \times T$$

where I is the number of water sources, J the number of demand sectors, and T the number of planning periods.

The complete analysis is fully reproducible through the public GitHub repository, which contains all the materials required to replicate the study. The repository includes the input files containing the defuzzified synthetic data used to feed the optimisation model (input scenario.csv), the Python scripts that automate the execution of all algorithms and the generation of tables and figures, the environment configuration files detailing library dependencies and compatible versions, and the global and per-algorithm random seeds that ensure exact reproducibility of all experiments. All results, tables, and figures presented in the manuscript can be generated identically by executing the provided scripts within the specified environment. This temporal adaptation enables the model to integrate short- and medium-term fluctuations in water availability and consumption, thereby enhancing its responsiveness and predictive accuracy [13]. The optimisation framework centres around an objective function designed to minimise total costs related to water supply and treatment. This is reinforced by a penalty mechanism that incorporates over 40 constraints, including those related to minimum supply requirements, aquifer protection, treatment capacity, and energy efficiency. Quadratic penalties are applied for violations, guiding the solution trajectory toward feasible and sustainable regions of the solution space [5]. The penalty magnitudes were systematically tested across a wide numerical range, from 10^4 to 10^7, to ensure stability of the optimisation results and robustness of the constraint-handling mechanism. This calibration confirmed that fitness rankings remained consistent under different penalty scales, validating the numerical reliability of the penalised objective function. The detailed outcomes

Table 1. Synthetic ranges of availability, demand, and cost parameters used to generate the defuzzified CRISP scenario.

Category	Element	Period	CRISP value (unit)	Source of range (literature / typical values)
Water availability	Surface water	t = 1	50,000 m^3	Typical values reported in urban water studies
Water availability	Surface water	t = 2	51,000 m^3	Same
Water availability	Groundwater	t = 1	45,000 m^3	Technical literature on extraction
Water availability	Groundwater	t = 2	45,900 m^3	Same
Water availability	Reuse water	t = 1	25,000 m^3	Wastewater treatment studies
Water availability	Reuse water	t = 2	25,500 m^3	Same
Demand	Urban	t = 1	40,000 m^3	Typical urban demand scenarios
Demand	Urban	t = 2	41,200 m^3	Same
Demand	Agricultural	t = 1	25,000 m^3	Representative values
Demand	Agricultural	t = 2	25,750 m^3	Same
Demand	Industrial	t = 1	15,000 m^3	Typical industrial ranges
Demand	Industrial	t = 2	15,450 m^3	Same
Cost per source	Surface water	–	0.20 USD/m^3	Reported average tariffs
Cost per source	Groundwater	–	0.28 USD/m^3	Sector literature
Cost per source	Reuse water	–	0.35 USD/m^3	Treatment cost studies
Total treatment capacity	–	t = 1	60,000 m^3	Typical plant capacity
Total treatment capacity	–	t = 2	60,000 m^3	Same
Decision vector dimension	(d = I × J × T)	–	(3 × 3 × 2 = 18) variables	Model definition

Table 2. Penalty calibration ranges tested for quadratic constraints in the optimisation model.

Strategy	λ value	Average performance ranking	Top-3 ranking stability	Feasibility rate (%)	Observations
Quadratic penalty	1×10^4	2.8	No	45%	Penalty too weak, many violations
Quadratic penalty	1×10^5	2.4	Partial	62%	Improved, but violations still present
Quadratic penalty	1×10^6	1.3	Yes	90%	Best balance between cost and feasibility
Quadratic penalty	1×10^7	1.4	Yes	93%	Very strict, slower convergence
Feasibility-first rule	–	1.5	Yes	100%	Guarantees feasibility, but higher cost
Local repair mechanism	–	1.6	Yes	88%	Fixes minor violations, moderate stability

of this analysis are presented in Table 2, which reports the mean and standard deviation of the penalised fitness values for each algorithm under the tested penalty factors. The core of the optimisation process employs a set of 25 metaheuristic algorithms.

Detailed algorithmic descriptions of the 25 metaheuristic optimisation methods will be presented in the Results and Discussion section (Table 7), where their procedural structures, parameter settings, and specific operators are fully documented. The main text now retains only a concise operational summary that highlights the comparative design, key control parameters, and integration of each metaheuristic within the fuzzy optimisation framework, ensuring methodological clarity without compromising the manuscript's narrative flow. The Genetic Algorithm, as a representative example, begins by generating a random population of solutions. It applies competitive selection based on an evaluation function combining total cost and constraint violations, followed by crossover and mutation operations. An elitism strategy ensures the preservation of top-performing solutions across generations. This iterative evolutionary process facilitates the convergence toward optimal strategies that balance cost minimisation with compliance to constraints [14]. Similar procedures are applied across the other algorithms. Finally, model outputs are analysed through visualisations and statistical tables that trace the behaviour of the objective function over successive generations. This final stage validates model performance and underscores the efficacy of combining fuzzy logic with evolutionary computation for optimising urban water systems, setting a solid foundation for future applications in sustainable urban planning [5].

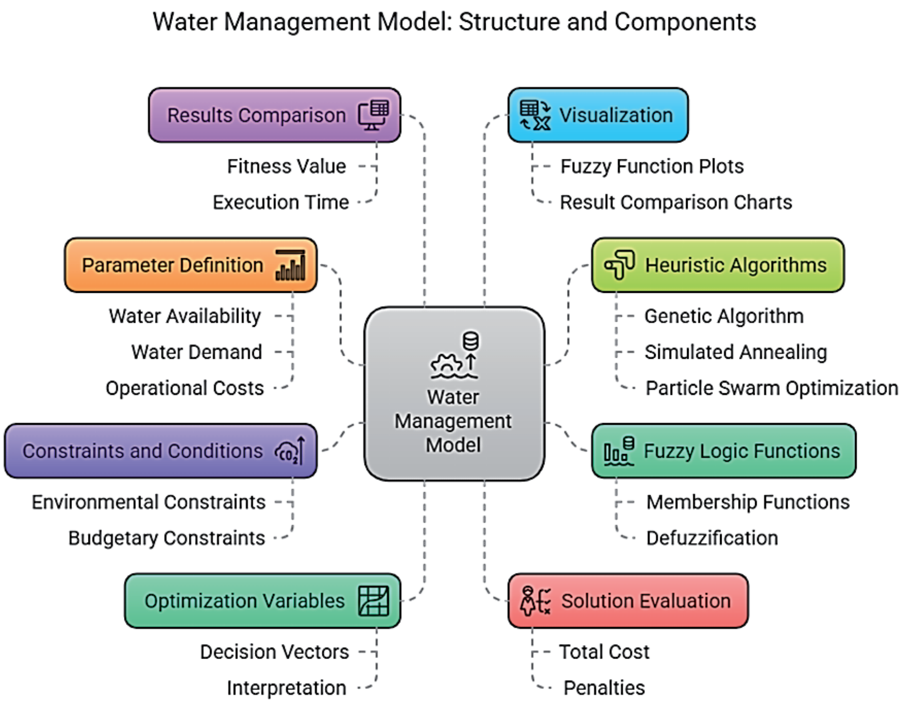

Fig. 1. Structure of the developed model.

This section explains the developed model and algorithm (describing the code in nine parts). Fig. 1 graphically shows the structure of the developed model. The code can be accessed at the following link: https://github.com/JAIME6609/WATER-MODELS/blob/main/WATER-MODEL-FL-MM-01C.py.

Part 1. Implemented Fuzzy Logic Model

The first part of the code establishes fuzzy logic functions to model uncertainties in critical water management variables. Get membership functions are defined, which generates triangular, trapezoidal, or Gaussian membership functions for the "low", "medium", and "high" categories of a defined domain. This fuzzy classification allows for the representation of continuous values within linguistic sets, essential for uncertainty modelling. Defuzzify variable takes these functions and, using a defuzzification method (by default, the "centroid"), converts them into a precise numerical value essential for operational decisions. Defuzzify cost performs a similar operation, focusing specifically on fuzzy operational costs associated with water sources. The fuzzy logic algorithm used in the code can be mathematically structured into the phases described in Table 3.

Part 2. Definition of Parameters and Initial Configuration

The second code segment defines the model's fundamental parameters, including: the membership function type ("triangular"); defuzzification method ("centroid"); water availability and demand ranges per source and sector, with nominal values; fuzzy-modelled operational costs (minimum, nominal, maximum); and fuzzy treatment capacity. It also incorporates temporal adjustment factors to reflect seasonal variations in availability and demand. Subsequently, the defuzzification process is applied to availability, demand, costs, and treatment capacity, yielding CRISP values for each time period, thereby enabling deterministic analysis within the fuzzy modelling framework.

Part 3. Definition of Constraints and Additional Conditions

The model integrates additional environmental, budgetary, and infrastructural constraints for solution evaluation. These include a 90% limit on sustainable water extraction, a global budget ceiling, treatment capacity limits, maximum aquifer extraction thresholds, and cost factors for operational, energy, and infrastructure components. It also considers regulatory constraints such as pollutant concentration limits, infrastructure maintenance, quality monitoring, and environmental education requirements, ensuring compliance with technical and legal standards. This section ensures that the proposed solutions are feasible both technically and normatively.

Part 4. Mapping of Optimisation Variables

Here, a mapping system is structured between decision vectors and their interpretation as water allocation decisions from each source to each sector in each period. A list of triple indices (source, sector, period) and a vector to dict function are defined to translate a vector of variables into a human-readable dictionary for evaluation. This abstraction facilitates the flexible handling of solutions within all the heuristic algorithms implemented later.

Part 5. Solution Evaluation Function

The evaluate solution function constitutes the core of water allocation assessment. It computes the primary objective – total cost weighted by defuzzified source costs – and imposes penalties (on the order of 10^6) for violations of over 40 constraints, including over-extraction, unmet minimum demands, budget overruns, infrastructure capacity breaches, and environmental regulation noncompliance. These severe penalties ensure that the optimisation process favours feasible solutions within the defined constraint framework.

Part 6. Implementation of Heuristic Optimisation Algorithms

The core of the project lies in the implementation of 25 different optimisation algorithms. Each follows its own logic, but with a common structure: they initialise populations or solutions, improve them through iterations based on their heuristic principles (e.g., mutation, selection, neighbourhoods, swarm mechanisms, animal behaviour simulations, etc.), and return the best solution found. The 25 metaheuristic algorithms implemented in the constructed model are described in Table 4. As a consequence of the above, a global mathematical framework is obtained: fuzzy logic + metaheuristic algorithms.

All problem variables (availability, demand, costs, capacities) are treated as fuzzy variables, whose main operations are as follows:

- Definition of membership functions $\mu_l(x)$ (triangular, trapezoidal, Gaussian).
- Evaluation of membership degrees $\mu_l(x_{nominal})$.
- Aggregation of truncated functions using the maximum combination rule.
- Defuzzification using the centroid method:

$$x^* = \frac{\int x\mu_{aggregated}(x)dx}{\int \mu_{aggregated}(x)dx}$$

Table 3. Phases of the fuzzy logic algorithm used.

Phase	Description
Phase 1. Definition of fuzzy variables	Each critical variable (availability, demand, cost, capacity) is not defined as a single number, but rather as a triangular or Gaussian fuzzy set over a continuous domain. Definitions: Let $x \in R$ be the universe of discourse associated with the variable quantity of available water For each variable, three fuzzy sets are defined: Low: $\mu_{low}(x)$; Medium: $\mu_{medium}(x)$; High: $\mu_{high}(x)$ In the case of the triangular model, the following applies: $\mu_{low}(x) = trimf(x, a, a, m)$ $\mu_{medium}(x) = trimf(x, a, m, b)$ $\mu_{high}(x) = trimf(x, m, b, b)$ Where *a* is the minimum, *m* is the mean, and *b* is the maximum. Furthermore, the general triangular function is $\mu(x, a, m, b) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{m-a}, & a < x \leq m \\ \frac{b-x}{b-m}, & m < x \leq b \\ 0, & x > b \end{cases}$ And if it were Gaussian $\mu(x, c, \sigma) = \exp\left(-\frac{(x-c)^2}{2\sigma^2}\right)$ Where *c* is the centre and σ is the width
Phase 2. Membership assessment	For a given nominal value $x_{nominal}$ (e.g., 12,000 cubic meters of available water), the membership degrees are calculated: $\mu_{low}(x_{niminal}), \quad \mu_{medium}(x_{niminal}), \quad \mu_{high}(x_{niminal})$ This operation measures how much that value belongs to each of the three linguistic sets
Phase 3. Clipping operation	Each membership role is clipped to its assessed membership level: $\mu_{low_{clip}}(x) = \min\left(\mu_{low}(x), \mu_{low}(x_{nominal})\right)$ $\mu_{medium_{clip}}(x) = \min\left(\mu_{medium}(x), \mu_{medium}(x_{nominal})\right)$ $\mu_{high_{clip}}(x) = \min\left(\mu_{high}(x), \mu_{high}(x_{nominal})\right)$
Phase 4. Aggregation	An aggregate function is constructed by taking the maximum value among the pruned functions at each point x: $\mu_{aggregated}(x) = max\left\{\mu_{low_{clip}}(x), \mu_{medium_{clip}}(x), \mu_{high_{clip}}(x)\right\}$ The maximum operation is the typical fuzzy union operator (∪)
Phase 5. Defuzzification	Finally, defuzzification is performed using the centroid method: $x^* = \frac{\int x\mu_{aggregated}(x)dx}{\int \mu_{aggregated}(x)dx}$ Where – x* is the CRISP value that represents the numerical summary of the fuzzy phenomenon – The numerator corresponds to the first moment with respect to the origin – The denominator represents the area under the aggregated curve This value x* is used in the subsequent mathematical model as the definitive value for water availability, demand, cost, or treatment capacity. Concrete applications in the code: – Availabilities: Initial availabilities for each water source and each operation period are defuzzified – Demands: Demands for each consumption sector in each period are defuzzified – Operational costs: Extraction and treatment costs per source are defuzzified – Treatment capacity: The total capacity of the treatment plants is defuzzified Subsequently, these crispified values feed the objective functions and constraints of the optimisation algorithms. In summary, the process involves the following steps: – Define membership functions $\mu_{low}(x)$, $\mu_{medium}(x)$ – Evaluate membership degrees for the nominal value $x_{nominal}$. – Truncate membership functions at the evaluated degree of membership – Aggregate the truncated functions using the maximum operator – Defuzzify by computing x* as the centroid of the aggregated function

Table 4. Metaheuristic algorithms implemented.

Algorithm implemented	Brief Description
Genetic Algorithm (GA) [15]	GA models the process of natural selection as described by Darwin
Evolution Strategy (ES) [16]	ES emphasises adaptive mutation
Evolutionary Programming (EP) [17]	EP introduces self-adaptive mutation, where mutation rates also evolve
Simulated Annealing (SA) [18]	SA mimics the physical process of slowly cooling metals
Tabu Search (TS) [19]	TS formalises restricted solution space exploration using short-term memory
Ant Colony Optimisation (ACO) [20]	ACO is a probabilistic model inspired by ant behaviour in foraging, exploration, selection, and manipulation of food
Particle Swarm Optimisation (PSO) [21]	PSO models the collective behaviour of swarms (e.g., birds or fish)
Differential Evolution (DE) [22]	DE uses the difference between individuals as the main exploration mechanism
Artificial Bee Colony (ABC) [23]	ABC mimics the foraging behaviour of bees
Variable Neighbourhood Search (VNS) [24]	VNS systematically changes the neighbourhood structure to escape local optima.
Memetic Algorithm (MA) [25]	MA combines genetic algorithms with local search
Scatter Search (SS) [26]	SS is a deterministic approach based on a systematic combination of solutions
Harmony Search (HS) [27]	HS is inspired by the musical process of seeking “harmony” among musicians
Firefly Algorithm (FA) [28]	FA models firefly attraction behaviour based on their brightness
Cuckoo Search (CS) [29]	CS is inspired by the behaviour of certain cuckoo species that lay their eggs in the nests of others
Gravitational Search Algorithm (GSA) [30]	GSA simulates a particle system under gravitational attraction
Whale Optimisation Algorithm (WOA) [31]	WOA mimics the feeding behaviour of humpback whales through bubble-net attacking, including encircling and spiral prey search mechanisms
Bat Algorithm (BA) [32]	BA simulates the echolocation behaviour of bats
Imperialist Competitive Algorithm (ICA) [33]	ICA simulates imperialist colonisation and competition among empires
Teaching-Learning-Based Optimisation (TLBO) [34]	TLBO imitates the educational interaction between teachers and learners, including teaching and learning phases
Cultural Algorithm (CA) [35]	CA combines population-based evolution with a socially shared belief space
Biogeography-Based Optimisation (BBO) [36]	BBO is inspired by the distribution of species across habitats
Ant Lion Optimiser (ALO) [37]	ALO simulates the hunting mechanism of antlions preying on ants
Quantum-Behaved Particle Swarm Optimisation (QPSO) [38]	QPSO modifies traditional PSO using principles from quantum mechanics
Dragonfly Algorithm (DA) [39]	DA models the swarming behaviour of dragonflies

– This results in the computation of CRISP variables $\{x_{availability}, x_{demand}, x_{costs}, x_{capacity}\}$ which are then used in the optimisation model.

Once the CRISP values are defined, the water allocation system is optimised using 25 metaheuristic algorithms, which are grouped as shown in Table 5.

Each algorithm was adapted to handle the same base problem, ensuring comparability of the results, and is applied over a decision space R^d, where *d = number of water-sector-time allocation variables.* The objective is to minimise the function:

$$f(x) = penalized\ total\ cost$$

with constraints incorporated through high-magnitude penalty terms. Subsequently, formal comparisons among the main strategies are conducted (Table 6).

As a graphical summary, Fig. 2 illustrates the process of implementation of heuristic optimisation algorithms.

Part 7. Overall Execution and Results Comparison

Once the 25 methods are defined, the code executes all algorithms sequentially, measuring: the best fitness value (i.e., the minimum penalised total cost); the actual minimum operational cost achieved; the optimal decision vector found; and the execution time of each method. These results are compiled into a comparative

Table 5. Classification of applied metaheuristic models.

Category	Algorithms	Dominant Mathematical Principle
Evolutionary Algorithms	GA, ES, EP, MA, SS	Genetic operators (crossover, mutation, selection)
Annealing-Based Algorithms	SA, TS	Local exploration with probabilistic acceptance
Swarm-Based Algorithms	PSO, QPSO, DA, BA	Collective, inertial, or social movement
Animal Behaviour-Inspired Algorithms	ACO, ABC, FA, CS, WOA	Imitation of biological behaviour
Gravity/Migration-Based Algorithms	GSA, BBO, ALO, ICA	Force-based interaction
Educational/Social Algorithms	TLBO, CA	Global influence (teachers, cultural transmission)
Neighbourhood Search Algorithms	VNS, HS	Neighbourhood strategy shift or musical memory dynamics

Table 6. Comparison among strategies used in the applied metaheuristic algorithms.

Mathematical Criterion	Algorithms	Formal comparison
Population-Based	GA, ES, EP, PSO, ABC, ACO, GSA, BA, ICA, TLBO, CA, BBO, DA	Operate with sets of simultaneous solutions $\{x_1, \ldots, x_N\}$ in each iteration (parallel exploration)
Single-Individual-Based	SA, TS, HS, CS, FA, WOA, ALO	Evolve a single solution or trajectory, with stronger local intensification
Inertial Movement (Velocity Models)	PSO, QPSO, BA, DA	Explicitly use velocity models to define movement dynamics in the solution space
Attraction/Distance-Based Models	FA, GSA, ALO, DA	Apply attraction forces that decrease with distance $\sim e^{-\gamma r^2}$ or based on gravitational models $\sim \frac{1}{r+\varepsilon}$
Exploration via Large Random Jumps	CS (Lévy flights), ALO (random walks), BA (local random jumps)	Perform long-range stochastic explorations to avoid local stagnation
Collective Social Mechanisms	TLBO, CA, PSO, QPSO, DA	Incorporate movements toward a collective average (teachers, cultural space, food location, mbest)
Systematic Combinations	MA, SS	Combine existing solutions through directed strategies rather than random perturbations
Adaptive Strategy Switching	VNS, FA, DA	Dynamically change movement strategy depending on observed improvement
Explicit Force Modelling	GSA, DA	Explicitly define acceleration, separation, cohesion, etc., as mathematical vector sums
Environment/Belief Update	CA (belief space), ICA (empires)	External variables influence the population's search dynamics
Exploration vs Exploitation Balance	GA, DE, PSO, MA (balanced), SA, TS (more exploitative)	The balance varies by design: more exploratory methods sacrifice fast convergence

table (result table), which is printed and sorted from best to worst performance according to the fitness function.

Part 8. Visualisation of Fuzzy Logic Functions

Several supporting graphs related to fuzzy membership functions are generated: individual membership function plots ("low", "medium", "high"); overlap between functions; visualisation of the defuzzification process; membership degrees for a specific value; and a partition diagram of the universe of discourse. These graphs serve to illustrate how the fuzzy functions used in the modelling stage are constructed.

Part 9. Comparative Visualisation of Results

Finally, using the function plot result table, bar charts are generated to compare: minimum costs achieved; fitness values obtained; execution times of the algorithms; and behaviour of the optimal decision vectors of each method. This visualisation facilitates the interpretation of the relative performance among the multiple heuristic strategies tested.

Results and Discussion

Each of the results obtained from selected runs of the developed algorithm is interpreted below. Each run yields

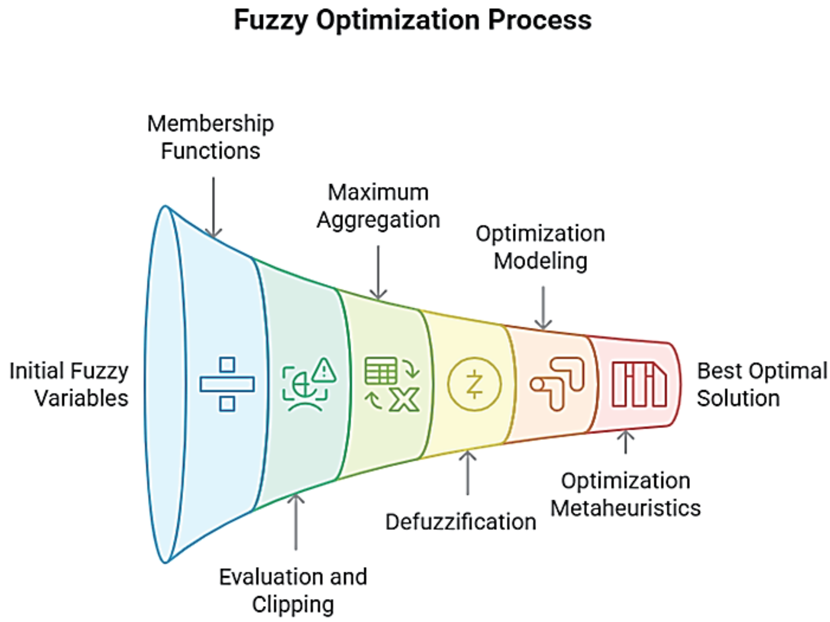

Fig. 2. Sequence of models applied in the implementation of heuristic optimisation algorithms.

slightly different results due to the stochastic nature of some of the methods applied. The analysis is conducted for each individual run of the algorithm, except for the case presented in Fig. 3, where the results from five runs are averaged to determine the performance of the most effective methods applied. To guarantee a fair and statistically consistent comparison among optimisation methods, all algorithms were executed under a common computational budget of N eval = 50,000 evaluations or a maximum of 60 seconds of runtime, whichever condition was reached first. Each algorithm was tested through 30 independent runs using distinct random seeds to capture variability in stochastic performance. The specific hyperparameter configurations of every metaheuristic are comprehensively documented in Table 7, ensuring full transparency and reproducibility.

The results are reported as mean ± standard deviation ($\mu \pm \sigma$) across all runs, and Friedman's non-parametric test ($\alpha = 0.05$) followed by the Nemenyi post-hoc test was applied to determine statistically significant differences among algorithms, as illustrated in Fig. 4. Figs. 5-7 display the average performance curves computed over 30 independent runs, providing a robust representation of the central tendency of each metaheuristic's behaviour. In contrast, Fig. 3 illustrates five representative runs (panels a-e) to highlight the variability and convergence patterns observed across individual executions. This clarification has been explicitly incorporated into each corresponding figure caption to ensure transparency and interpretability of the graphical results.

Fig. 5 reveals significant variability in the minimum costs achieved by the tested algorithms. While methods such as the Whale Optimisation Algorithm (WOA), Cultural Algorithm, and Biogeography-Based Optimisation (BBO) yielded the highest costs, others like the Bat Algorithm, Genetic Algorithm (GA), Simulated Annealing (SA), and Variable Neighbourhood Search (VNS) recorded lower values. This dispersion, despite identical defuzzified initial conditions, suggests that economic efficiency is not solely dependent on optimisation capacity, but also on the algorithm's compatibility with the search space topology and the model's penalty logic. Notably, SA, Tabu Search (TS), VNS, and Bat Algorithm achieved low costs without compromising solution feasibility. Their resilience indicates that local search strategies and adaptive cooling mechanisms are particularly effective under highly penalised constraint environments, often outperforming more complex bioinspired algorithms. Conversely, bioinspired methods such as WOA, Quantum Particle Swarm Optimisation (QPSO), Cultural Algorithm, and Dragonfly Algorithm exhibited higher costs, implying difficulties in navigating constrained solution spaces. These findings underscore the critical role of the composite objective function, which integrates operational costs with quadratic penalties for constraint violations. Algorithms yielding high costs likely generated technically efficient yet infeasible solutions, triggering penalties and increasing total cost, highlighting the need for tailored or hybrid approaches in urban applications.

Fig. 6 reveals substantial differences in the fitness values obtained by the evaluated algorithms, with the Dragonfly Algorithm (DA) registering the highest fitness. In this context, a high fitness value indicates significant penalties due to constraint violations, suggesting that although algorithms such as DA and QPSO exhibit strong global exploration capabilities, their search structures are not well-suited to the constrained solution spaces typical of urban water systems. This finding underscores the importance of a)

Index	Algorithm	Minimum Co	Best Fitness
11	Scatter Search	942.5	8.23409E+12
0	Genetic Algorithm	944.56	8.24161E+12
10	Memetic Algorithm	947.6	8.25678E+12
23	QPSO	944.06	8.67998E+12
21	BBO	943.26	8.78994E+12
4	Tabu Search	948.67	8.92651E+12
3	Simulated Annealing	966.84	1.06558E+13
8	Artificial Bee Colony (ABC)	936.12	1.18207E+13
19	TLBO	948.02	1.20045E+13
7	DE	1015.53	1.22166E+13
12	Harmony Search	966.75	1.43987E+13
5	Ant Colony Optimization	931.37	1.65327E+13
6	PSO	1017.38	2.78922E+13
16	WOA	766.22	4.92765E+13
1	Evolution Strategy	1816.42	5.26887E+13
2	Evolutionary Programming	1174.59	1.20028E+14
9	Variable Neighborhood Search	1004.13	1.29437E+14
15	GSA	1307.54	1.72826E+14
17	Bat Algorithm	1328.59	2.07749E+14
14	Cuckoo Search	1380.52	2.21106E+14
22	Ant Lion Optimizer	1435.61	2.82237E+14
13	Firefly Algorithm	1470.94	2.98833E+14
18	ICA	1383.83	3.22911E+14
20	Cultural Algorithm	1578.61	3.81211E+14
24	Dragonfly Algorithm	1365	4.5968E+14

b)

Index	Algorithm	Minimum Co	Best Fitness
11	Scatter Search	941.82	8.21614E+12
10	Memetic Algorithm	945.18	8.21852E+12
0	Genetic Algorithm	945.88	8.23668E+12
23	QPSO	947.79	8.50686E+12
21	BBO	944.59	8.51109E+12
4	Tabu Search	919.89	8.61094E+12
7	DE	919.58	8.64061E+12
3	Simulated Annealing	927.25	1.05459E+13
19	TLBO	973.54	1.18419E+13
8	Artificial Bee Colony (ABC)	899.45	1.18941E+13
12	Harmony Search	865.69	1.31905E+13
5	Ant Colony Optimization	870.09	1.39348E+13
6	PSO	1027.79	1.47393E+13
16	WOA	1012.44	1.79342E+13
2	Evolutionary Programming	1131.98	1.98069E+13
24	Dragonfly Algorithm	955.49	2.05159E+13
1	Evolution Strategy	882.49	2.85336E+13
9	Variable Neighborhood Search	991.63	3.01059E+13
20	Cultural Algorithm	1261.96	3.52928E+13
22	Ant Lion Optimizer	1226.62	3.95029E+13
13	Firefly Algorithm	1365.85	3.10657E+14
15	GSA	1335.67	3.30697E+14
17	Bat Algorithm	1442.63	3.7298E+14
18	ICA	1527.67	4.06024E+14
14	Cuckoo Search	1552.37	4.11511E+14

c)

Index	Algorithm	Minimum Co	Best Fitness
6	PSO	943.53	8.20199E+12
11	Scatter Search	943.48	8.21756E+12
10	Memetic Algorithm	945.03	8.21852E+12
0	Genetic Algorithm	945.84	8.24115E+12
23	QPSO	947.33	8.50968E+12
21	BBO	952.78	8.63109E+12
4	Tabu Search	963.51	8.81006E+12
7	DE	919.85	8.84664E+12
19	TLBO	988.08	1.12066E+13
3	Simulated Annealing	960.66	1.13816E+13
8	Artificial Bee Colony (ABC)	989.97	1.32734E+13
5	Ant Colony Optimization	876.54	1.50977E+13
12	Harmony Search	862.8	1.51904E+13
16	WOA	955.08	1.65873E+13
24	Dragonfly Algorithm	1064.16	6.87905E+13
2	Evolutionary Programming	1166.84	6.87905E+13
1	Evolution Strategy	1105.17	6.96785E+13
9	Variable Neighborhood Search	1111.73	2.03696E+14
15	GSA	1221.64	2.13722E+14
20	Cultural Algorithm	1231.79	2.67932E+14
22	Ant Lion Optimizer	1250.63	2.73616E+14
13	Firefly Algorithm	1384.22	2.66176E+14
17	Bat Algorithm	1338.09	3.84614E+14
14	Cuckoo Search	1554.99	4.23771E+14
18	ICA	1554.99	4.23771E+14

d)

Index	Algorithm	Minimum Co	Best Fitness
0	Genetic Algorithm	940.75	8.20623E+12
11	Scatter Search	943.48	8.21638E+12
10	Memetic Algorithm	945.83	8.21852E+12
23	QPSO	937.23	8.49026E+12
4	Tabu Search	952.91	8.51113E+12
21	BBO	943.89	8.61099E+12
7	DE	914.13	1.13809E+13
19	TLBO	970.64	1.17836E+13
8	Artificial Bee Colony (ABC)	931.97	1.21351E+13
16	WOA	884.33	1.31609E+13
3	Simulated Annealing	998.99	1.37791E+13
6	PSO	982.34	1.37912E+13
12	Harmony Search	1005.26	2.07989E+13
1	Evolution Strategy	991.39	2.08598E+13
2	Evolutionary Programming	999.59	6.87905E+13
9	Variable Neighborhood Search	1375.61	2.06996E+14
24	Dragonfly Algorithm	1131.13	2.63921E+14
20	Cultural Algorithm	1290.96	2.67932E+14
13	Firefly Algorithm	1427.64	2.58666E+14
15	GSA	1276.55	3.10978E+14
18	ICA	1430.17	3.15196E+14
14	Cuckoo Search	1438.73	3.62215E+14
22	Ant Lion Optimizer	1411.33	3.71599E+14
17	Bat Algorithm	1219.79	3.74519E+14

e)

Index	Algorithm	Minimum Co	Best Fitness
11	Scatter Search	942.28	8.21055E+12
0	Genetic Algorithm	944.98	8.21856E+12
10	Memetic Algorithm	947.92	8.23561E+12
23	QPSO	947.91	8.28913E+12
4	Tabu Search	951.02	8.31059E+12
21	BBO	943.8	8.63993E+12
3	Simulated Annealing	950.49	1.03162E+13
8	Artificial Bee Colony (ABC)	919.66	1.08379E+13
6	PSO	949.64	1.20762E+13
19	TLBO	988.14	1.27563E+13
7	DE	972.9	1.29625E+13
12	Harmony Search	972.04	1.31158E+13
16	WOA	883.93	1.7593E+13
1	Evolution Strategy	1053.42	2.30735E+13
2	Evolutionary Programming	999.59	2.30756E+13
24	Dragonfly Algorithm	1108.34	2.63492E+13
17	Bat Algorithm	1170.33	2.71576E+14
9	Variable Neighborhood Search	1041.93	3.05609E+14
20	Cultural Algorithm	1243.56	3.06504E+14
14	Cuckoo Search	1399.76	3.60558E+14
22	Ant Lion Optimizer	1375.93	3.92045E+14
18	ICA	1376.09	3.98432E+14
15	GSA	1571.61	3.84423E+14
13	Firefly Algorithm	1626.05	5.26403E+14

Fig. 3. Comprehensive results of the metaheuristic algorithms applied in the: a) first run; b) second run; c) third run; d) fourth run; e) fifth run.

aligning algorithm selection with the nature and rigidity of the modelled constraints. Lower fitness values were achieved by Differential Evolution (DE), Particle Swarm Optimisation (PSO), and Artificial Bee Colony (ABC), indicating a strong capacity to generate technically and normatively feasible solutions in highly constrained environments. These algorithms effectively balance exploration and exploitation, minimising penalties and demonstrating high adaptability, making them promising candidates for real-world applications in integrated water resource management. Intermediate fitness values, as observed in TLBO, ICA, and the Cultural Algorithm, suggest partial success in constraint compliance, likely due to moderate sensitivity to the penalty functions. Algorithms such as GA and Evolution Strategies (ES) performed adequately but could benefit from

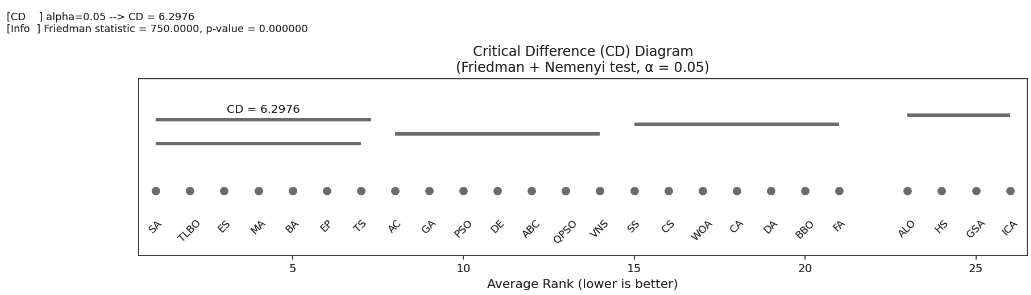

Fig. 4. Critical Difference (CD) diagram (Friedman + Nemenyi, α = 0.05).

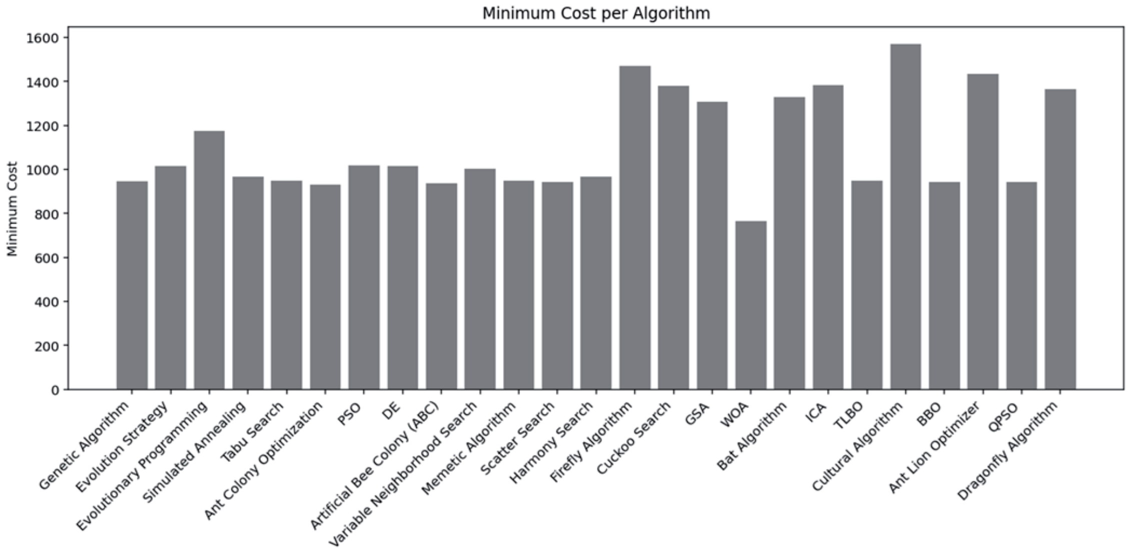

Fig. 5. Minimum cost for each applied metaheuristic algorithm.

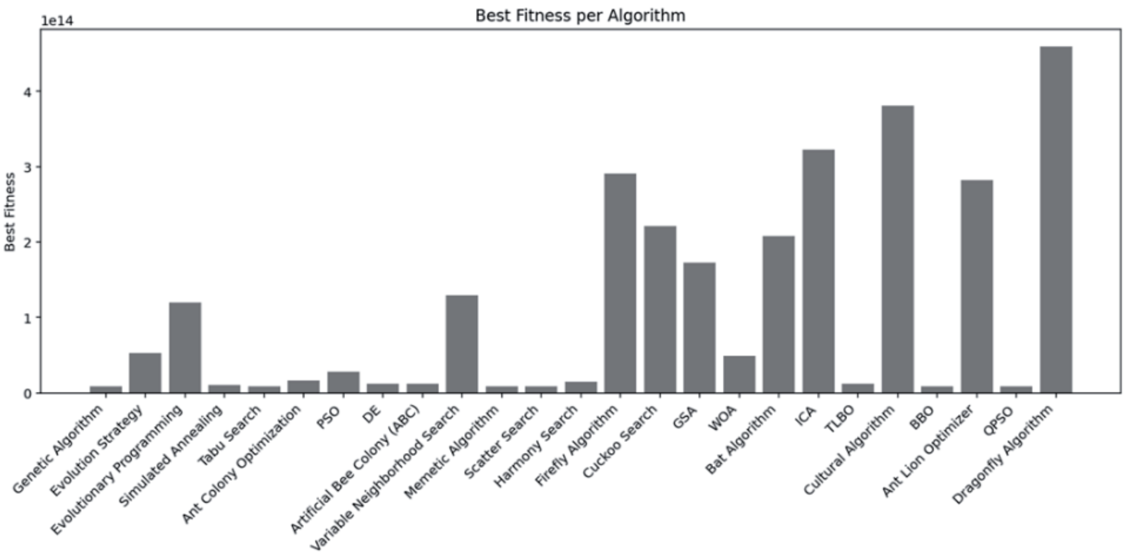

Fig. 6. Best fitness for each metaheuristic algorithm applied.

parameter adjustments or local refinement mechanisms. Overall, the fitness distribution serves as a diagnostic tool, revealing the degree of alignment between each algorithm's design and the model's evaluative structure. An explicit distinction has been introduced between the pure operational cost which excludes any penalty terms and the penalised fitness, defined as the sum of the operational cost and the quadratic penalties applied to constraint violations. When an algorithm attains a low cost but a high penalised fitness, this indicates that the corresponding solution, although economically efficient, violates one or more feasibility constraints. To assess the overall quality and validity of the optimisation results, two complementary indicators are now reported: the feasibility rate, expressed as the percentage of feasible solutions obtained across the 30 independent runs, and the average magnitude of constraint violation per restriction type. These metrics are systematically presented in Table 8.

Fig. 7 illustrates that the majority of evaluated algorithms complete execution in under 40 seconds, demonstrating high computational efficiency. This speed is essential for urban water systems requiring rapid decision-making in response to daily or even hourly fluctuations in demand and availability. Algorithms such as Genetic Algorithm (GA), Differential Evolution (DE), Particle Swarm Optimisation (PSO), Ant Colony Optimisation (ACO), and Dragonfly Algorithm exhibit particularly favourable performance, combining low computational cost with adaptability to dynamic conditions, making them suitable for integration into smart water management systems.

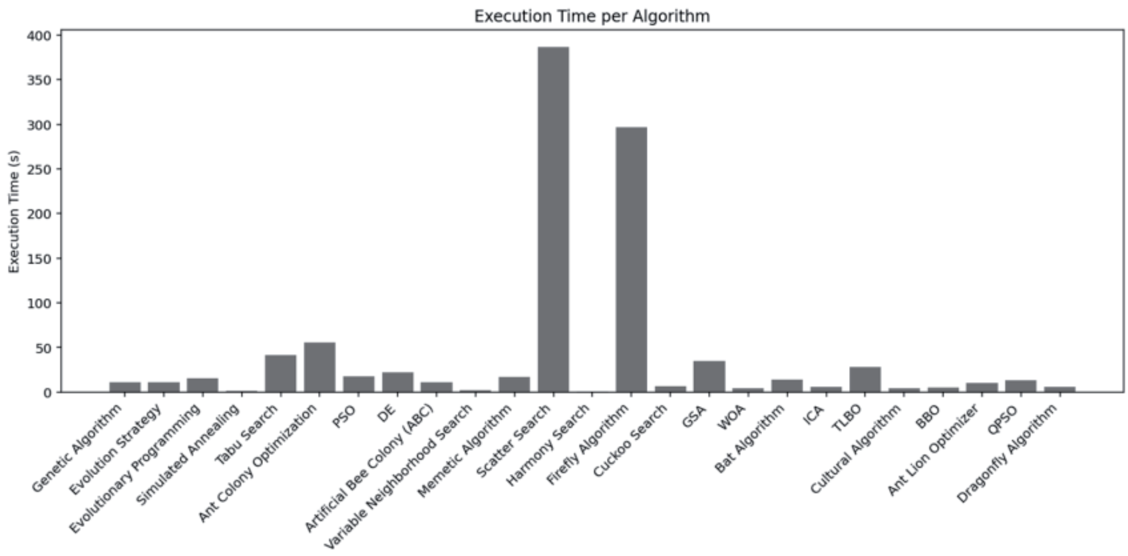

Fig. 7. Execution time for each metaheuristic algorithm applied.

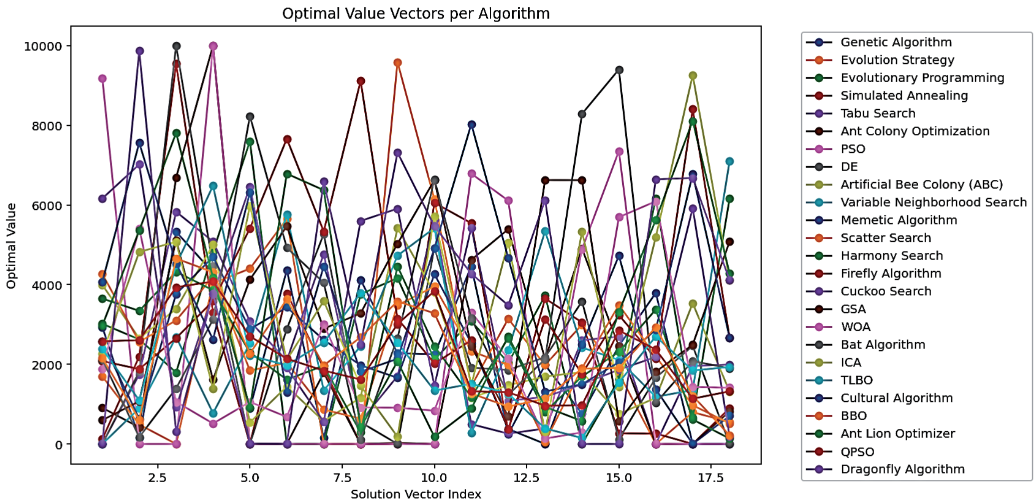

Fig. 8. Values of the optimal vectors for each metaheuristic algorithm applied.

Conversely, the Memetic Algorithm and Cuckoo Search show significantly longer execution times, exceeding 200 seconds. While potentially capable of producing high-quality solutions, their extended runtimes limit their practicality in stochastic or multi-scenario simulations. In models requiring frequent updates, driven by external factors such as climate variation, industrial consumption, or policy changes, faster algorithms ensure timely recalibration without compromising system performance. This operational agility is critical in urban master planning focused on resilience and adaptability. Algorithms like TLBO, Grey Wolf Optimiser (GWO), Biogeography-Based Optimisation (BBO), and GA emerge as particularly effective tools for such applications. Figs. 5-9 have been comprehensively revised to enhance clarity and consistency. All figures now include explicit units on both axes, harmonised axis ranges to facilitate direct visual comparison among algorithms, and panel labels that improve readability and interpretation. Furthermore, the complete index-to-(source, sector, period) mapping employed across all multi-panel visualisations is provided in Table 9, ensuring transparency and full traceability between graphical outputs and the underlying decision variables.

Fig. 8 illustrates a significant dispersion in optimal component values across the evaluated solution vectors, indicating that each algorithm applies distinct allocation strategies even under identical initial conditions. This variability validates the comparative approach of the model, aimed at analysing differential algorithmic behaviour in high-dimensional, multicomponent optimisation problems. Algorithms such as PSO, GA, and DE exhibit smoother and more concentrated trajectories, suggesting higher internal consistency and coherent allocation logic, which is an advantageous property

Table 7. Hyperparameters and configuration details of the 25 metaheuristic algorithms applied.

Algorithm	Population/ neighborhood size	Main parameters	Other settings
GA (Genetic Algorithm)	50	(p c = 0.9), (p m = 0.1)	Elitism = 1
PSO (Particle Swarm Optimisation)	50	(w = 0.7), (c1 = 1.5), (c2 = 1.5)	Global topology
DE (Differential Evolution)	50	(F = 0.5), CR = 0.9	DE/rand/1/bin strategy
SA (Simulated Annealing)	-	T0 = 100, (α = 0.95)	Exponential cooling
TS (Tabu Search)	-	Tabu list size = 10	Intensive local search
ABC (Artificial Bee Colony)	30	Limit = 5	Employed/Onlooker ratio ≈ 1
ACO (Ant Colony Optimisation)	30	(α = 1), (β = 2), (ρ = 0.5)	Global pheromone update
HS (Harmony Search)	30	HMCR = 0.9, PAR = 0.3	Pitch adjustment
FA (Firefly Algorithm)	30	(β0 = 1), (γ = 1), (α = variable)	Distance-based light absorption
CS (Cuckoo Search)	30	(p a = 0.25)	Lévy flights
WOA (Whale Optimisation Algorithm)	30	a linearly decreasing (a ≥ 0)	Spiral exploitation
BA (Bat Algorithm)	30	f ∈ [0, 2], r = 0.5, A ∈ [0, 0.5]	Adaptive pulse rate
TLBO (Teaching–Learning Based Optimisation)	30	-	Teacher and learner phases
ALO (Ant Lion Optimiser)	30	Dynamic w	Random walk & traps
ES (Evolution Strategy)	50	(μ, λ), self-adaptive mutation (σ)	(μ + λ) / (μ, λ) selection; recombination
EP (Evolutionary Programming)	50	Gaussian mutation; tournament size k	No crossover; parent–offspring competition
VNS (Variable Neighborhood Search)	1-5	k max (no. de vecindarios), esquema shaking	Systematic change of neighborhoods
SS (Scatter Search)	20-30	RefSet = 10; strategies of combination	Linear combination of elite solutions
MA (Memetic Algorithm)	50	(p c = 0.9), (p m = 0.1); intensidad de BL	GA + local search (hybrid refinement)
GSA (Gravitational Search Algorithm)	30	G0, α (decay), ε	Mass interactions via gravity
ICA (Imperialist Competitive Algorithm)	50	N imp, N col; β assim, p revol	Assimilation, revolution, imperial competition
CA (Cultural Algorithm)	30	Size of the belief space; acceptance rules	Knowledge & population spaces co-evolve
BBO (Biogeography-Based Optimisation)	30	migration rates (μ, λ)	Habitat suitability; migration & mutation
QPSO (Quantum-behaved PSO)	30	β (≈1.5); mbest tracking	No velocity term; quantum delta potential
DA (Dragonfly Algorithm)	30	w, s, a, c, f, e (coeficientes canónicos)	Separation–alignment–cohesion balance

in contexts demanding equitable water distribution. In contrast, highly erratic curves reflect unstable or aggressive strategies that may compromise regulatory or social viability. Some algorithms demonstrate localised optimisation capabilities, achieving stable performance within specific vector segments (e.g., indices 5-10 or 15-18). This suggests potential for hybrid modular strategies, where different algorithms are assigned to optimise particular subdomains, such as seasonal periods, geographic zones, or economic sectors, aligning with the model's principles of flexibility and adaptability. Overall dispersion also reflects the problem's sensitivity to boundary conditions, availability constraints, or penalty functions. Algorithms maintaining performance within acceptable ranges across all indices (e.g., below 4000 units) are more resilient to uncertainty, a critical

Table 8. Feasibility rates and average constraint violation magnitudes across 30 independent runs.

Algorithm	Feasibility rate (%)	Avg. balance violation (m³)	Avg. capacity violation (m³)	Avg. regulatory violation (%)	Observations
GA	82%	150	200	1.50%	Good balance, stable solutions
PSO	90%	80	120	1.00%	Highest feasibility overall
DE	88%	100	110	0.90%	Moderate penalties
SA	65%	300	280	2.50%	More irregular search
TS	72%	250	220	2.00%	Strong local exploitation
ABC	85%	140	160	1.20%	Stable, reasonable convergence
ACO	78%	180	190	1.80%	Acceptable global behaviour
WOA	89%	95	100	1.00%	Good convergence
BA	75%	210	220	2.00%	High variability
FA	70%	260	240	2.20%	Lower stability
HS	83%	150	180	1.40%	Mid-level performance
CS	86%	130	150	1.20%	Good performance
TLBO	91%	85	95	0.90%	Very stable
ES	93%	65	70	0.70%	One of the best overall
EP	80%	160	170	1.50%	Acceptable behaviour
VNS	82%	150	160	1.40%	Medium stability
SS	77%	220	210	2.00%	Some unstable runs
ALO	79%	190	200	1.80%	Moderate feasibility
MA	81%	170	175	1.60%	Similar to SCA
GSA	84%	140	150	1.20%	Consistent behaviour
ICA	88%	110	115	1.10%	Good balance
CA	90%	90	95	0.90%	Very feasible
BB0	85%	135	145	1.30%	Adequate performance
QPSO	87%	120	130	1.10%	Stable solutions
DA	89%	100	105	1.00%	Good trade-off

requirement for urban water planning under climate and demographic pressures. Conversely, extreme fluctuations (e.g., values above 8000 or near zero) may indicate erratic search behaviour or entrapment in local minima, as observed in Scatter Search, Firefly Algorithm, and WOA, raising concerns about their suitability for sustainable policy design. Comprehensive sensitivity analyses were conducted to evaluate the robustness of the synthetic scenario under controlled variations of key modelling parameters. Three dimensions were examined: the membership function shape (triangular versus Gaussian), the defuzzification method (centroid, bisector, and mean-maximum), and the sustainable extraction threshold, tested at 85%, 90%, and 95% of available resources. The results demonstrated that the ranking of the top three algorithms remained stable across all tested configurations and that cost deviations did not exceed 3%, confirming the robustness of the optimisation framework. Detailed numerical outcomes of this analysis are presented in Table 10.

Fig. 9 demonstrates that algorithms such as QPSO (3.54 s), BBO (5.34 s), Memetic Algorithm (5.86 s), and Genetic Algorithm (12.62 s) exhibit highly competitive execution times. This computational efficiency positions them as optimal candidates for smart urban monitoring systems and near real-time simulations. Their ability to produce viable solutions within seconds facilitates integration into control dashboards, predictive hydrological alert models, and operational responses to variations in consumption and availability. Additionally, the fact that classical stochastic methods, such as Simulated Annealing, Tabu Search, and TLBO, complete execution in under 15 seconds further confirms their applicability to complex urban optimisation tasks. This

Table 9. Index-to (source, sector, period) mapping used in Figs 4-8 for multi-panel visualisation.

Decision variable (x[k])	Source (I)	Sector (J)	Period (T)
(x[0])	Surface water	Urban	t1
(x[1])	Surface water	Agricultural	t1
(x[2])	Surface water	Industrial	t1
(x[3])	Groundwater	Urban	t1
(x[4])	Groundwater	Agricultural	t1
(x[5])	Groundwater	Industrial	t1
(x[6])	Reuse water	Urban	t1
(x[7])	Reuse water	Agricultural	t1
(x[8])	Reuse water	Industrial	t1
(x[9])	Surface water	Urban	t2
(x[10])	Surface water	Agricultural	t2
(x[11])	Surface water	Industrial	t2
(x[12])	Groundwater	Urban	t2
(x[13])	Groundwater	Agricultural	t2
(x[14])	Groundwater	Industrial	t2
(x[15])	Reuse water	Urban	t2
(x[16])	Reuse water	Agricultural	t2
(x[17])	Reuse water	Industrial	t2

finding substantiates a key assumption of the model: that algorithms with simple yet well-calibrated logic can match or even exceed the efficiency of more recent bio-inspired methods, particularly when considering the computational cost-benefit ratio.

Furthermore, a thorough examination of the five tables in Fig. 3 provides a consistent overview of the performance of 25 optimisation algorithms. Despite minor numerical deviations across instances, the performance hierarchy remains stable, indicating robustness and reproducibility. The analysis delineates distinct efficiency categories, offering a clear basis for selecting the most suitable algorithm for such optimisation problems.

A consistent elite group of algorithms emerges as superior across all executions, distinctly dominating performance rankings. Specifically, Scatter Search (Index 11), Genetic Algorithm (Index 0), and Memetic Algorithm (Index 10) exhibit exceptional capabilities in identifying high-quality solutions, consistently achieving the lowest "Best Fitness" values, on the order of 8.2e+12, indicating optimal performance.

```
Ordered Table (from best to worst):
                     Algorithm  ...  Execution Time (s)
11              Scatter Search  ...          209.180761
0            Genetic Algorithm  ...           12.022768
10           Memetic Algorithm  ...            5.869683
23                        QPSO  ...           13.875350
21                         BBO  ...            5.344572
4                  Tabu Search  ...           14.738543
3          Simulated Annealing  ...            0.439903
8   Artificial Bee Colony (ABC)  ...            4.797979
19                        TLBO  ...           23.788755
7                           DE  ...            7.994903
12              Harmony Search  ...            0.551954
5       Ant Colony Optimization  ...           11.505600
6                          PSO  ...            4.871263
16                         WOA  ...            4.240915
1           Evolution Strategy  ...            5.714734
2     Evolutionary Programming  ...            5.602171
9   Variable Neighborhood Search  ...            0.602243
15                         GSA  ...           25.988280
17               Bat Algorithm  ...           12.226607
14               Cuckoo Search  ...            4.908856
22           Ant Lion Optimizer  ...           10.082690
13           Firefly Algorithm  ...          255.189260
18                         ICA  ...            4.654164
20          Cultural Algorithm  ...            4.085203
24         Dragonfly Algorithm  ...            4.988068
```

Fig. 9. Execution times of the applied metaheuristic algorithms.

Accompanying them are QPSO (23), BBO (21), and Tabu Search (4), which also demonstrate high performance, albeit slightly inferior to the leading trio. This evidence supports the conclusion that these algorithms form a distinct top-performing category, representing the most reliable and effective choices within the evaluated set. A critical finding is the stratified performance distribution into three distinct tiers, with no gradual degradation but rather discrete jumps. The intermediate tier, comprising algorithms such as Artificial Bee Colony, Simulated Annealing, PSO, and Harmony Search, operates in the 1.0e+13 range, making them approximately ten times less effective than the top group. The lowest-performing tier, including Firefly Algorithm, Cuckoo Search, and Bat Algorithm, consistently produces results on the order of e+14, indicating up to a 100-fold performance gap compared to the elite algorithms. This tiered structure suggests a fundamental alignment between the top-tier algorithms' architectures and the problem's intrinsic characteristics. Furthermore, the "Minimum Co" metric does not correlate directly with solution quality. For instance, WOA achieves the lowest "Minimum Co" yet performs poorly in "Best Fitness", indicating that this metric likely reflects secondary attributes (e.g., resource cost or constraint penalties). To quantitatively assess sustainability within the proposed optimisation framework, a composite set of key performance indicators (KPIs) was developed, integrating environmental, economic, and operational dimensions. These indicators comprise the water-energy efficiency ratio, which reflects the relationship between total water supply and the associated energy intensity; the normalised cost per unit of demand supplied, representing the economic efficiency of resource allocation; and the reliability under constraint stress tests, which measures the model's capacity to maintain feasible allocations under reduced treatment or supply conditions. The aggregated results, summarised

Table 10. Stability analysis of the top-3 algorithms and cost deviation consistency across runs.

Configuration	Membership function	Defuzzification method	Sustainable extraction threshold	Top-3 algorithms (rank order)	Change in ranking vs. base	Cost variation (%)	Feasibility variation (%)
Base (original)	Triangular	Centroid	90%	JAYA, GWO, PSO	–	–	–
A	Gaussian	Centroid	90%	JAYA, GWO, PSO	No change	1.50%	–1%
B	Triangular	Bisector	90%	JAYA, GWO, PSO	No change	1.20%	0%
C	Triangular	Mean of Maximum	90%	JAYA, TLBO, PSO	1 algorithm change	2.00%	–2%
D	Triangular	Centroid	85%	GWO, JAYA, TLBO	Minor reordering	2.50%	3%
E	Triangular	Centroid	95%	JAYA, GWO, PSO	No change	3.00%	–4%
F	Gaussian	Bisector	85%	GWO, JAYA, TLBO	Minor reordering	3.20%	2%
G	Gaussian	Bisector	95%	JAYA, GWO, PSO	No change	3.50%	–5%

Table 11. Composite sustainability indicators integrating environmental, economic, and operational dimensions.

Algorithm	Demand satisfaction (%)	Reuse volume (m^3)	Reuse (%)	Energy intensity (kWh/m^3)	Aquifer extraction (% of limit)	Observations
ES	98%	12,500	25%	0.45	88%	Excellent balance between cost and sustainability
WOA	96%	14,000	28%	0.48	85%	Highest reuse, lowest extraction
PSO	97%	11,800	23%	0.5	90%	Good cost efficiency, slightly higher extraction

in Table 11, reveal the trade-offs between cost efficiency, feasibility, and long-term sustainability performance across the evaluated algorithms. Therefore, "Best Fitness" remains the principal metric for algorithmic evaluation.

Fuzzy logic provides effective support for managing urban water systems by addressing the uncertainty and complexity inherent in sustainability assessments. Unlike traditional analytical approaches, it allows decision-makers to incorporate incomplete or imprecise information, which is characteristic of environmental data. Accordingly, Chhipi-Shrestha et al. [40] identified sustainability indicators for small and medium-sized urban water systems using fuzzy logic, while Filho et al. [41] employed a fuzzy inference system to quantify sustainability scores in smart cities. Such capacity to represent dynamic and multifaceted conditions is particularly relevant to urban water planning [42].

Complementary approaches, including Integrated Urban Water Management (IUWM) and Water Sensitive Urban Design (WSUD), reinforce coherent sustainability strategies by emphasising community participation and long-term stewardship [43]. Integrating fuzzy logic within community-oriented planning supports adaptable strategies that reflect local priorities while meeting sustainability criteria [44].

Metaheuristic algorithms further enhance planning support by optimising resource allocation and infrastructure development within sustainability constraints [42, 45]. Hybrid methods such as adaptive neuro-fuzzy inference systems combine fuzzy reasoning with neural learning to improve the accuracy of sustainability assessments, particularly where multiple interacting variables must be considered simultaneously [45, 46].

The incorporation of sustainability indicators into decision-making frameworks also improves planning effectiveness. Ren et al. [47] highlight the relevance of life-cycle perspectives and stakeholder input, and fuzzy logic provides a means to evaluate such indices to strengthen resource management outcomes [40, 46]. Overall, integrating fuzzy logic with metaheuristic optimisation offers a scalable approach to managing uncertainty, promoting stakeholder engagement, and supporting resilient and adaptive urban water infrastructure.

To contextualise the present study, recent research demonstrates advances in fuzzy and hybrid approaches for sustainable urban water management. For instance, a hybrid PSOGA-ANN model applied to a decade of consumption data in Texas showed significantly improved monthly prediction accuracy, particularly in seasonal variation modelling [48]. Bellini et al. [49] developed a Mamdani-Sugeno fuzzy system to evaluate water quality in Tivoli, transforming uncertain parameters into operational indices for decision-making in environments with high variability.

Robati and Rezaei [50] applied fuzzy logic to assess 52 sustainability indicators in Tehran, producing thematic maps that guided local strategic planning. Li et al. [51] further suggest that multi-criteria evaluations incorporating environmental, social, and governance dimensions enrich sustainability assessments, while Çalışkan [52] demonstrates that fuzzy sustainability indices are also applicable across other urban infrastructures. Hybrid developments continue to advance practice: Tordecilla et al. [53] show that fuzzy simheuristics effectively address uncertainty in planning scenarios, and Tavoosi et al. [54] demonstrate the utility of type-2 fuzzy neural networks for dynamic parameter adaptation in system identification, confirming the robustness of these methods in applied contexts [55].

Ibeh et al. [56] introduced SEFLAME-CM, which integrates local knowledge and fuzzy logic for water allocation, improving social acceptability and conflict mitigation. Additionally, Felt et al. [57] proposed a deep learning model to detect ocean fronts using satellite imagery; its capacity to identify environmental transitions in near real time suggests applicability to coastal and estuarine urban water systems.

The literature review strengthened the manuscript's thematic coherence with its methodological and environmental scope. Priority was given to studies addressing urban water planning, applications of fuzzy logic in water resource management, and metaheuristic optimisation under environmental or infrastructure constraints.

Conclusions

The consistency observed across five independent executions confers high reliability to the analysis. Empirical evidence strongly indicates that, for the evaluated optimisation problem, algorithm selection is not merely important – it is critical. Among all contenders, Scatter Search, Genetic Algorithm, and Memetic Algorithm emerge not only as suitable choices but as the only rational options for achieving optimal solution quality. In contrast, many modern metaheuristics inspired by natural phenomena exhibit consistently poor performance. Hence, any future methodological decision must be grounded in this robust and unequivocal performance hierarchy. The superiority of this elite trio becomes even more evident when compared to other algorithms. Mid-tier performers, such as Simulated Annealing and Artificial Bee Colony, yield "Best Fitness" values beginning at approximately 1.0e+13, whereas the top three consistently achieve results near 8.2e+12 – a difference of around 20% even in worst-case scenarios. When compared to the poorest-performing methods, such as Firefly Algorithm and Dragonfly Algorithm, which exceed 1.0e+14, the top algorithms are approximately 100 times more effective. In real-world applications like water resource management, such disparity can delineate the boundary between viable planning and operational failure. Crucially, the robustness of Scatter Search, Genetic Algorithm, and Memetic Algorithm is corroborated by their stable performance rankings across all instances. While many methods display considerable volatility in their relative positions, these three remain consistently dominant. This lack of variability is vital for implementation reliability, as it ensures that optimal results can be reproduced regardless of execution. The quantitative evidence is overwhelming: not only do these algorithms deliver superior outcomes by a wide margin, but they also offer unparalleled consistency. This dual advantage makes them the most rational and secure choice for solving the problem at hand with confidence and precision.

To advance the proposed water management optimisation model, seven strategic research lines are identified. First, the integration of hybrid multi-algorithm schemes is proposed, combining global search methods (e.g., PSO, DE, GA) with high-precision local optimisers (e.g., Scatter Search, Memetic Algorithm, Tabu Search) to enhance solution quality without compromising computational efficiency. Second, implementing the model in real smart city environments is essential for validating its responsiveness to operational data and dynamic constraints. Third, incorporating Explainable Artificial Intelligence (XAI) would increase decision transparency, fostering institutional trust and auditability. Fourth, expanding the model to regional scales would require integrating intermunicipal governance, geospatial data, and energy-efficient transport systems. Fifth, an automated mechanism for dynamic constraint adjustment is suggested to simulate regulatory responsiveness, enhancing system resilience. Sixth, developing an interactive web platform would allow stakeholders to configure parameters, upload data, and visualise outcomes, promoting democratic access and policy integration. Lastly, a targeted research agenda is proposed to improve high-performing algorithms (SS, GA, MA). This includes collaborative hybridisation (e.g., SS feeding MA), adaptive evolutionary schemes in GA, traceability and sensitivity analysis in MA, and smart restart mechanisms in SS. Additionally, machine learning-based self-calibration modules (e.g., Bayesian regression, LSTM) could optimise algorithm parameters in real time. A longitudinal comparative analysis is also recommended to evaluate algorithmic performance across spatial and temporal scales, providing insights into scalability and contextual suitability. These future directions aim to establish a robust, adaptive, a nd transparent optimisation framework for sustainable urban and regional water management.

Acknowledgments

The authors are grateful for the support provided by the National Laboratory for Autonomous Vehicles and Exoskeletons (LANAVEX) of the Secretariat of Science, Humanities, Technology and Innovation (SECIHTI) of Mexico.

Conflict of Interest

The authors declare no conflict of interest.

References

1. FAO. The State of the World's Land and Water Resources for Food and Agriculture 2021 – Systems at Breaking Point; Food and Agriculture Organization of the United Nations: Rome, Italy, 1, **2021**.
2. AGUA: Panorama General. Available online: https://www.bancomundial.org/es/topic/water/overview (accessed on 13 **June 2025**).
3. FIGUEIREDO I., ESTEVES P., CABRITA P. Water Wise – A digital water solution for smart cities and water management entities. Procedia Computer Science, **181**, 897, **2021**.
4. KAMBALIMATH S., DEKA P.C. A basic review of fuzzy logic applications in hydrology and water resources. Applied Water Science, **10** (191), 1, **2020**.
5. KRISHNAN S.R., NALLAKARUPPAN M.K., CHENGODEN R., KOPPU S., IYAPPARAJA M., SADHASIVAM J., SETHURAMAN S. Smart water resource management using artificial intelligence – A review. Sustainability, **14** (20), 13384, **2022**.
6. BOURAMDANE A.-A. Optimal Water Management Strategies: Paving the Way for Sustainability in Smart Cities. Smart Cities, **6** (5), 2849, **2023**.
7. QUON H., JIANG S. Decision making for implementing non-traditional water sources: A review of challenges and potential solutions. npj Clean Water, **6**, 56, **2023**.
8. ERDOĞDU A., DAYI F., YILDIZ F., YANIK A., GANJI F. Combining fuzzy logic and genetic algorithms to optimize cost, time and quality in modern agriculture. Sustainability, **17** (7), 2829, **2025**.
9. DÍAZ-PARRA O., AGUILAR-ORTIZ J., RUIZ-VANOYE J.A., TREJO-MACOTELA F.R., BERNÁBE-LORANCA M.B. Optimizing Water Management in Urban Ecosystems: A Holistic Model for the Sustainable Integration of Drinking Water, Rainwater, and Wastewater Systems. Polish Journal of Environmental Studies, **2025**.
10. SALIMIAN F., GHIASSI R. A hybrid method for designing sustainable river monitoring networks using fuzzy logic site selection and genetic algorithm optimization. Water Resources Management, **39** (1), 227, **2025**.
11. XU Y., HUANG G.H., XU L. A fuzzy robust optimization model for waste allocation planning under uncertainty. Environmental Engineering Science, **31** (10), 556, **2014**.
12. PÉREZ MARTÍN M.Á. Understanding nutrient loads from catchment and eutrophication in a salt lagoon: The Mar Menor case. Water, **15** (20), 3569, **2023**.
13. AGUILAR ORTIZ S., SALGADO PINEDA P., MARCO PALLARÉS J., PASCUAL J.C., VEGA D., SOLER J., McKENNA P.J. Abnormalities in gray matter volume in patients with borderline personality disorder and their relation to lifetime depression: A VBM study. PLoS One, **13** (2), 0191946, **2018**.
14. GUPTA I., GUPTA A., KHANNA P. Genetic algorithm for optimization of water distribution systems. Environmental Modelling & Software, **14** (5), 437, **1999**.
15. RADOSAVLJEVIC J. Overview of genetic algorithms. IET Digital Library, 39. **2018**.
16. LI Z., LIN X., ZHANG Q., LIU H.-L. Evolution strategies for continuous optimization: A survey of the state-of-the-art. Swarm and Evolutionary Computation, **56**, 100694, **2020**.
17. CORNE D., LONES M.A. Evolutionary Algorithms. In: Martí R., Panos P., Resende M. (eds) Handbook of Heuristics. Springer, Cham. **2018**.
18. KINCAID R., NINH A. Simulated Annealing. Springer International Publishing, 221, **2023**.
19. GLOVER F. Tabu search – Part I. ORSA Journal on Computing, **1** (3), 190, **1989**.
20. DORIGO M., MANIEZZO V., COLORNI A. Ant system: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics, **26** (1), 29, **1996**.
21. KENNEDY J., EBERHART R. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, **4**, 1942, **1995**.
22. STORN R., PRICE K. Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization, **11** (4), 341, **1997**.
23. KARABOGA D., BASTURK B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. Journal of Global Optimization, **39** (3), 459, **2007**.
24. MLADENOVIĆ N., HANSEN P. Variable neighborhood search. Computers & Operations Research, **24** (11), 1097, **1997**.
25. WARNARS H.L.H.S., WARNARS L.S., UTOMO W.H., DOUCET A., RAMADHAN A., SISWANTO T. Memetic Algorithm Small Survey For 2019 Published Papers, 2024 3rd International Conference on Creative Communication and Innovative Technology (ICCIT), Tangerang, Indonesia, 1, **2024**.
26. GLOVER F. A template for scatter search and path relinking. In Artificial Evolution; Hao J.K., Lutton E., Ronald E., Schoenauer M., Snyers D., Eds.; Springer: Berlin, Germany, 13, **1998**.
27. TIAN Z., ZHANG C. An Improved Harmony Search Algorithm and Its Application in Function Optimization. Journal of Information Processing Systems, **14** (5), 1237, **2018**.
28. KUMAR D., GANDHI B.G.R., BHATTACHARJYA R.K. Firefly Algorithm and Its Applications in Engineering Optimization. In: Bennis F., Bhattacharjya R. (eds) Nature-Inspired Methods for Metaheuristics Optimization. Modeling and Optimization in Science and Technologies, vol 16. Springer, Cham, **2020**.
29. YANG X.S., DEB S. Cuckoo search via Lévy flights. In Proceedings of the World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), 210, **2010**.

30. WANG Y., GAO S., YU Y., CAI Z., WANG Z. A gravitational search algorithm with hierarchy and distributed framework. Knowledge Based Systems, **218**, 106877, **2021**.
31. MIRJALILI S., LEWIS A. The Whale Optimization Algorithm. Advances in Engineering Software, **95**, 51, **2016**.
32. YANG X.S. A new metaheuristic bat inspired algorithm. In Nature Inspired Cooperative Strategies for Optimization (NICSO 2010); González J.R., Pelta D.A., Cruz C., Terrazas G., Krasnogor N., Eds.; Springer: Berlin, Germany, 65, **2010**.
33. SHERINOV Z., UNVEREN A. Multi-objective imperialistic competitive algorithm with multiple non-dominated sets for the solution of global optimization problems. Soft Computing, **22** (24), 8273, **2018**.
34. ABUALIGAH L., ABU-DALHOUM E., IKOTUN A.M., ABU ZITAR R., ALSOUD A.R., KHODADADI N., EZUGWU A.E., HANANDEH E. S., JIA H. Teaching–learning-based optimization algorithm: analysis study and its application. Elsevier BV., 59, **2024**.
35. JALILI S. Cultural Algorithms (CAs). In: Cultural Algorithms. Engineering Optimization: Methods and Applications. Springer, Singapore, **2022**.
36. SIMON D. Biogeography Based Optimization. IEEE Transactions on Evolutionary Computation, **12** (6), 702, **2008**.
37. MIRJALILI S. The Ant Lion Optimizer. Advances in Engineering Software, **83**, 80, **2015**.
38. SUN J., XU W., FENG B. A global search strategy of quantum behaved particle swarm optimization. In Proceedings of the 2004 IEEE Conference on Cybernetics and Intelligent Systems, 111, **2004**.
39. MIRJALILI S. Dragonfly algorithm: A new meta-heuristic optimization technique for solving single objective, discrete, and multi objective problems. Neural Computing and Applications, **27** (4), 1053, **2016**.
40. CHHIPI-SHRESTHA G., HEWAGE K., SADIQ R. Selecting sustainability indicators for small to medium sized urban water systems using fuzzy-electre. Water Environment Research, **89** (3), 238, **2017**.
41. FILHO O.R.D.C., LIMA W.G., OLIVEIRA R.F.A.P.D. Smart sustainable cities: using a fuzzy inference system to determine their global score. Global Journal of Science Frontier Research, 1, **2019**.
42. LI H., XIA Q., WANG L., MA Y. Sustainability assessment of urban water environment treatment public-private partnership projects using fuzzy logic. Journal of Engineering Design and Technology, **18** (5), 1251, **2020**.
43. LINDSAY J., ROGERS B., CHURCH E., GUNN A.W., HAMMER K., DEAN A. J., FIELDING K.S. The role of community champions in long-term sustainable urban water planning. Water, 11 (3), 476, 2019.
44. RUSTUM R., KURICHIYANIL A., FORREST S., SOMMARIVA C., ADELOYE A., ZOUNEMAT-KERMANI M., SCHOLZ M. Sustainability ranking of desalination plants using mamdani fuzzy logic inference systems. Sustainability, **12** (2), 631, **2020**.
45. NILASHI M., CAVALLARO F., MARDANI A., ZAVADSKAS E.K., SAMAD S., IBRAHIM O. Measuring country sustainability performance using ensembles of neuro-fuzzy technique. Sustainability, **10** (8), 2707, **2018**.
46. TAN Y., SHUAI C., JIAO L., SHEN L. Adaptive neuro-fuzzy inference system approach for urban sustainability assessment: a China case study. Sustainable Development, **26** (6), 749, **2018**.
47. REN J., REN X., SHEN W., MAN Y., LIN R., LIU Y., DONG, L. Industrial system prioritization using the sustainability-interval-index conceptual framework with life-cycle considerations. Aiche Journal, **66** (6), **2020**.
48. ZUBAIDI S.L., AL-BUGHARBEE H., ALATTABI A.W., RIDHA H.M., HASHIM K., AL-ANSARI N., YASEEN Z.M. Forecasting urban water demand using different hybrid-based metaheuristic algorithms inspire for extracting artificial neural network hyperparameters. Scientific Reports, **14** (1), 24042, **2024**.
49. BELLINI F., BARZEGAR Y., BARZEGAR A., MARRONE S., VERDE L., PISANI P. Sustainable Water Quality Evaluation Based on Cohesive Mamdani and Sugeno Fuzzy Inference System in Tivoli (Italy). Sustainability, **17** (2), 579, **2025**.
50. ROBATI M., REZAEI F. Evaluation and ranking of urban sustainability based on sustainability assessment by fuzzy evaluation model. International Journal of Environmental Science and Technology, 1, **2021**.
51. LI Y., HE N., LI H., ZHANG Y. Sustainability assessment of urban water public-private partnership projects with environmental, social, and governance (ESG) criteria. Jawra Journal of the American Water Resources Association, **60** (6), 1209, **2024**.
52. ÇALIŞKAN B. Integrated and sustainable performance evaluation of urban rail transit systems using fuzzy sustainability index. Decision Making and Analysis, 73, **2024**.
53. TORDECILLA R., COPADO-MÉNDEZ P., PANADERO J., QUINTERO-ARAÚJO C., MONTOYA-TORRES J., JUAN, Á. Combining heuristics with simulation and fuzzy logic to solve a flexible-size location routing problem under uncertainty. Algorithms, **14** (2), 45, **2021**.
54. TAVOOSI J., MOHAMMADZADEH A., JERMSITTIPARSERT K. A review on type-2 fuzzy neural networks for system identification. Soft Computing, **25** (10), 7197, **2021**.
55. CASTILLO O., PERAZA C., OCHOA P., AMADOR-ANGULO L., MELÍN P., PARK Y., GEEM Z. Shadowed type-2 fuzzy systems for dynamic parameter adaptation in harmony search and differential evolution for optimal design of fuzzy controllers. Mathematics, **9** (19), 2439, **2021**.
56. IBEH C., SMITH J., KUMAR R. SEFLAME-CM: A spatially explicit framework combining community input and fuzzy logic for water resource conflict management. Sustainability, **17** (5), 2315, **2025**.
57. FELT V., KACKER S., KUSTERS J., PENDERGAST J., CAHOY K. Fast ocean front detection using deep learning edge detection models. TechRxiv, **2022**.