International Journal of Engineering Trends and Technology
ISSN: 2231-5381/ https://doi.org/10.14445/22315381/IJETT-V74I6P113

Volume 74 Issue 6, 187-200, June 2026

Original Article

A Systematic Literature Review on Fuzzy-based Metaheuristic Algorithms for Combinatorial Test Suite Generation

Nur Syabila Zabidi[1], Noraini Ibrahim[1], Asmau Osman[2]

[1]*Faculty of Computer Science & Information Technology, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia.*
[2]*Faculty of Computing, Nile University of Nigeria, Abuja, Nigeria.*

[1]*Corresponding Author : nursyabilazabidi@gmail.com*

Received: 25 September 2025 Revised: 04 March 2026 Accepted: 20 April 2026 Published: 27 June 2026

Abstract - *Combinatorial Testing (CT), also known as t-way test suite generation, results in compact test suites that ensure that all t-way parameter interactions are tested. In this regard, metaheuristics such as Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and the Sine Cosine Algorithm (SCA) have been used to generate compact test suites. However, these methods often rely on fixed parameter settings, which may lead to premature convergence and poor solution quality. To overcome this limitation in combinatorial testing methods, fuzzy logic has been proposed as a dynamic parameter-tuning method. Despite the potential of fuzzy logic in combinatorial testing, there is no systematic review on this topic. This research aims to guide a Systematic Literature Review using the PRISMA and Barbara Kitchenham frameworks. Out of 61 studies identified initially, only three primary journal articles were found to meet the selection criteria after a careful process following identification, screening, eligibility, quality evaluation, and data extraction. The results are synthesized into four major themes: algorithmic categories, fuzzy-logic implementation, parameter tuning, and performance results. The results show that fuzzy logic has been applied to teaching-learning-based, swarm-based, and ant colony optimization algorithms using high-level and parameter tuning approaches. All studies show improved results in test suite reduction, convergence speedup, and coverage efficiency.*

Keywords - *Combinatorial Testing, Fuzzy Logic, Metaheuristic Algorithms, Systematic Literature Review (SLR), T-Way Testing.*

1. Introduction

Software testing remains a crucial aspect of ensuring the stability and quality of software systems, particularly as program complexity continues to increase. Among the many testing strategies established over the decades, Search-Based Software Testing (SBST) has emerged as a viable paradigm that uses metaheuristic optimization to solve a variety of test-related problems through search. SBST, a subsection of Search-Based Software Engineering (SBSE), has been widely utilized to generate automated test data, prioritize test cases, minimize test suites, and optimize test oracles [1-3]. Metaheuristic algorithms such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO) [4, 5], Ant Colony Optimization (ACO) [6, 7], and Sine Cosine Algorithm (SCA) [8, 9] have shown remarkable effectiveness in exploring vast, complex search spaces, especially when dealing with problems involving numerous constraints and conditions. These four algorithms are frequently recognized in existing studies as some of the most applied and representative metaheuristic methods in software testing [10], which makes them appropriate choices for benchmarking in this research.

Within the SBST domain, one of the most intensively explored areas is combinatorial testing, particularly t-way testing, which aims to produce compact test suites that ensure coverage of all t-way parameter interactions [11]. While conventional metaheuristic-based methods have enhanced the effectiveness of t-way test generation, they still face a recurring issue: the reliance on fixed or manually adjusted parameters [12], such as crossover probability [4, 13], inertia weight [14], and learning coefficients [15]. These static parameters often fail to generalize across different testing contexts, resulting in poor robustness in dynamic environments. Improper settings may lead to premature convergence, stagnation in local optima, or instability across different problem instances [16].

Despite the maturity of combinatorial testing research, a critical gap exists in the systematic synthesis of adaptive mechanisms. While traditional reviews focus on the efficiency of search algorithms themselves, there is a notable lack of analysis regarding how these algorithms can self-adjust during the search process. The lack of these algorithms drives the development of fuzzy-based metaheuristic strategies, especially fuzzy logic. For instance, by using a dynamic search operator, the Fuzzy Adaptive Teaching Learning-Based Optimization (FATLBO) approach accomplished better test suite reduction [17]. In another work that employs fuzzy inference to control inertia weight and acceleration coefficients, the Fuzzy Adaptive PSO (FAPSO) achieved an effective balance between exploration and exploitation [18]. Nevertheless, the literature exhibits a scattered pattern. Contrary to other literature on search-based software testing, this paper's literature review fails to provide an in-depth overview of algorithm types, fuzzy logic systems, and parameter tuning techniques, and to categorize them by their hybridization mechanisms [19].

This paper proposes the first comprehensive literature review that emphasizes the importance of using fuzzy-based metaheuristic algorithms in combinatorial test suite generation. This paper established a benchmark for creating self-adaptive search-based testing tools using fuzzy control systems. In addition, this paper recognized the limitations of the ongoing process of evaluating search-based testing tools. While conventional research uses five key performance indicators, the literature review of this paper found that only three parameters, namely size, convergence, and coverage, are used in evaluating fuzzy-based metaheuristic algorithms. This indicates a promising direction for future research. This paper aims to answer five research questions, namely: (i) What types of fuzzy-based metaheuristic algorithms have been proposed for t-way testing? (ii) How do these algorithms implement fuzzy logic? (iii) What types of parameter tuning techniques can metaheuristic algorithms implement using fuzzy logic? (iv) How do fuzzy-based metaheuristic algorithms deal with test suite size and coverage in comparison with other algorithms? (v) What limitations and suggestions for future research exist for fuzzy-based metaheuristic algorithms, particularly in t-way testing?

This paper aims to answer the five research questions through an in-depth literature review of the proposed themes, namely: (i) fuzzy-based metaheuristic algorithms, which answers RQ1; (ii) fuzzy logic implementation techniques, which answers RQ2; (iii) parameter tuning techniques, which answers RQ3; and (iv) performance analysis in the context of test suite generation, which answers RQ4 and RQ5. This study proposes the first broad, systematic literature review that highlights fuzzy-based metaheuristic algorithms for combinatorial test suite generation. By incorporating fuzzy control systems into search-based testing strategies, this study established a benchmark for the development of self-adaptive testing tools in future research. In addition, the findings acknowledge limitations in ongoing evaluation practices. Although standard research commonly utilizes five key performance indicators, fuzzy-based approaches are generally evaluated using only three metrics, specifically size, convergence, and coverage, signifying a meaningful direction for future research. Five research questions will be discussed: (i) RQ1: What types of fuzzy-based metaheuristic algorithms have been suggested for t-way testing? (ii) RQ2: How is fuzzy logic used in these algorithms? (iii) RQ3: By using fuzzy logic, which parameter tuning techniques are suitable for metaheuristic algorithms? (iv) RQ4: In relation to test suite size and coverage, how do fuzzy-based algorithms operate? (v) RQ5: What are the constraints and future research suggestions for fuzzy-based metaheuristic algorithms, particularly in t-way testing?

By addressing these research questions, the study aims to (i) identify and categorize fuzzy-based metaheuristic algorithms, (ii) examine fuzzy logic implementation techniques, (iii) analyze parameter tuning strategies, (iv) evaluate performance outcomes in terms of test suite size, convergence behaviors, and computational efficiency, and (v) highlight research gaps with directions for future work. The review is also organized into four themes: fuzzy-based metaheuristic algorithms (addressing RQ1), fuzzy-logic implementation techniques (addressing RQ2), parameter-tuning strategies (addressing RQ3), and performance analysis in the context of test suite generation (addressing RQ4 and RQ5).

The remainder of this paper is organized as follows: Section 2 reviews related work on metaheuristics and fuzzy logic, and their applications in combinatorial testing. Section 3 describes the research methodology, including the review methodology, database selection, search strategy, and inclusion criteria. Section 4 presents the results of the review, structured into four thematic categories: performance analysis, types of fuzzy-based algorithms, fuzzy logic implementation techniques, and parameter tuning strategies. Section 4 presents the key components of fuzzy-based metaheuristics, specifically in the context of combinatorial testing. Section 5 also discusses the findings in relation to the five research questions, identifying prevailing trends, challenges, and opportunities for future research. Finally, Section 6 concludes the paper by summarizing the main contributions and outlining implications for both research and practice.

2. Related Works

Several Systematic Literature Reviews (SLRs) have already investigated the use of metaheuristic algorithms in Combinatorial Testing (CT). For instance, a systematic review of combinatorial testing strategies was presented in [20], robust swarm intelligence algorithms in search-based software engineering were examined in [21], and a review of the emperor penguin optimizer was conducted in [22]. These works demonstrate the significance of metaheuristics in software testing. However, the integration of fuzzy logic with combinatorial testing has not been discussed. Traditional literature reviews contribute to supplying the general understanding of different metaheuristics. On the contrary, several methodological issues, publication bias, a narrow scope, and comprehensive deficiencies were discovered [23]. In view of this situation, a Systematic Literature Review (SLR) will yield precise, reproducible knowledge. The SLR consists of a straightforward search process, inclusion and exclusion criteria, and systematic data analysis [24].

2.1. Metaheuristics in Combinatorial Testing

Several metaheuristics have been employed in combinatorial testing. Genetic Algorithms (GA) utilize crossover and mutation to refine solutions iteratively [25]. Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) mimic the collective behaviors of natural swarms to balance the exploration and exploitation process [6, 18]. Additionally, Teaching-Learning Based Optimization (TLBO) interacts with the knowledge transfer between the teacher and learner phases during the search process [17]. The Sine Cosine Algorithm (SCA) also employs trigonometric functions to search the search space. These algorithms rely on manual parameter tuning. Consequently, research has increasingly focused on adaptive and hybrid approaches to achieve greater optimization efficiency and algorithmic robustness.

2.2. Performance Analysis in Metaheuristics-based Combinatorial Testing

Several performance indicators are employed to measure the effectiveness of metaheuristic algorithms. To evaluate the covering array, the test suite size has been used as the first indicator [26]. To detect how quickly an algorithm reaches near-optimal solutions, the second indicator, convergence speed, has been used [27]. The third indicator is coverage efficiency, which has been employed to measure the completeness of interaction coverage within different configurations [11]. At the same time, execution time has also been used to evaluate computational efficiency in situations with increasing numbers of parameters and interaction strengths [28].

2.3. Limitations of Static Parameter Settings

Static parameters, such as crossover and mutation rates in GA [17], inertia weight and acceleration coefficients in PSO [18], pheromone evaporation rates in ACO [6], and the teaching and learning phases in TLBO [26], cannot be adjusted to different conditions and situations [29]. Therefore, several parameter control strategies were outlined to tackle the limitation [12]. The first category is deterministic control. This strategy adjusts parameters according to a predefined schedule, such as linearly reducing the inertia weight as iterations progress [30]. The second category is adaptive control. This technique requires a real-time response in order to check population diversity or convergence rate [31]. The third category is self-adaptive control. This category allows the algorithm to accommodate its behavior dynamically. Within the candidate solutions, this strategy controls the parameters [32]. Among all these parameter control strategies, fuzzy logic attracts attention because of its ability to handle uncertainty and rigid mathematical assumptions [33].

2.4. Fuzzy Logic as an Adaptive Mechanism in Metaheuristics

Fuzzy systems operate by utilizing a collection of linguistic rules that convert uncertain or changing conditions into practical control decisions [33]. Rather than relying on precise numerical thresholds, a concept of low, medium, and high guides the system in a more flexible method. This situation has therefore positioned fuzzy logic as an adaptive component in metaheuristic optimization. Several studies highlight the benefits of embedding fuzzy logic into the search algorithms to improve their responsiveness during execution. The Fuzzy Adaptive Teaching Learning-Based Optimization (FATLBO) approach, as a variant of the traditional TLBO method, has been proposed as a specialized tool to overcome the limitations of the fixed parameterization approach for combinatorial testing. By representing the problem constraints using qualitative values such as "low" and "high" diversity, the proposed approach ensures robustness for different types of testing. This eliminates the problem of instability that arises when using the traditional approach, thereby ensuring the generation of quality results. The use of fuzzy rules to fine-tune the inertia weight and acceleration coefficients is an important aspect of Fuzzy Adaptive Swarm Optimization, referred to as FAPSO, proposed to overcome the limitations of the traditional Swarm Intelligence-based approach, such as Particle Swarm Optimization (PSO). Variable-Strength Tuned Ant Colony Optimization (VS-TACO), as a variant of the traditional Ant Colony Optimization (ACO), has been proposed as a specialized tool for the generation of variable-strength t-way test suites. Fuzzy logic, as an important aspect of the proposed approach, helps to overcome the limitations of the traditional ACO, which often makes use of fixed parameterization, such as pheromone evaporation rates and ant population. In conclusion, the transition from fixed mathematical models to flexible fuzzy-logic frameworks reduces sensitivity to initial configurations and allows generators to produce stable, high-quality test suites across various problem settings without the need for manual retuning.

3. Methodology

This section begins with the selection of a review protocol (Section 3.1), followed by the formulation of research questions (Section 3.2), the design of systematic search strategies (Section 3.3), and subsequent steps including eligibility assessment, quality appraisal, and data extraction (Sections 3.3.1-3.3.5).

3.1. Review Protocol - PRISMA

This SLR refers to the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA), first introduced in [34] and later enhanced in [35]. PRISMA fosters thorough searches, emphasizes the selection of high-quality articles through a quality evaluation procedure, and provides explicit instructions on the approach to be used, making it ideal for SLR (Figure 1). Even though this SLR is unrelated to the medical and health domain, the methodological flexibility of PRISMA was emphasized in [35], making it acceptable for use in other fields. This SLR was conducted using four primary methodological procedures: the formulation of research questions, a systematic search, article quality assessment, and data extraction and analysis, in accordance with the PRISMA guidelines. This SLR also refers to the proposed methodology in [36], which was primarily aimed at software engineering research. The activities include planning, conducting, and reporting to SLR. This methodology has been utilized in [20-22] to present an in-depth study of the adoption of metaheuristics in software engineering. Therefore, this SLR combines the PRISMA methodology and Kitchenham's guidelines.

3.2. Formulation of the Research Question

Research questions are important in SLR. Research questions are one element that helps researchers accomplish their objectives. This SLR's research questions are based on the PICO mnemonic. PICO was introduced in [37], where P stands for population, I for interest, and CO for context. Metaheuristics represent P, fuzzy-based represents I, and combinatorial testing represents CO for this SLR. This SLR has generated research questions on fuzzy-based metaheuristic algorithms, focusing on these three keywords.

3.3. Systematic Search Strategies

The second step in this Systematic Literature Review (SLR) involves developing a systematic search strategy that enables researchers to conduct an exhaustive search. Three primary procedures underpin this SLR's systematic search strategies: eligibility, screening, and identification (Figure 2).

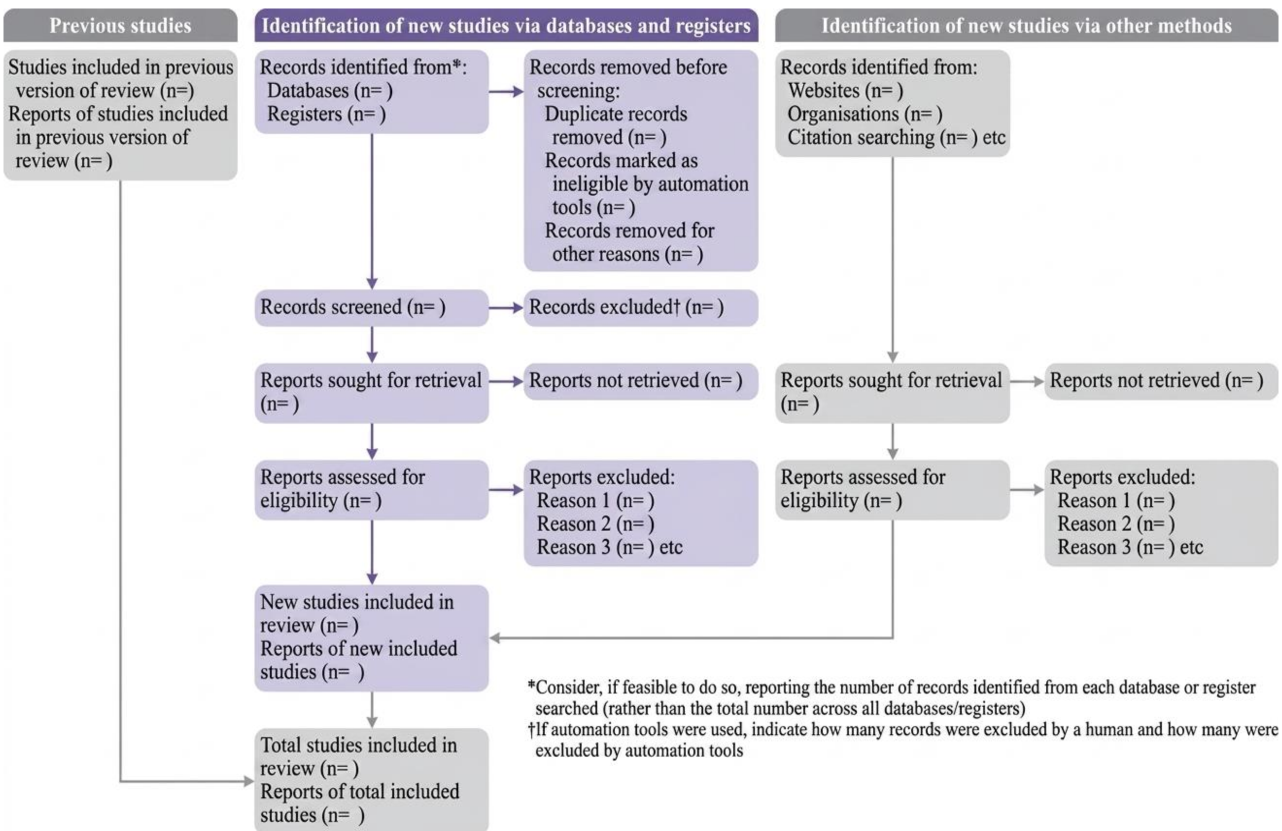

Fig. 1 PRISMA 2020 flow diagram template for systematic reviews

3.3.1. Identification

Identification is the first step in the systematic search process, during which they choose several appropriate keywords depending on the predetermined study topics. Three primary keywords have been chosen: combinatorial testing, fuzzy-based, and metaheuristics. The researchers have since sought synonyms, related terms, and variants to expand the keyword selection. ChatGPT was used alongside online thesauruses, expert opinions, previous research terms, and databases such as Scopus in the current study. Through this method, the author identified several other terms, including fuzzy logic and *t*-way testing. To ensure the comprehensiveness of the search strings, a pilot search was conducted using expert consultation and AI-assisted keyword expansion (ChatGPT). This statement pertains to the identification phase, the first step in the systematic search approach adopted for the Systematic Literature Review (SLR). The main aim of this phase is to go beyond the initial keywords and conduct a thorough search to ensure no research is left behind due to different terminologies. Selecting Scopus, IEEE Xplore, and ScienceDirect as the main databases was a strategic move to ensure the identification of high-quality, relevant research on fuzzy-based metaheuristics for combinatorial testing.

To maximize the effectiveness of these searches, researchers employed advanced search techniques and a multi-stage filtering process to refine the initial results into a final set of primary studies. As shown in Table 1, the search strings developed illustrate how these functions were applied to the primary keywords: combinatorial testing, fuzzy-based, and metaheuristics. The logical operators used in the search were AND and OR, where OR combined words that were related or synonymous, such as "t-way" OR "tway" OR "covering array." The operator AND was used to ensure that the search results included all three search components: the testing domain, the optimization method, and the fuzzy logic component. Double quotes were used to ensure precise search results, for example, when searching for "combinatorial testing" and "covering array."

This ensures that the database does not retrieve unrelated results that may contain any of the search words. The researchers used the asterisk (*) as a wildcard to search for different word endings, for example, in searching for "meta*heuristic*," the database would retrieve results containing "metaheuristic," "metaheuristics," or "meta-heuristic." Similarly, in searching for "algorithm*," the database would retrieve results containing either "algorithm" or "algorithms." The database codes, such as TITLE-ABS-KEY in Scopus, were used to search in the title, abstract, and keywords of articles. This ensures that the results are centrally relevant to the research topic rather than just mentioning a term in passing within the full text.

Within the identification phase, a total of 61 potentially relevant articles were captured, as compiled in Table 1. The identification of the 61 potentially relevant articles was the first stage of the systematic search strategy. This phase was intended to ensure that the subsequent review was built on a robust and exhaustive foundation of existing literature.

Table 1. The search string

Database	Search String
Scopus	TITLE-ABS-KEY ((“combinatorial testing” OR “t-way” OR “tway” OR “covering array”) AND (“meta*heuristic*” OR “algorithm*”) AND (“fuzzy”))
IEEE Xplore	combinatorial testing "combinatorial testing" OR "t way" OR tway OR "covering array" OR "meta * heuristic *" OR "algorithm *" "fuzzy."
Science Direct	“covering array” AND “metaheuristic” AND “fuzzy.”

3.3.2. Screening

The specific criteria used in this screening process are given in Table 2. Three major selection criteria used in this SLR were applied in this study: publication year (2020-2024), publication type (journal), and subject area (combinatorial testing). Forty-nine articles were removed from this screening process. The remaining articles were moved on to the next process in this research: eligibility assessment. The selection criteria were set to ensure a high level of quality. By limiting this review to journal articles from 2020 to 2024, this research focuses on in-depth results rather than a broader, lower-quality set from conference papers. This approach ensures that the resulting synthesis and taxonomy are built upon the most robust evidence available in current literature.

Table 2. Inclusion and exclusion criteria

Criterion	Inclusion	Exclusion
Timeline	2020-2024	2019 and earlier
Document Type	Journal (with empirical data)	Review articles, chapters in a book, book, conference proceedings, etc.
Subject Area	Combinatorial Testing and Fuzzy-based	Engineering, feature selection, optimization problems, medical, image segmentation, etc.

3.3.3. Eligibility

Eligibility is the final process in a systematic search strategy, which involves determining whether the selected articles are truly relevant to the research questions and objectives of the SLR. The focus during eligibility is primarily on reading the titles and abstracts of the articles; if necessary, the researcher will also review the main content. Based on this process, a total of 6 articles were excluded due to their focus on global optimization problems rather than software testing, a lack of clear explanation of fuzzy-based strategies, and a lack of clear explanation of the context of combinatorial testing. The remaining articles are then carried forward to the next process, the quality assessment.

3.3.4. Quality Appraisal

A total of six articles were assessed for quality based on the six criteria proposed in [38]. The six criteria used are QA1. Is the purpose of the study clearly stated? QA2. Are the interest and the usefulness of the work clearly presented? QA3. Is the study methodology clearly established? QA4. Are the concepts of the approach clearly defined? QA5. Is the work compared and measured with other similar work? QA6.

Are the limitations of the work clearly mentioned? The lead researcher conducted the quality assessment of the selected articles with the assistance of one co-researcher. For each criterion, the researchers provided three response options: Yes (score of 1.0), Partially Yes (score of 0.5), and No (no score). Articles assessed must record a minimum score of 3.0 to be accepted in this SLR. Based on the assessment, 3 articles achieved the minimum score of 3.0 and were carried forward to the next step: data extraction and analysis.

3.3.5. Data Extraction and Analysis

The next process is data extraction and analysis. The extracted data will then be properly sorted into a table. The gathered data is then evaluated thematically.

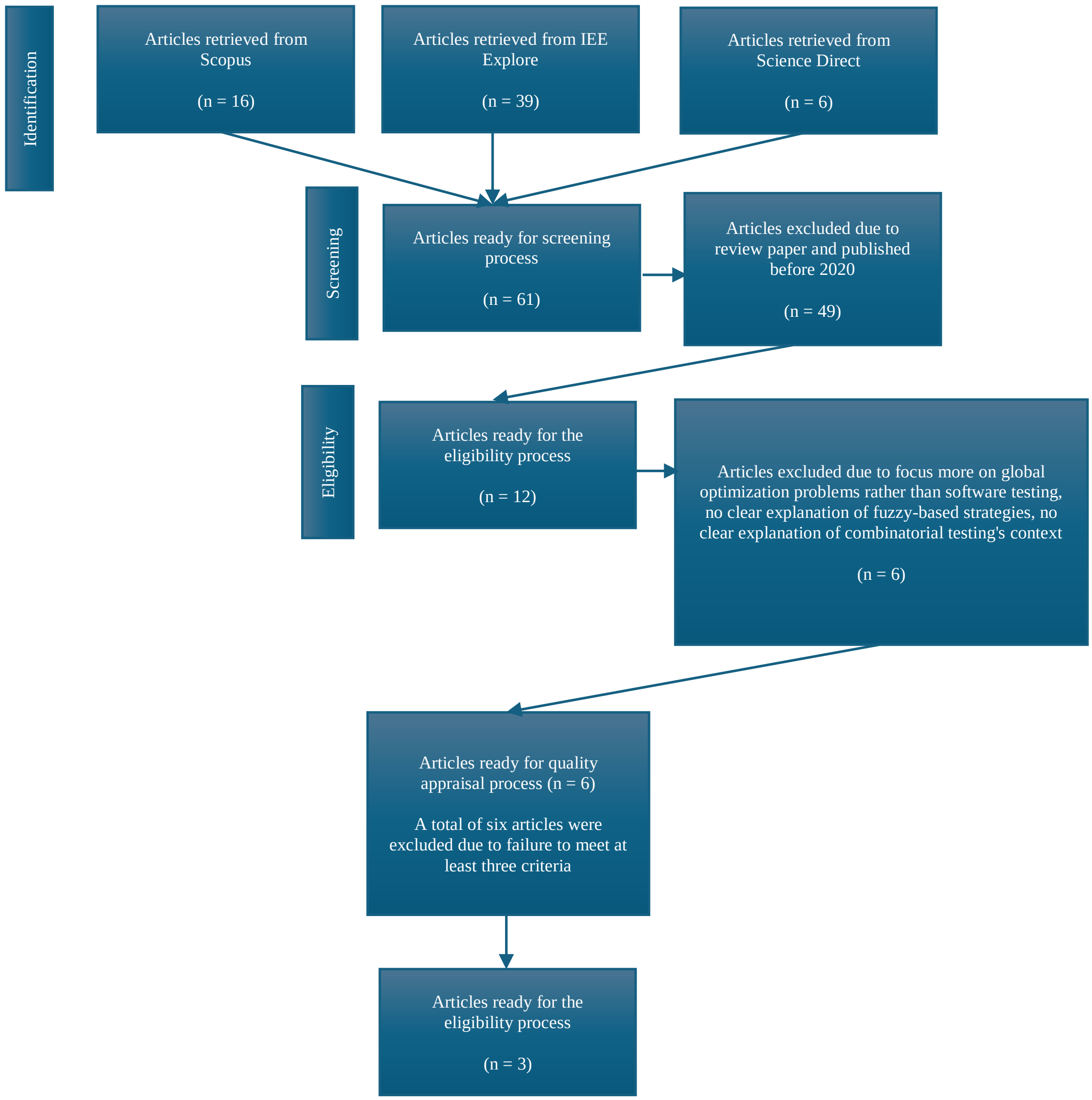

Fig. 2 Three primary processes

Based on this, four themes were selected, namely performance analysis of fuzzy-based metaheuristic algorithms, types of fuzzy-based algorithms, fuzzy logic implementation techniques, and parameter tuning strategies.

4. Key Components for Fuzzy-Based Adaptive Metaheuristics in Combinatorial Testing

4.1. Problem Formulation

Combinatorial testing is often closely related to the mathematical idea of Covering Arrays (CA). Because of this, combinatorial testing sometimes uses CA notation to denote t-way tests [39]. An array of size N with v values is expressed by the notation. $CA_\lambda\,(N;\ t, k, v)$, where k is the number of components (or parameters), and every $N \times t$ sub-array includes all ordered subsets from the v values of size t at least λ times [40, 41].

To simplify this definition, it can be said that the notation. $CA_\lambda\,(N;\ t, k, v)$ represents an array with N rows and k columns, filled with v different values. It means that in any set of t columns you pick, every possible combination of t values appears at least λ times in those rows. To enhance clarity, the covering array is often represented as $CA\,(N;\,t,k,v)$ or simply as $CA\,(N;\,t,v^k)$. A comprehensive example to describe the definitions can be found in Table 3. Consider a scenario where a pizzeria offers a menu with 7 different pizza toppings, each of which can be either available (represented by 1) or unavailable (represented by 0) on a pizza. If the pizza worker creates 8 different pizzas using these toppings, they must ensure that every possible combination of any two toppings (00, 01, 10, 11) appears exactly twice in any 8×2 subarray of the pizza configurations. This scenario indicates that the array has an index $\lambda = 2$, meaning each pair of topping combinations is equally represented across the pizzas. Referring to the example of pizza toppings selection in Table 3, the test suite can be expressed as $CA\,(8;\,2,2^7)$.

Table 3. A representation of $OA\,(8;2,7,2)$

Pizzas	Toppings						
	Pepperoni	Mushroom	Onion	Sausage	Chicken	Green Peppers	Olive
1	0	0	0	0	0	0	0
2	0	0	0	1	1	1	1
3	0	1	1	0	0	1	1
4	0	1	1	1	1	0	0
5	1	0	1	0	1	0	1
6	1	0	1	1	0	1	0
7	1	1	0	0	1	1	0
8	1	1	0	1	0	0	1

The problem of generating a covering array can be formulated as an optimization task, as shown in Equation (1). In this formulation, $f(x)$ is the objective function representing the total number of covered interactions; x is the set of decision variables; and x_i is the domain of each variable, that is, $x_i = \{x_i(1), x_i(2), \dots, x_i(K)\}$, where K is the number of discrete values for each parameter (columns). The goal is to find an optimal set of configurations that maximizes interaction coverage.

$$Maximize\ f(x) = \sum_{i=1}^{N} x_i \tag{1}$$

$$Subject\ to\ x \in x_i, i = 1,2,\dots,N$$

4.2. The Fuzzy Inference System

A fuzzy reasoning system, also known as a Fuzzy Inference System (FIS), comprises four primary components: fuzzification, a knowledge base, inference mechanisms, and defuzzification [42].

As depicted in Figure 3, fuzzification converts crisp numerical inputs into fuzzy values by mapping them to predefined membership functions. The inference engine processes these fuzzy rules to derive fuzzy outputs, which are subsequently converted back into crisp values through the defuzzification process [43]. A crisp value refers to a precise numerical result produced after a fuzzy logic system interprets fuzzy input into specific outputs.

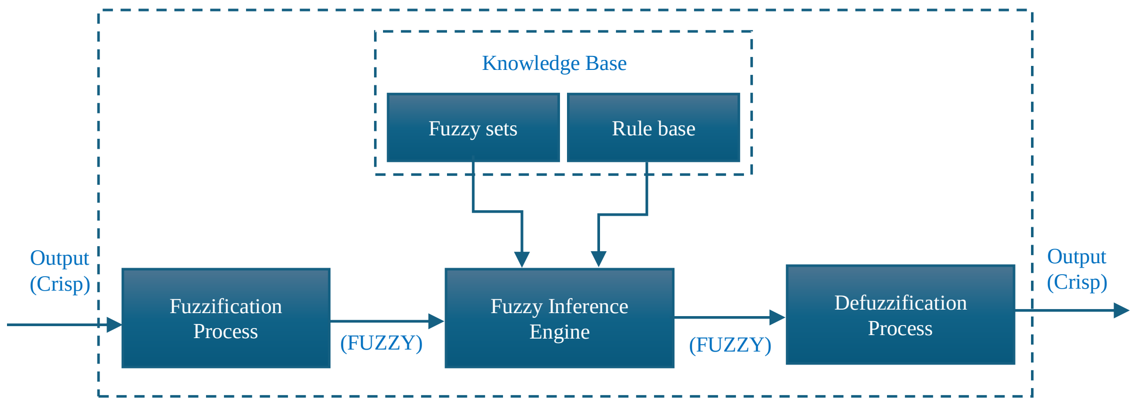

Fig. 3 Fuzzy inference system

Mamdani fuzzy systems, also known as Fuzzy Inference Systems (FIS), are designed to handle imprecision and uncertainty by utilizing a framework that mirrors human-like reasoning through linguistic rules. Unlike traditional mathematical models that rely on precise numerical thresholds, these systems use qualitative descriptors, such as "low," "medium," or "high," to guide system behavior [44]. In search-based optimization, integrating fuzzy systems provides a powerful adaptive mechanism to overcome the limitations of traditional metaheuristic algorithms that rely on fixed or manually adjusted parameters. Traditionally, this transition is governed by static linear decay, a deterministic approach in which control parameters follow a predefined schedule, such as linearly reducing the inertia weight as iterations progress. However, this method is often unreliable because it assumes a uniform search landscape and cannot adapt to real-time search conditions. Specifically, linguistic variables such as population diversity and iteration progress are used as fuzzy inputs. For instance, terms like "low diversity," "medium diversity," or "high diversity" represent the spread of solutions within the population.

5. Results and Discussion

5.1. Background of Selected Studies

Following the systematic processes of identification, where potential studies were gathered using predefined keywords and database searches; screening, where articles were filtered based on inclusion and exclusion criteria; eligibility, where the relevance of each study was assessed through title, abstract, and content review; quality appraisal, where studies were evaluated against established methodological criteria; and data extraction and analysis, where relevant information was synthesized into thematic categories, a total of three primary studies were identified as highly relevant to the objectives of this SLR. These studies represent the core body of work on fuzzy-based metaheuristic algorithms applied to generating combinatorial test suites. The first study introduced the Fuzzy Adaptive Teaching Learning-Based Optimization (FATLBO) strategy [17]. This work demonstrated how fuzzy inference could dynamically adjust the exploration and exploitation behavior of TLBO operators, resulting in reduced test suite size and improved performance in generating mixed-strength covering arrays. The second study proposed a fuzzy logic-based adaptive swarm optimization method [18]. This study employed fuzzy rules to dynamically adjust the inertia weight and acceleration coefficients in real-time. The results showed significant improvements in convergence rate and coverage efficiency when compared to conventional swarm-based techniques. The third study introduced the Variable-Strength Tuned Ant Colony Optimization (VS-TACO) algorithm [6]. This approach integrates fuzzy logic to dynamically adjust ant counts and search strategies in variable-strength combinatorial testing. Table 4 outlines all these studies.

5.2. The Developed Themes

5.2.1. Performance Analysis of Fuzzy-Based Metaheuristic Algorithms

The first theme discusses the performance of fuzzy-based metaheuristic algorithms in generating combinatorial test suites. As stated in the related works, metaheuristic-based combinatorial testing studies evaluate performance using five key indicators: test suite size, convergence speed, coverage efficiency, execution time, and scalability. Among these, the three selected fuzzy-based studies consistently focused on test suite size, convergence speed, and coverage efficiency. The Fuzzy Adaptive Teaching Learning-Based Optimization (FATLBO) strategy consistently reduced test suite size by 15-20% compared to the standard TLBO while ensuring complete interaction coverage, thereby confirming improvements in both test suite size and coverage efficiency [17].

The FATLBO strategy uses a fuzzy inference system to dynamically balance the exploration and exploitation phases of the original Teaching-Learning Based Optimization algorithm. FATLBO also achieved more efficient search behavior and avoided premature convergence. Similarly, the fuzzy adaptive swarm optimization approach outperformed conventional swarm algorithms [18].

Their results demonstrated faster convergence and coverage efficiency above 98%. Additionally, the Variable-Strength Tuned Ant Colony Optimization (VS-TACO) employed fuzzy logic to adapt ant counts and search strategies in variable-strength combinatorial testing [6]. Their study demonstrated a 12-18% reduction in test suite size compared to the baseline ACO. This suggests that these three indicators represent the most relevant measures for assessing the effectiveness of fuzzy-based metaheuristic algorithms in combinatorial testing.

Table 4. Background of selected studies

Year	Algorithm	Fuzzy/Adaptive Role	Application Focus	Main Outcomes
2017	Fuzzy Adaptive Teaching Learning-Based Optimization (FATLBO)	Fuzzy inference system dynamically adjusted exploration and exploitation in TLBO	Mixed-strength t-way test suite generation	Reduced test suite size (15−20%), improved convergence, and better coverage
2015	Fuzzy Adaptive Swarm Optimization	Fuzzy rules tuned swarm parameters (inertia weight, acceleration coefficients) in real time	Covering array construction for software testing	Faster convergence, higher coverage efficiency (above 98%), better balance between exploration and exploitation
2022	Variable-Strength Tuned Ant Colony Optimization (VS-TACO)	Fuzzy logic dynamically adjusted ant counts and search strategies	Variable-strength t-way test suite generation	Improved exploration-exploitation balance, reduced test suite size (12−18% reduction), and enhanced computational efficiency.

5.2.2. Types of Fuzzy-Based Metaheuristic Algorithms

The second theme explores the variety of fuzzy-based metaheuristic algorithms. The Fuzzy Adaptive Teaching Learning-Based Optimization (FATLBO) represented fuzzy rules and a population-based algorithm to lead the learning phases [17]. A fuzzy logic-based adaptive swarm optimization also exhibited that swarm intelligence paradigms can benefit from dynamic fuzzy control of sensitive parameters [18]. The Variable-Strength Tuned Ant Colony Optimization (VS-TACO) integrated fuzzy logic to dynamically adjust ant counts and search strategies for variable-strength combinatorial testing [6].

The main components of the fuzzy inference system include fuzzification, rule evaluation, and defuzzification. The system's parameters have been defined to include three input parameters: quality measure (Qm), intensification measure (Im), and diversification measure (Dm). Additionally, the system includes an output parameter, namely the amplitude. The selection of these parameters was based on the work in [45], which used similar parameters to control the algorithm's adaptive behavior. The quality measure (Qm) represents the quality of the candidate test case in terms of its coverage of interaction elements. However, the algorithm's intensification and diversification measures are derived from the Hamming distance results. While the quality and diversification measures of the algorithm ensure the generation of diverse candidate solutions, the intensification measure hastens convergence to the optimal solutions. These parameters represent the main dynamics of the optimization process.

5.2.3. Fuzzy Logic Implementation Techniques

The third theme concerns the incorporation of fuzzy logic into the chosen algorithms. Fuzzy logic was first introduced by Lotfi Zadeh in 1965. Fuzzy logic is a mathematical approach developed to deal with the imprecision and uncertainties associated with complex systems [44]. The degree of membership of an element x in a fuzzy set A is given by $\mu_A(x)$, where $\mu_A(x)$ varies from 0 (no membership) to 1 (full membership) [44]. Fuzzy logic is a powerful tool for modeling real-world systems with associated ambiguities and linguistic uncertainties [46].

Fuzzy sets and inference rules, together with the use of inertia weight and acceleration coefficients, were utilized in the FATLBO algorithm [17, 18]. In Variable-Strength Tuned Ant Colony Optimization (VS-TACO), fuzzy inference was utilized for the adjustment of the number of ants and the use of edge selection strategies during the search process [6]. From the above, it can be concluded that the application of fuzzy logic can be performed at different levels of the optimization process, i.e., high-level search control and parameter-specific adjustment, as also proved by other studies [12, 47]. For example, the FATLBO algorithm can be considered a high-level search control application, where the use of fuzzy inference rules controls the transition between the exploration and exploitation search phases of the optimization process [17]. In another study, the application of fuzzy rules was utilized for the dynamic adjustment of the inertia weight and acceleration coefficients of the swarm optimization process [18]. In the Variable-Strength Tuned Ant Colony Optimization (VS-TACO) algorithm, the application of fuzzy logic was utilized for the adjustment of the number of ants and the use of edge selection strategies, simultaneously controlling the global search process of the optimization process [6].

5.2.4. Parameter Tuning Strategies

The last theme deals with parameter tuning strategies using fuzzy logic [17]. Fuzzy logic was used in TLBO to balance exploration and exploitation. Fuzzy rules were also incorporated to tune swarm parameters in real-time [18]. Variable-Strength Tuned Ant Colony Optimization (VS-TACO) used fuzzy logic to adaptively tune the number of ants and edge selection strategies [6].

Table 5. Mapping of selected studies, identifying primary themes and sub-themes

Authors	Country	Performance analysis	Algorithm Type	Fuzzy Logic Implementation	Parameter Tuning Strategy
Zamli et al., 2017 [17]	Malaysia	Test suite size reduction, coverage efficiency	Teaching-Learning Based	High-level search control via fuzzy inference for exploration-exploitation balance	Adaptive adjustment of exploration-exploitation phases
Mahmoud & Ahmed, 2015 [18]	Egypt	Convergence speed, coverage efficiency	Swarm-Based	Parameter-specific adjustment of inertia weight and acceleration coefficients	Real-time fuzzy tuning of swarm parameters
Zahir Ahmad et al. [6]	Malaysia	Test suite size reduction, convergence speed, coverage efficiency	Ant Colony Optimization (ACO)	Combined approach: high-level control of search strategies and parameter-specific adjustment of ant counts and edge selection	Adaptive fuzzy regulation of ant numbers and edge selection

From studies on fuzzy logic in metaheuristics, parameter tuning was identified as the major aspect for incorporating fuzzy logic. For instance, in FATLBO, fuzzy inference was used to replace the necessity to set balances for exploration and exploitation during the learning phases [18].

Similarly, VS-TACO avoided fixed ant population settings by using fuzzy logic to adaptively regulate ant counts and edge selection strategies during the search.

To provide a clearer overview of how the selected studies addressed these themes, Table 5 presents a summary mapping each study to the identified sub-themes.

5.3. Summary of Performance Metrics and Evaluated Parameters

To ensure clarity in terminology across the synthesized literature, Table 6 summarizes the performance metrics and specific parameters quantified in each primary study.

Table 6. Metrics and parameters used in primary study

Study	Performance Metrics	Adaptive Parameter	Evaluation Context
FAPSO	Test Suite Size	Inertia Weight	t-way benchmarks
FATLBO	Test Suite Size	Teaching Factor	Fixed-strength covering array
VS-TACO	Tuple Coverage Percentage	Pheromone Evaporation Rate	Variable-strength covering array

5.4. Discussion

This review aims to synthesize current research on fuzzy-based metaheuristic algorithms for generating combinatorial test suites. In the related works, it was identified that while metaheuristic algorithms, such as GA, PSO, ACO, and TLBO, have been widely applied in combinatorial testing, their reliance on static parameter settings limits their adaptability and robustness. Fuzzy logic was highlighted as a potential solution, yet no prior work had systematically examined evidence of its application in combinatorial testing. To address this gap, the study employed a systematic methodology based on PRISMA and Kitchenham's framework, which narrowed an initial pool of 61 studies to three primary journal articles through identification, screening, eligibility assessment, quality appraisal, and data extraction.

With respect to RQ1 (types of fuzzy-based metaheuristic algorithms), the results presented in Section 4.2.2 show that fuzzy logic has been integrated into different categories of algorithms: a population-based teaching-learning approach (FATLBO), a swarm intelligence algorithm (fuzzy adaptive swarm optimization), and an ant colony optimization variant (VS-TACO). This diversity demonstrates that fuzzy logic is not restricted to a single algorithm but can enhance multiple metaheuristic paradigms, directly addressing the gap identified in related work, where no consolidated mapping of fuzzy-based CT strategies has been provided.

For RQ2 (fuzzy logic implementation techniques), Section 4.2.3 highlighted how fuzzy logic was embedded at different levels of the optimization process. FATLBO applied fuzzy inference as a high-level controller of the exploration-exploitation balance. Mahmoud and Ahmed implemented parameter-specific adjustments to the inertia weight and acceleration coefficients, while VS-TACO combined high-level control with these adjustments by adaptively tuning the ant counts and search strategies. For RQ3 (parameter tuning strategies), Section 4.2.4 confirmed that all three studies used fuzzy inference rules to replace fixed parameter values. The exploration and exploitation phases were dynamically balanced in TLBO, where swarm coefficients were tuned in real-time, and ant counts, along with edge selection strategies, were regulated in VS-TACO. These findings confirm the validity of related work demonstrating that fuzzy-based parameter tuning directly enhances robustness by responding to dynamic search conditions.

Regarding RQ4 (performance outcomes), Section 4.2.1 showed consistent improvements across three performance indicators. The findings revealed that FATLBO achieved notable reductions in test suite size, fuzzy adaptive swarm optimization resulted in faster convergence and improved coverage efficiency, while VS-TACO demonstrated enhancements in both test suite size and convergence speed.

In addressing RQ5 (limitations and research opportunities), this review found that although the three analyzed studies produced encouraging outcomes, their scope and quantity remain limited. In the broader body of research on metaheuristics and combinatorial testing, performance is typically evaluated using metrics such as test suite size, convergence speed, coverage efficiency, execution time, and scalability. However, the reviewed fuzzy-based studies consistently concentrated on only three of these: test suite size, convergence speed, and coverage efficiency, highlighting a gap and the need for broader performance assessment in future investigations.

This SLR therefore confirms that these three indicators are the most relevant performance measures for evaluating fuzzy-based metaheuristic strategies, and they will be adopted in our subsequent research to ensure consistency and comparability with existing work. Overall, the discussion shows that integrating fuzzy logic into metaheuristic algorithms effectively addresses the limitations identified in the literature review, is systematically supported by the research methodology, and is consistently validated across all four key themes. As noted in Section 4.2.1, although prior studies on metaheuristics and combinatorial testing commonly used five performance indicators- test suite size, convergence speed, coverage efficiency, execution time, and scalability- the three fuzzy-based studies reviewed here consistently focused on test suite size, convergence speed, and coverage efficiency. This consistent emphasis reinforces the importance of these three metrics for evaluating fuzzy-based metaheuristics and supports their selection as the main performance indicators for the subsequent phases of this research.

5.4.1. Practical Implications

The findings of this review provide several indications for both research and practice. For software testers, the findings suggest that fuzzy-based metaheuristic algorithms provide a practical way for generating smaller test suites while maintaining complete interaction coverage. Since the findings consistently demonstrate improvements in test suite size reduction, convergence speed, and coverage efficiency, these three aspects should be prioritized as key indicators when assessing future fuzzy-based or hybrid algorithms in practical applications. The way fuzzy logic has been utilized across different implementations, such as FATLBO, fuzzy adaptive swarm optimization, and VS-TACO, suggests promising opportunities for hybrid designs. Researchers can extend fuzzy integration to additional algorithmic paradigms and investigate multi-objective optimization.

5.4.2. Limitations of the Review

Several limitations have been acknowledged. First, only a small number of studies satisfied the inclusion and quality assessment criteria. As a result, only three primary journal articles were analyzed in depth.

Second, although all three studies consistently reported improvements in test suite size, convergence speed, and coverage efficiency, differences in their algorithm types, experimental settings, and evaluation metrics make it difficult to draw direct comparisons.

Third, this review specifically focused on journal publications from 2020 to 2024 to maintain relevance and quality, but this may have excluded earlier foundational research that could offer additional context. Fourth, since the review relied on three major databases, Scopus, IEEE Xplore, and ScienceDirect, it is possible that some relevant works in grey literature or other indexing platforms may not have been captured. Lastly, this review focused on fuzzy-based metaheuristics applied to combinatorial testing, so the conclusions may not be directly applicable to other areas of software engineering or optimization.

5.4.3. Comparative Technical Analysis: The Rationale for Fuzzy-Based Superiority

From the primary studies considered for the review, it has been clearly indicated that fuzzy-based metaheuristics have consistently performed better than static state-of-the-art methods. The rationale for the development of the algorithm, based upon the progression of techniques, has been based on the move away from rigid mathematical models towards more adaptive linguistic control systems, allowing the search process to be more adaptive to the search conditions. In the case of traditional metaheuristics, the search parameters, such as weights, coefficients, or step sizes, are usually controlled by linear or nonlinear decreasing functions. These static methods are based upon the assumption of a uniform search landscape for the search space, which is rarely the case for combinatorial testing.

Instead of these fixed functions, fuzzy-based methods use a Mamdani Fuzzy Inference System (FIS). This FIS evaluates real-time inputs and adjusts its parameters to meet the current requirements of the search process. This guarantees the algorithm provides large steps during the initial exploration and fine-tuned adjustments during the final exploitation phase. Modern-day metaheuristics are highly sensitive to their initial configurations. For example, an initial configuration suitable for $t = 2$ can be completely inappropriate for $t = 6$. Research literature has proven that fuzzy logic can greatly minimize such sensitivities. By employing problem-specific constraints in the form of linguistic rules, fuzzy-based hybrids ensure a very high robustness. This enables the generator to produce stable, high-quality test suites for different levels of interaction strength without requiring tedious, labor-intensive parameter retuning, as is typical with static methods.

6. Conclusion

This systematic literature review examines the role of fuzzy logic in generating combinatorial test suites. In this area, metaheuristic algorithms such as Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Sine Cosine Algorithm (SCA) have been utilized to build efficient combinatorial test suites. Conventional metaheuristic algorithms typically rely on fixed parameter configurations, limiting their ability to adapt to dynamic search conditions. Consequently, these algorithms are exposed to premature convergence within local optima. To address this issue, researchers have increasingly explored integrating fuzzy logic as an adaptive control mechanism that adjusts parameters in response to the developing state of the search process.

Although individual studies have shown promising results, the available evidence remains scattered and has not yet been systematically organized into a comprehensive body of knowledge. To address this gap, the present study employs a systematic review approach guided by the PRISMA framework and the methodological guidelines proposed by Barbara Kitchenham.

Commencing with an initial set of 61 potentially appropriate publications, a proper and transparent process was accomplished, including identification, screening, eligibility assessment, quality appraisal, and systematic data extraction.

This systematic procedure reduced the selection to three primary journal articles that met all inclusion criteria and quality standards for detailed analysis.

The results were categorized into four themes. First, the review confirmed that fuzzy logic has been integrated into multiple categories of metaheuristics, including teaching-learning based optimization (FATLBO), swarm-based optimization, and ant colony optimization (VS-TACO), thereby demonstrating its flexibility (RQ1). Second, the studies employed different fuzzy logic implementation techniques, ranging from high-level search control to parameter-specific adjustments, with VS-TACO combining both (RQ2). Third, fuzzy-based parameter tuning strategies successfully replaced static parameter values with adaptive rules, enabling real-time adjustments to exploration-exploitation balance (RQ3). Fourth, the performance outcomes across the three studies consistently showed improvements in test suite size reduction, convergence speed, and coverage efficiency (RQ4).

Finally, the review identifies key limitations, including the limited number of empirical studies, limited benchmarking, and the deficiency of industrial-scale validation. Addressing these deficiencies requires a shift toward more comprehensive experimental designs and the formulation of hybrid adaptive frameworks capable of satisfying the demands of complex, real-world software systems (RQ5).

In conclusion, this review explains that fuzzy-based metaheuristic algorithms introduce an encouraging direction for strengthening combinatorial testing. The justification of test suite size, convergence speed, and coverage efficiency as the three most relevant performance measures provides a strong basis for guiding future empirical work.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper. The research was funded by the UTHM Publisher's Office via Publication Fund E15216 and Centre for Graduate Studies (CGS). The funding body had no role in the design, execution, analysis, or interpretation of the study.

Funding Statement

Communication of this research is made possible through monetary assistance by the UTHM Publisher's Office via Publication Fund E15216 and the Centre for Graduate Studies (CGS).

Acknowledgments

The authors would like to thank the Publisher's Office and Centre for Graduate Studies (CGS) at Universiti Tun Hussein Onn Malaysia for its support. The authors confirm contribution to the paper as follows: study conception and design: Nur Syabila Zabidi, Noraini Ibrahim, and Asmau Osman; data collection: Nur Syabila Zabidi; analysis and interpretation of results: Nur Syabila Zabidi, Noraini Ibrahim, and Asmau Osman; draft manuscript preparation: Nur Syabila Zabidi; critical review and editing: Noraini Ibrahim and Asmau Osman. All authors reviewed the results and approved the final version of the manuscript.

References

[1] Mark Harman, Yue Jia, and Yuanyuan Zhang, "Achievements, Open Problems and Challenges for Search based Software Testing," *2015 IEEE 8th International Conference on Software Testing, Verification and Validation (ICST)*, Graz, Austria, pp. 1-12, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[2] Abhishek Singh Verma, Ankur Choudhary, and Shailesh Tiwari, "Software Test Case Generation Tools and Techniques: A Review," *International Journal of Mathematical, Engineering and Management Sciences*, vol. 8, no. 2, pp. 293-315, 2023. [CrossRef] [Google Scholar]

[3] Muhammad Afiq Bin Ariffin et al., "Efficient Test Case Prioritization using the Rat Swarm Optimizer Algorithm," *2024 IEEE 22nd Student Conference on Research and Development (SCOReD)*, Selangor, Malaysia, pp. 513-520, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[4] Braulio J. Solano-Rojas, Ricardo Villalón-Fonseca, and Rafael Batres, "Micro Evolutionary Particle Swarm Optimization (MEPSO): A New Modified Metaheuristic," *Systems and Soft Computing*, vol. 5, pp. 1-13, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] Pavi Saraswat, and Abhishek Singhal, "A Hybrid Approach for Test Case Prioritization and Optimization using Meta-Heuristics Techniques," *2016 1st India International Conference on Information Processing (IICIP)*, Delhi, India, pp. 1-6, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[6] Mohd Zamri Zahir Ahmad et al., "VS-TACO: A Tuned Version of Ant Colony Optimization for Generating Variable Strength Interaction in T-Way Testing Strategy," *Proceedings of the 2022 11th International Conference on Software and Computer Applications*, Association for Computing Machinery, New York, NY, United States, pp. 48-54, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] N. Ramli et al., "T-Way Test Suite Generation Strategy based on Ant Colony Algorithm to Support T-Way Variable Strength," *Journal of Physics: Conference Series*, vol. 1755, no. 1, pp. 1-13, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[8] Masrulehsan Mamat et al., "A Hybrid Sine Cosine Algorithm with Cluster Voting for Combinatorial Testing," *2024 1st International Conference on Cyber Security and Computing (CyberComp)*, Melaka, Malaysia, pp. 106-112, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[9] Amir Seyyedabbasi, "WOASCALF: A New Hybrid Whale Optimization Algorithm based on Sine Cosine Algorithm and Levy Flight to Solve Global Optimization Problems," *Advances in Engineering Software*, vol. 173, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[10] Manju Khari, and Prabhat Kumar, "RETRACTED ARTICLE: An Extensive Evaluation of Search-based Software Testing: A Review," *Soft Computing*, vol. 23, no. 6, pp. 1933-1946, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[11] Joshua E. Hernández-Suárez et al., "Search-based Software Testing Driven by Bioinspired Algorithms: A Systematic Literature Mapping," *2024 12th International Conference in Software Engineering Research and Innovation (CONISOFT)*, Puerto Escondido, Mexico, pp. 101-108, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[12] Changwu Huang, Yuanxiang Li, and Xin Yao, "A Survey of Automatic Parameter Tuning Methods for Metaheuristics," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 2, pp. 201-216, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[13] Shubham Gupta, "Enhanced Sine Cosine Algorithm with Crossover: A Comparative Study and Empirical Analysis," *Expert Systems with Applications*, vol. 198, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[14] Aasam Abdul Karim, Nor Ashidi Mat Isa, and Wei Hong Lim, "Hovering Swarm Particle Swarm Optimization," *IEEE Access*, vol. 9, pp. 115719-115749, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[15] Aasam Abdul Karim, Nor Ashidi Mat Isa, and Wei Hong Lim, "Modified Particle Swarm Optimization with Effective Guides," *IEEE Access*, vol. 8, pp. 188699-188725, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[16] Aminu Aminu Muazu, Ahmad Sobri Hashim, and Aliza Sarlan, "Application and Adjustment of 'Don't Care' Values in T-Way Testing Techniques for Generating an Optimal Test Suite," *Journal of Advances in Information Technology*, vol. 13, no. 4, pp. 347-357, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[17] Kamal Z. Zamli et al., "Fuzzy Adaptive Teaching Learning-based Optimization Strategy for the Problem of Generating Mixed Strength T-Way Test Suites," *Engineering Applications of Artificial Intelligence*, vol. 59, pp. 35-50, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[18] Thair Mahmoud, and Bestoun S. Ahmed, "An Efficient Strategy for Covering Array Construction with Fuzzy Logic-based Adaptive Swarm Optimization for Software Testing use," *Expert Systems with Applications*, vol. 42, no. 22, pp. 8753-8765, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[19] Caroline Heins, "Artificial Intelligence in Retail - A Systematic Literature Review," *Foresight*, vol. 25, no. 2, pp. 264-286, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[20] Ariffin Abdullah, Rohayanti Hassan, and Zuraini Ali Shah, "A Systematic Literature Review of Combinatorial Testing," *International Journal of Advance in Soft Computing its Application*, vol. 9, no. 2, pp. 128-139, 2017. [Google Scholar]

[21] Alam Zeb et al., "A Systematic Literature Review on Robust Swarm Intelligence Algorithms in Search-based Software Engineering," *Complexity*, vol. 2023, no. 1, pp. 1-22, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[22] Md. Abdul Kader, Kamal Z. Zamli, and Bestoun S. Ahmed, "A Systematic Review on Emperor Penguin Optimizer," *Neural Computing and Applications*, vol. 33, no. 23, pp. 15933-15953, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[23] Hayrol Azril Mohamed Shaffril, Asnarulkhadi Abu Samah, and Raidah Mazuki, "A Systematic Literature Review on the Adaptation of Women in Fisheries-based Families on Climate Change Impacts," *Journal of Environmental Studies and Sciences*, vol. 15, no. 3, pp. 648-665, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[24] Derek Cabrera, Laura Cabrera, and Derek Cabrera, "The Steps to Doing a Systems Literature Review (SLR)," *Journal of Systems Thinking Preprints*, pp. 1-28, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[25] Priti Bansal et al., "Construction of Variable Strength Covering Array for Combinatorial Testing using a Greedy Approach to Genetic Algorithm," *E-Informatica Software Engineering Journal*, vol. 9, no. 1, pp. 87-105, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[26] Xu Guo et al., "An Effective Approach to High Strength Covering Array Generation in Combinatorial Testing," *IEEE Transactions on Software Engineering*, vol. 49, no. 10, pp. 4566-4593, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[27] Md. Abdul Kader, Jamal A. Jamaluddin, and Kamal Z. Zamli, "An Educational Tool Aimed at Learning Metaheuristics," *Proceedings of the 2020 9th International Conference on Software and Computer Applications*, Association for Computing Machinery, New York, NY, United States, pp. 79-83, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[28] Asma Benmessaoud Gabis et al., "A Comprehensive Survey of Sine Cosine Algorithm: Variants and Applications," *Artificial Intelligence Review*, vol. 54, no. 7, pp. 5469-5540, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[29] Jyotheesh Gaddam, "*Development of Self-Adaptive Hybrid Algorithms : A Hyper-Heuristic Approach to Dynamic Optimisation*," Thesis, Deakin University, 2023. [Google Scholar] [Publisher Link]

[30] Qusay Shihab Hamad et al., "Q-Learning Embedded Sine Cosine Algorithm (QLESCA)," *Expert Systems with Applications*, vol. 193, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[31] Abdelaziz A. Abdelhamid et al., "Innovative Feature Selection Method based on Hybrid Sine Cosine and Dipper Throated Optimization Algorithms," *IEEE Access*, vol. 11, pp. 79750-79776, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[32] Hussein Almulla, and Gregory Gay, "Learning how to Search: Generating Effective Test Cases Through Adaptive Fitness Function Selection," *Empirical Software Engineering*, vol. 27, no. 2, pp. 1-62, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[33] Hasitha R. Jayetileke, W.R. De Mel, and Subhas Chandra Mukhopadhyay, "Real-Time Metaheuristic Algorithm for Dynamic Fuzzification, De-Fuzzification and Fuzzy Reasoning Processes," *Applied Science*, vol. 12, no. 16, pp. 1-38, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[34] David Moher et al., "Preferred Reporting items for Systematic Reviews and Meta-Analyses: The PRISMA Statement," *Annals of Internal Medicine*, vol. 151, no. 4, pp. 264-269, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[35] Matthew J Page et al., "The PRISMA 2020 Statement: An Updated Guideline for Reporting Systematic Reviews," *BMJ*, vol. 372, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[36] Barbara Kitchenham, and Stuart Charters, "*Guidelines for Performing Systematic Literature Reviews in Software Engineering*," Keele University, UK, 2007. [Google Scholar]

[37] Craig Lockwood et al., "Qualitative Research Synthesis: Methodological Guidance for Systematic Reviewers Utilizing Meta-Aggregation," *International Journal of Evidence-based Healthcare*, vol. 13, no. 3, pp. 179-187, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[38] Anas Abouzahra, Ayoub Sabraoui, and Karim Afdel, "Model Composition in Model Driven Engineering: A Systematic Literature Review," *Information and Software Technology*, vol. 125, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[39] Brett Stevens, and Eric Mendelsohn, "Efficient Software Testing Protocols," *Proceedings of the 1998 Conference of the Centre for Advanced Studies on Collaborative Research*, IBM Press, IBM Canada, NRC, 1998. [Google Scholar] [Publisher Link]

[40] C. Yilmaz, M.B. Cohen, and A.A. Porter, "Covering Arrays for Efficient Fault Characterization in Complex Configuration Spaces," *IEEE Transactions on Software Engineering*, vol. 32, no. 1, pp. 20-34, 2006. [CrossRef] [Google Scholar] [Publisher Link]

[41] Charles J. Colbourn, "Strength Two Covering Arrays: Existence Tables and Projection," *Discrete Mathematics*, vol. 308, no. 5-6, pp. 772-786, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[42] Naseer Sabri et al., "Fuzzy Inference System: Short Review and Design," *International Review of Automatic Control*, vol. 6, no. 4, pp. 441-449, 2013. [Google Scholar]

[43] Erdal Kayacan, and Mojtaba Ahmadieh Khanesar, "Fundamentals of Type-2 Fuzzy Logic Theory," *Fuzzy Neural Networks Real Time Control Appl*, pp. 13-24, 2016. [Google Scholar]

[44] L.A. Zadeh, "Fuzzy Logic= Computing with Words," *IEEE Transactions on Fuzzy Systems*, vol. 4, no. 2, pp. 103-111, 1996. [CrossRef] [Google Scholar] [Publisher Link]

[45] Fakhrud Din et al., "Fuzzy Adaptive Teaching Learning-based Optimization for Solving Unconstrained Numerical Optimization Problems," *Mathematical Problems in Engineering*, vol. 2022, no. 1, pp. 1-17, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[46] Ezgi Erturk, and Ebru Akcapinar Sezer, "Software Fault Prediction using Mamdani Type Fuzzy Inference System," *International Journal of Data Analysis Techniques and Strategies*, vol. 8, no. 1, pp. 14-28, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[47] Stefan Lessmann, Marco Caserta, and Idel Montalvo Arango, "Tuning Metaheuristics: A Data Mining based Approach for Particle Swarm Optimization," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12826-12838, 2011. [CrossRef] [Google Scholar] [Publisher Link]