Middle East Journal of Applied Science & Technology (MEJAST)
Volume 9, Issue 3, Pages 18-38, July-September 2026

Hybrid Salp Swarm-Genetic Algorithm Optimization for the Multidimensional Knapsack Problem: A Conceptual Review and Framework Synthesis

Asaju La'aro Bolaji[1], Sanfo Bala[2], Andrew Ishaku Wreford[3] & Anagu Emmanuel John[4*]

[1,2,3,4]Department of Computer Science, Federal University Wukari, Taraba State, Nigeria.
Corresponding Author Email: anague@fuwukari.edu.ng[*]

DOI: Under Assignment

Article Received: 23 May 2026 Article Accepted: 27 July 2026 Article Published: 12 August 2026

ABSTRACT

The Multidimensional Knapsack Problem (MKP) is a classical NP-hard combinatorial optimization problem used in wide variety of applications such as in logistics, cloud computing, manufacturing, telecommunications, scheduling and resource allocation. Metaheuristic algorithms are widely used because as the size of the problem and complexity of the optimization problem grows, the traditional exact methods are not able to compute them. In this regard, the Genetic Algorithm (GA) and Salp Swarm Algorithm (SSA) have received high interest due to their complementary search capabilities. The global exploration via the adaptive leader–follower mechanism in SSA, and the strong local exploitation by evolutionary operators in GA, complement each other well. In recent years, these algorithms are being incorporated into hybrid frameworks to accelerate the convergence process, preserve the diversity of the population and increase the quality of the solutions in large-scale optimization problems. But most of the current research is implementation oriented, and there are very few conceptual syntheses of concepts for the theoretical foundations, evolution, hybridization strategies and emerging developments of SSA–GA optimization for MKP. This paper discusses the complete overview of the conceptual review of Hybrid Salp Swarm–Genetic Algorithm optimization in Multidimensional Knapsack Problem. It outlines the development of the MKP, metaheuristic optimization, evolutionary computation, swarm intelligence and hybrid optimization and discusses the complementary nature of exploring/exploiting, constraint-handling and adaptive optimization mechanisms. The review also outlines the recent research trends, conceptual gaps, and suggests a common framework to inform the design of the scalable, adaptive, and computationally efficient hybrid optimization models. By consolidating current knowledge and outlining future research directions, this review provides a valuable reference for researchers and practitioners working in combinatorial optimization and intelligent resource allocation.

Keywords: Multidimensional Knapsack Problem; Genetic Algorithm; Salp Swarm Algorithm; Resource Allocation.

1.0. Introduction

The complexity of decision making today has grown, and the challenge to optimize (make efficient use of) limited resources with multiple constraints has become more profound. These are frequently encountered in logistics, transportation, manufacturing, cloud computing, healthcare, telecommunication, finance and supply chain management systems, where an organisation aims to get the most for their budget, capacity or processing power. With increasing size and dimensionality of these problems, traditional optimization methods are becoming too cumbersome, so a need of scalable solution methods arises (Cacchiani et al., 2022; Li et al., 2024).

The Multidimensional Knapsack Problem (MKP) is one of the most popular combinatorial optimization problems, extending the 0-1 Knapsack Problem in order to model multiple resource constraints at the same time. The MKP is a more appropriate formulation than the single-constraint problem to represent an environment in which multiple constraints need to be met simultaneously, which is often encountered in real-world decision-making scenarios. Since it is NP-hard, the number of computations needed to find the optimal solution grows exponentially with the number of elements in the problem; therefore, the deterministic methods are impractical for medium and large-scale applications (Martello & Toth, 1990; Bushaj & Büyüktahtakın, 2024).

To overcome these drawbacks, researchers have turned to the use of metaheuristic optimization algorithms that are able to obtain near-optimal solutions of good quality in limited computation time. Metaheuristics are related to other methods of optimization which do not require a complete search of the solution space, such as exact optimisation methods, and use stochastic search procedures to efficiently navigate in large solution spaces. In the last 30 years several algorithms based on swarm behaviour and biological evolution have proven to be very effective for solving complex optimization problems (Blum & Roli, 2003; Talbi, 2009).

The Genetic Algorithm (GA) is one of the most known optimization algorithms among the evolutionary methods. GA is inspired by the process of natural selection, and candidate solutions are generated by selection, crossover, and mutation to try to find good candidate solutions in the search space, especially in regions that are promising. In certain optimization problems where dimensions are large, however, early convergence and a loss of population diversity are possible problems that GA could face (Goldberg 1989; Holland 1975).

On the other hand, Salp Swarm Algorithm (SSA) proposed by Mirjalili et al. (2017) is a competitive swarm intelligence method with an adaptive leader–follower search mechanism which not only has a simple computation but also improves the global exploration ability. While SSA has a great exploration feature, the performance of exploitation locally may reduce in the last phase of optimization. In this way, many studies are recent that have focused on hybrid optimization frameworks that combine the strengths of exploration of SSA with the strengths of exploitation of GA for better convergence, solution quality, and robustness (Mirjalili et al., 2017; Abdel-Basset et al., 2023).

Although the hybrid metaheuristics area has been the subject of several studies, most of the research existing in the literature is very much implementation-based and focused on testing the algorithm performance rather than conceptual development. The theoretical evolution of the MKP, the complementary role between GA and SSA, and the principles that underlie their hybridization are limitedly synthesized. This extends into the need to fill a gap to support future work in scalable and adaptive optimization frameworks.

The Multidimensional Knapsack Problem has been studied with the application of the Genetic Algorithms and Salp Swarm Algorithms among others, however most of the studies addressed issues of benchmark evaluation, algorithm implementation and performance comparison. Only a few studies offer a thorough conceptual synthesis of the theoretical development of the MKP, the complementary nature of GA and SSA and the motivation for including them in hybrid optimization processes. Moreover, new advances like adaptive parameter control, machine learning based optimization and intelligent hybrid metaheuristics have been poorly discussed in the previously conceptual reviews. Thus, there is a need for a review that incorporates the above developments and presents them in a unified and cohesive framework that can facilitate further research and algorithm design.

A detailed conceptual review of hybrid Salp Swarm–Genetic Algorithm optimization for Multidimensional Knapsack Problem is presented in this paper, including its theoretical background, hybridization methods, latest advancement and future research directions.

░ 2.0. Research Methodology

The study used the structured conceptual review approach to synthesize the current knowledge of hybrid Salp Swarm – Genetic Algorithm (SSA – GA) optimization method for solving the Multidimensional Knapsack Problem (MKP). A conceptual review differs from a systematic review in that it focuses on theoretical developments, methodological advances, and emerging research trends in a research domain so as to give a general understanding of the domain (Snyder, 2019).

The literature was searched extensively from Scopus, Web of Science, IEEE Xplore, ScienceDirect, SpringerLink, ACM Digital Library and Google Scholar for relevant literature. The search was conducted on publications from the year 2015 to 2026 to cover the latest developments in hybrid metaheuristic optimization, and to ensure that some seminal papers that laid the theoretical foundations of the Genetic Algorithm, swarm intelligence and the Salp Swarm Algorithm were included. The keywords used were Multidimensional Knapsack Problem, Genetic Algorithm, Salp Swarm Algorithm, Hybrid Metaheuristics, Swarm Intelligence, Evolutionary Computation, Combinatorial Optimization, Resource Allocation.

Only peer-reviewed journal articles, conference proceedings and scholarly books were included if they mentioned the MKP, Genetic Algorithms, Salp Swarm Algorithms, hybrid metaheuristics or other similar optimization methods. Any publications related to the field, considered to be seminal, have been kept, regardless of year of publication. The editorials, opinion papers, duplicate records, and study papers which are not related to combinatorial optimization and hybrid metaheuristic algorithms were excluded.

Thematic synthesis was used to analyse the selected literature. Instead of summarizing the results from a statistical perspective, the findings were organized under the main conceptual themes (i) evolution of the Multidimensional Knapsack Problem, (ii) metaheuristic optimization, (iii) Genetic Algorithms, (iv) swarm intelligence and the Salp Swarm Algorithm, (v) hybrid SSA–GA optimization, and (vi) emerging research trends and future directions. This facilitated the identification of common theoretical principles, complementary optimization mechanisms and unsolved research problems in the literature.

The review forms a forward-looking integration of the past studies and recent developments, offering a holistic view of the evolution of hybrid SSA–GA optimization and suggesting avenues to create more adaptable, scalable and computationally efficient optimization structure for the Multidimensional Knapsack Problem.

3.0. Conceptual Review

3.1. The Multidimensional Knapsack Problem

The Multidimensional Knapsack Problem (MKP) is one of the most widely studied NP-hard combinatorial optimization problems in Operations Research (OR), since it is a good model of decision-making situations where limited resources must be allocated among competing alternatives. It is an extension of the classical 0–1 Knapsack Problem where several resource constraints exist while the objective is again to maximize the total profit or utility. In contrast, this extension allows the MKP to model more realistic optimization scenarios, where multiple capacity constraints come into play, such as financial budgets, storage capacities, energy consumption, processing resources, or time constraints (Martello & Toth, 1990; Cacchiani et al., 2022).

Mathematically, the MKP aims to find an optimal subset of items that maximises the total objective value, without exceeding any of the resource constraints. It is an NP-hard problem, which means that for medium and large-scale instances, exact solving is computationally very demanding because the number of feasible solutions grows

Middle East Journal of Applied Science & Technology (MEJAST)
Volume 9, Issue 3, Pages 18-38, July-September 2026

exponentially with the number of decision variables (Kellerer et al., 2004). As a result, the MKP is now a test problem for comparing and assessing optimization algorithms with their capacity to deliver a compromise between solution quality and computational efficiency.

The MKP has many application domains, all of which are very practical. It is used in logistics and transportation for loading vehicles and allocating cargo. It is used in the placement of virtual machines and resource scheduling in cloud computing. The MKP is utilized in manufacturing systems for production planning and capacity allocation, and in finance for selecting a portfolio within a budget constraint. Other applications involve wireless sensor networks, telecommunications, healthcare resource planning, and supply chain optimization, where multiple operational constraints must be met simultaneously (Bushaj & Büyüktahtakın, 2024; Li et al., 2024).

Early research was focused on the MKP and applied exact optimization methods like Dynamic Programming, Branch-and-Bound and Integer Linear Programming, which provide optimal solutions for fairly small problems. Their usefulness is restricted to large-scale optimization problems, however, due to an exponential increase of the search space. As the complexity of the problems grew, researchers turned to heuristic and metaheuristic methods that could find near-optimal solutions within a reasonable timeframe to compute (Martello & Toth, 1990; Dellinger et al., 2022).

Modern studies have broadened the conceptions of the MKP. In addition to the traditional deterministic formulation, recent studies have also included consideration of dynamic environments, uncertainty, stochastic parameters, multi-objective optimization, and sustainability. These advances are associated with the need for optimization models that will be able to assist in intelligent decision-making in a rapidly changing and data-intensive environment. In this way, the MKP has become a powerful tool to model complex resource allocation problems for various scientific and industrial applications and is not just a classical mathematical problem anymore (Scherer et al., 2024).

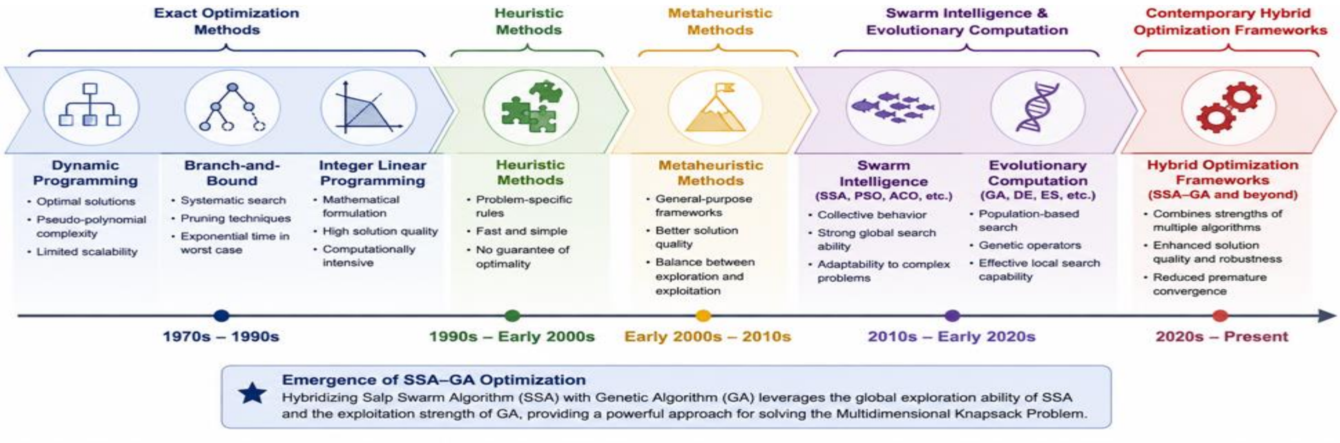

Figure 1. Evolution of Optimization Approaches for the Multidimensional Knapsack Problem

The computational difficulty of the MKP has also shaped the development of optimization techniques. Although some algorithms are exact, they are usually impractical to use in large-scale real-world problems where there are thousands of decision variables and many constraints. This restriction has spurred the growth of metaheuristic algorithms focusing on high-quality solutions with efficiency in mind. Consequently, evolutionary computation and swarm intelligence have emerged as the main paradigms for the MKP solution, with some hybrid optimizers like the Salp Swarm–Genetic Algorithm (SSA–GA) relying on these approaches.

Overall, the MKP is a good benchmark for combinatorial optimization because it exhibits the key features of resource allocation problems. Its computational complexity has given impetus to optimization research and spurred the development of more adaptive and intelligent algorithms that can account for trade-offs between exploration/exploitation and computation.

3.2. Evolution of Metaheuristic Optimization

Heuristic and metaheuristic approaches are gradually supplanting exact methods to solve combinatorial optimization problems as they become increasingly complex. The early attempts for optimization were based on deterministic methods like Dynamic Programming, Integer Linear Programming and Branch-and-Bound methods that provide optimal solutions under well formulated mathematical conditions. While they work well for small-scale optimization problems, they become too time-consuming for large-scale problems like the Multidimensional Knapsack Problem (MKP) (Kellerer et al., 2004; Dellinger et al., 2022) which are NP-hard.

To overcome these limitations, heuristic methods were introduced to obtain satisfactory solutions in reasonable time. Applying greedy algorithms, local search, hill climbing, and tabu search decreased computational complexity considerably, but these methods were problem specific, and often were easily stuck in local optima. The fact that their adaptability was limited to different optimization domains spurred the development of more flexible search frameworks (Talbi, 2009). Metaheuristic optimization was born as a generic solution that could solve a variety of optimization problems with stochastic search techniques. Metaheuristics do not depend on the problem and need only the evaluations of the objective function, without the detailed mathematical formulation of the problem as is in the case of heuristics. They use iterative improvement mechanisms which involve exploring the solution space to find new regions of it, and exploiting the regions to improve upon promising solutions. It is generally accepted that optimization performance relies mostly on achieving an adequate tradeoff between these complementary processes (Blum & Roli, 2003; Boussaïd et al., 2013).

Inspired by these 4 areas, modern metaheuristics can be broadly categorized:There are four general categories of modern metaheuristics that are inspired by these 4 sources:

Evolutionary algorithms such as the Genetic Algorithm (GA), Differential Evolution (DE) and Evolution Strategies (ES) mimic natural selection and biological evolution to improve candidate solutions.

Some algorithms that mimic the collective behaviour of biological systems are called swarm intelligence algorithms, including Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Grey Wolf Optimizer (GWO), Whale Optimization Algorithm (WOA) and the Salp Swarm Algorithm (SSA).

Physics-inspired algorithms such as Simulated Annealing and Gravitational Search Algorithm simulate the nature.

Human-inspired algorithms include algorithms inspired by learning systems such as Teaching–Learning Based Optimization (TLBO) and social decision-making processes such as Imperialist Competitive Algorithm (ICA) (Yang, 2020).

Of these, evolutionary computation and swarm intelligence are the dominant paradigms for solving NP-hard combinatorial optimization problems because they offer a good balance between solution quality and computational efficiency. Population-based search allows several candidate solutions to be evolved at the same time and provides more diversity, hence less risk of premature convergence than single solution approaches (Poli et al., 2007).

Although the success, there is no one metaheuristic that is always the best performer for all optimization problems. The No Free Lunch (NFL) Theorem formalizes this observation by claiming that optimization algorithms perform differently depending on the problem in which they are applied (Wolpert & Macready, 1997). Recently, there has been a growing interest in hybrid metaheuristics in which complementary algorithms are used together, benefiting from the strengths and reducing the weaknesses of each algorithm.

One of the most promising developments is the addition of swarm intelligence to evolutionary algorithms. The Genetic Algorithm has been shown to excel at exploitation, while the Salp Swarm Algorithm is effective at global exploration which leads to a more well-balanced search process. This results in better convergence speed, population diversity, and solution quality for high-dimensional optimization problems (Abdel-Basset et al., 2023; Castelli et al., 2022).

In addition, recent progress has further enhanced the capabilities of metaheuristic optimization by integrating adaptive parameter control, reinforcement learning, a machine learning-based search, surrogate modelling, and explainable artificial intelligence. These advancements allow optimization algorithms to adapt their search strategies in response to changes in the problem, leading to more robust, scalable, and efficient optimization systems for ever more complex decision-making contexts (Lu et al., 2024; Li et al., 2024).

A general overall view of the development of metaheuristic optimization indicates the shift from the deterministic mathematical programming to adaptive and intelligent search procedures, which are able to solve the large-scale combinatorial optimization problem. The evolution has laid the ground preparation for hybrid optimization methods, such as combining the Genetic Algorithms with the Salp Swarm Algorithm on the Multidimensional Knapsack Problem.

3.3. Genetic Algorithm

The GA is one of the old and most influential evolutionary optimisation methods that was developed to solve complex optimisation problems using a mechanism inspired by natural selection and biological evolution. Holland (1975) first suggested GA and Goldberg (1989) popularized it; it is so flexible, robust, and powerful that it has become a standard method for optimization without using gradients or mathematical formulations specific to the problem.

The basic idea behind GA is to use the principle of survival of the fittest to iteratively evolve a population of candidate solutions to progressively improve the solution. The solutions, or potential solutions, provided by the candidates are called chromosomes. Over the course of generations, chromosomes are selected, crossed over and mutated, creating new offspring with selected desirable traits from parents. Better-fitted candidate solutions have higher chances of surviving and then their children are more likely to survive, and so on, until the quality of the population on average improves over time (Goldberg, 1989).

Usually, a population is randomly initialized and the optimization process starts. The fitness of each chromosome is computed based on a pre-defined fitness function, which is used to quantify or qualify how well a solution is, according to the optimization objective. Selection operators are then used to find high-performing individuals for reproduction and crossover is used to mix the genetic material from the parent chromosomes to produce new candidate solutions.

To maintain the diversity of the population and to prevent too early convergence to the local optima, the mutation is used to introduce controlled randomness by making changes on selected genes. This evolutionary process is repeated until a stopping criterion is met (e.g., a maximum number of generations or a convergence goal).

An important asset of GA is its capability of solving global optimization problems in the case of nonlinear search spaces. Population-based search allows for the simultaneous exploration of multiple areas of the solution space, thereby boosting the chances of finding a quality solution to the problem as compared to single solution optimization methods. Moreover, GA is very flexible since it can be used for solving continuous, discrete, and combinatorial optimization problems with only a few modifications (Talbi, 2009).

Even with these benefits, performance of GA is very much sensitive to the settings used for the population size, crossover probability, mutation rate and selection procedure. If the parameters are not set well, the convergence speed of the algorithm could be slow or it might converge to the locally optimal solutions before it has explored the search space sufficiently. So, a good balance between exploration and exploitation is still a significant challenge in GA-based optimization (Eiben & Smith, 2015).

These restrictions have inspired significant work in adaptive and hybrid evolutionary algorithms. Adaptive Genetic Algorithms dynamically tune the parameters of the evolution process during optimization, and hybrid approaches involve using GA together with other optimization techniques that help GA to explore or exploit better. The aim of such integrations is to improve convergence speed, preserve solution diversity and to increase the accuracy of the optimization solution, especially for large-scale NP-hard problems like the Multidimensional Knapsack Problem (Abdel-Basset et al., 2023).

GA has been successfully implemented in many fields of optimization such as scheduling, feature selection, engineering design, supply chain optimization, cloud resource allocation, wireless sensor networks, portfolio optimization and combinatorial optimization. It is versatile and has one of the best exploitation algorithms in use both in academia and industry. For the MKP, GA is an efficient exploration tool that can be used to further improve solutions within a feasible search space, especially after promising search regions have been found.

But as exploitation is its hallmark, in later evolutionary stages, search diversity may lessen, making it harder to escape from local optima, which may lead to a drop in the value of GA. In recent years, the research of optimization techniques has increasingly integrated the GA with swarm intelligence algorithms with better global exploration ability. Of these, the Salp Swarm Algorithm (SSA) is one of the more promising algorithms that has been

Middle East Journal of Applied Science & Technology (MEJAST)
Volume 9, Issue 3, Pages 18-38, July-September 2026

developed because it uses adaptive leader–follower search mechanism and highly explores complex solution spaces.

Although the Genetic Algorithm provides efficient exploitation through evolutionary operators, its performance may deteriorate when population diversity declines or when optimization becomes trapped in local optima. Swarm intelligence algorithms address these challenges by emphasizing collective exploration and adaptive search behaviour. Among these, the Salp Swarm Algorithm has gained considerable attention because of its simplicity, fast convergence, and effective global search capability.

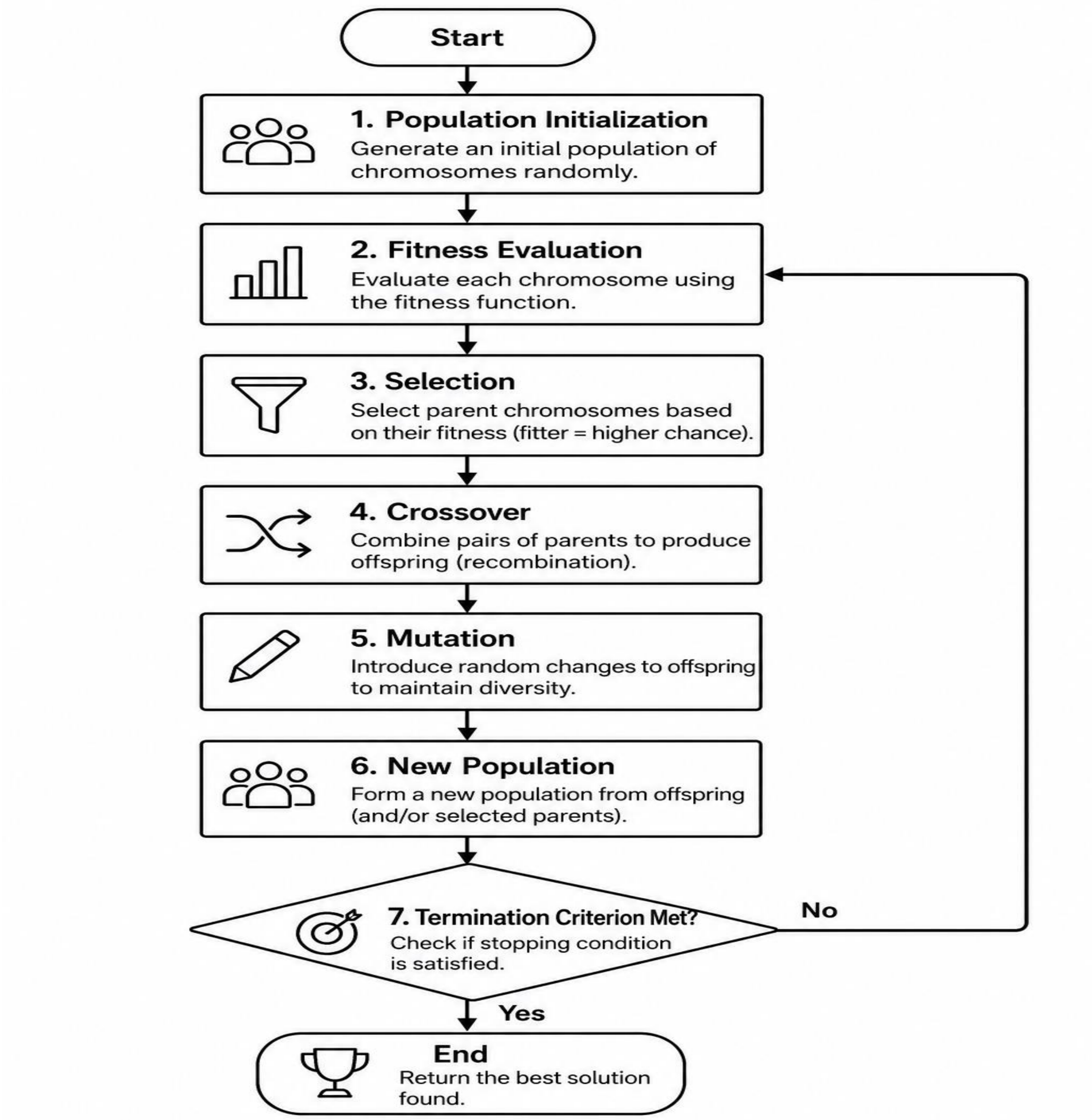

Figure 2. General Workflow of the Genetic Algorithm.

Although the Genetic Algorithm provides efficient exploitation through evolutionary operators, its performance may deteriorate when population diversity declines or when optimization becomes trapped in local optima. Swarm intelligence algorithms address these challenges by emphasizing collective exploration and adaptive search behaviour. Among these, the Salp Swarm Algorithm has gained considerable attention because of its simplicity, fast convergence, and effective global search capability.

3.4. Salp Swarm Algorithm

The Salp Swarm Algorithm (SSA) is a population-based optimization algorithm from swarm intelligence that has been introduced by Mirjalili et al. (2017). The algorithm is based on the swarming behaviour of salps, an organism that is transparent and lives in the oceans and when moving as a swarm creates a chain of moving salps. When moving, a salp swim toward food with the other salps adjusting their position in relation to the salp in front of them. The framework of this coordinated mechanism of leadership and followership is the foundation of SSA's search strategy, which allows an efficient trade-off between global exploration and local exploitation.

In contrast to many swarm intelligence algorithms which use complex mathematical operators, SSA uses a relatively simple search mechanism. The population is divided into two groups: leaders and followers. The leader moves towards the food source (best solution found so far) and the follower salps adjust their position based on the Newtonian principles of motion, incorporating the movement of the salps that have gone before. The swarm's positions are iterated many times until the swarm is gradually drawn towards good solutions; however, the swarm is still able to coordinate its exploration in the search space (Mirjalili et al., 2017).

The optimization process starts with randomly generating an initial population of salps. The candidate solution corresponds to each of the salps and the quality of the solution is checked with a previously defined objective function. The algorithm will determine which individual is the best performer and that individual is the current food source. In each step, the learner moves towards the food source with adaptive learning parameters controlling exploration and exploitation. Followers then adjust their positions based on the movement of the other salps, which forms a chain of searches, allowing the spread of information through the population. The iterations are repeated until a stopping criterion is met, e.g., until the maximum number of iterations has been reached or until the difference between iterates becomes less than a specified tolerance.

An important difference of SSA is its adaptive exploration mechanism. The algorithm makes a lot of global explorations at the beginning of the optimization process, then it is doing more local exploitation in later iterations, due to a time-varying control parameter, which is gradually lowered. This adaptive behaviour is able to decrease the chances of premature convergence of the algorithm and hence increase the chances of finding the promising region in the search space (Mirjalili et al., 2017).

SSA has shown competitive performance in a wide variety of problems – such as feature selection, engineering design, image processing, wireless sensor networks, power system optimization, cloud computing, medical diagnosis, scheduling, and resource allocation problems. Its relatively small number of control parameters, its simplicity of computation, and its rapid convergence make it very attractive to solve large-scale optimization problems in which computational efficiency is crucial (Abdel-Basset et al., 2023; Li et al., 2024).

However, SSA has some drawbacks. With the advancement of optimization, diversity of the population can diminish, which makes the algorithm less capable of conducting extensive local search around good solutions. This may lead to slower convergence at the end of optimization or to less accurate solutions if the search space is high dimensional. Moreover, a proper mechanism to preserve diversity should be added to SSA to avoid instability in highly constrained or multimodal optimization problems, as many swarm intelligence algorithms do, otherwise (Castelli et al., 2022).

To overcome these shortcomings, many improved models of SSA have been proposed, such as adaptive SSA, chaotic SSA, opposition-based SSA, Lévy flight SSA, multi-population SSA, and hybrid SSA frameworks. The changes were designed to enhance the convergence speed, to boost exploitation ability, to preserve population diversity and to boost robustness over various optimization areas. Of these, hybridization with the Genetic Algorithm (GA) has proven to be one of the most successful approaches due to the powerful global exploration ability of SSA and the powerful local exploitation ability of GA. This complementary integration has led to more even search behaviour and has resulted in better performance on several NP-hard combinatorial optimization problems, like the Multidimensional Knapsack Problem.

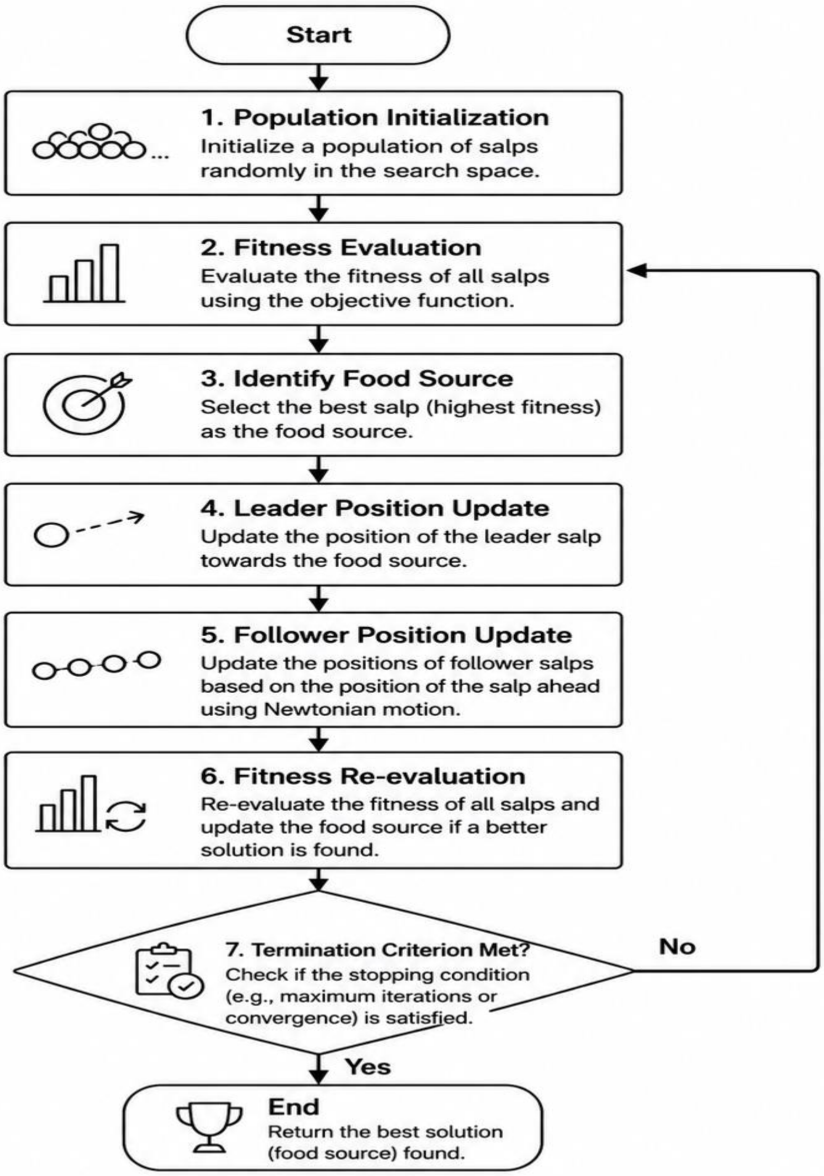

Figure 3. Flowchart of the Salp Swarm Algorithm.

In conclusion, the Salp Swarm Algorithm is a promising development in the field of swarm intelligence optimization. The adaptive search strategy of leader–follower, its computational efficiency, and flexibility have made it an effective and competitive optimization technique. However, it has the best potential in hybrid optimization frameworks that use its exploration ability in combination with the exploitation ability of evolutionary algorithms like the Genetic Algorithm.

The complementary strengths of the Genetic Algorithm and the Salp Swarm Algorithm have motivated increasing interest in hybrid optimization frameworks. While GA excels at refining high-quality solutions through evolutionary operators, SSA effectively explores diverse regions of the search space using its adaptive leader–follower mechanism. Integrating these algorithms provides a balanced exploration–exploitation strategy that enhances convergence, solution quality, and robustness. The following section therefore examines the theoretical foundations, hybridization strategies, and optimization mechanisms of hybrid SSA–GA approaches for the Multidimensional Knapsack Problem.

3.5. Hybrid Salp Swarm–Genetic Algorithm Optimization

Combining optimization algorithms has proven to be a good approach to complement the shortcomings of single algorithms, a process often referred to as hybrid metaheuristic optimization. A hybrid approach is a combination of different search methods to improve the quality of solutions, convergence rate, robustness and computational efficiency. It is very useful for solving problems that can be formulated as an NP hard combinatorial optimization problem like Multidimensional Knapsack Problem (MKP) which are characterized by a proper balance between global exploration and local exploitation, and for which maintaining the proper balance between them is a very challenging issue (Talbi, 2009; Abdel-Basset et al., 2023).

Since then, the combination of the Salp Swarm Algorithm (SSA) with the Genetic Algorithm (GA) has gained growing popularity due to the synergy of the two algorithms. In addition to this, SSA has a desirable capability of exploring the world of the search space, as it employs an adaptive leader–follower search mechanism that allows the population to explore different areas of the search space efficiently. On the other hand, GA performs very well in exploiting the good solutions with the evolutionary operators like selection, cross over and mutation. The combination of these complementary mechanisms makes for a more well-balanced optimization process that can balance search diversity and solution refinement.

The basic idea behind the SSA–GA hybridization is that there are no universal optimizers that can work best in every problem domain. This is in line with the No Free Lunch (NFL) Theorem, which claims that the effectiveness of an optimization algorithm is dependent upon the structure of the problem being optimized (Wolpert & Macready, 1997). Thus, hybrid algorithms aim to integrate complementary search behaviours and thus make up for the weaknesses of the other. In the SSA–GA framework, the primary responsibility of SSA is to direct the global exploration in the initial search phase, and the primary responsibility of GA is to improve the quality of the candidate solutions in local exploitation phase in the later search stage.

There are some proposed hybridization methods in the literature. The sequential approach runs one algorithm at a time, usually followed by the SSA running a wide exploration and then GA running and refining the solutions that the SSA produced. The embedded strategy includes the GA operators (crossover and mutation) directly in the iterative SSA search process to continuously improve the diversity of the population. A parallel strategy runs both algorithms simultaneously and passes information between them periodically, while adaptive hybridization switches the support of each algorithm according to the convergence behavior, the solution quality or the diversity of the population. In particular, adaptive hybridization has gained significant attention for its ability to adapt the optimization process to varying search conditions, instead of relying on predetermined algorithmic structures (Castelli et al., 2022; Li et al., 2024).

Hybrid SSA–GA is successful mainly because an optimal balance between exploration and exploitation is reached. Exploration allows the search process to explore a wider area of the solution space, which helps to avoid premature convergence and increases the chances of finding globally promising solutions. Exploitations are used to deeply explore these interesting areas in order to make the solutions more accurate. If the exploration is excessive, convergence may get stalled while if the exploitation is too large, the search process may fall into local optima. To overcome this trade-off, hybrid SSA–GA optimization is used, where SSA is used to broaden the search space, and GA fine-tunes candidate solutions.

Another factor to take into account when optimizing a hybrid is constraint handling, especially for the MKP, which has many constraints. Candidates' solutions often fail to satisfy one or more of the capacity constraints at some point during the search process and special mechanisms are needed to ensure that the solutions remain feasible. Typical methods are repair algorithms, penalty functions, decoder-based representations, feasibility-preserving operators and adaptive constraint-handling strategies. More recently, adaptive constraint-handling techniques have gained increasing popularity, as they enable that optimization algorithms adapt their penalty parameters in an adaptive manner that reflects the nature of the changing search process, thereby enhancing convergence and solution quality (Coello Coello, 2002; Li et al., 2024).

Recent developments have further improved on SSA–GA optimization by the addition of intelligent adaptation mechanisms. To predict promising search regions, machine learning models have been added; to dynamically adjust algorithm parameters, reinforcement learning algorithms have been integrated; and to allow optimization algorithms to automatically adjust their crossover rates, mutation probability, and exploration parameter during the optimization process, self-adaptive parameter control has been used. These intelligent mechanisms decrease the need for manual parameter tuning, while simultaneously enhancing the robustness of the algorithm in various optimization issues (Lu et al., 2024).

Hybrid SSA–GA optimization is not only applied to the Multidimensional Knapsack Problem, but also in energy management, cloud computing, smart manufacturing, wireless sensor networks, feature selection, healthcare resource allocation, scheduling, and engineering design. In all these domains, hybrid algorithms have better convergence speed, higher solution quality and better computational efficiency than their individual component algorithms. The results reveal the increasing significance of “algorithm hybridization” as a generic approach to solving increasingly complicated optimisation problems.

Although these developments have taken place, there are a number of research issues that remain. The majority of hybrid SSA–GA models use pre-specified parameter values, leading to less adaptation to different optimisation environments. This comparative evaluation is typically performed only for benchmark data sets, and only a few studies consider scenarios in which optimization is dynamic, uncertain, or real-time. In addition, explainability is a recent issue, as the relationship between swarm intelligence and evolutionary operators isn't always clear. Future research is needed to overcome these limitations and formulate adaptive, interpretable, and scalable hybrid optimization frameworks that are able to adapt to dynamic optimization environments in a way that enables them to respond appropriately.

Overall, the Salp Swarm–Genetic Algorithm (GA) optimization is a remarkable improvement to the combinatorial optimization field. The combination of the complementary strengths of swarm intelligence and evolutionary computation provides a more effective balance between exploration and exploitation and these approaches are especially effective for solving the computational problems encountered with the Multidimensional Knapsack Problem and other problems of resource allocation in large-scale domains.

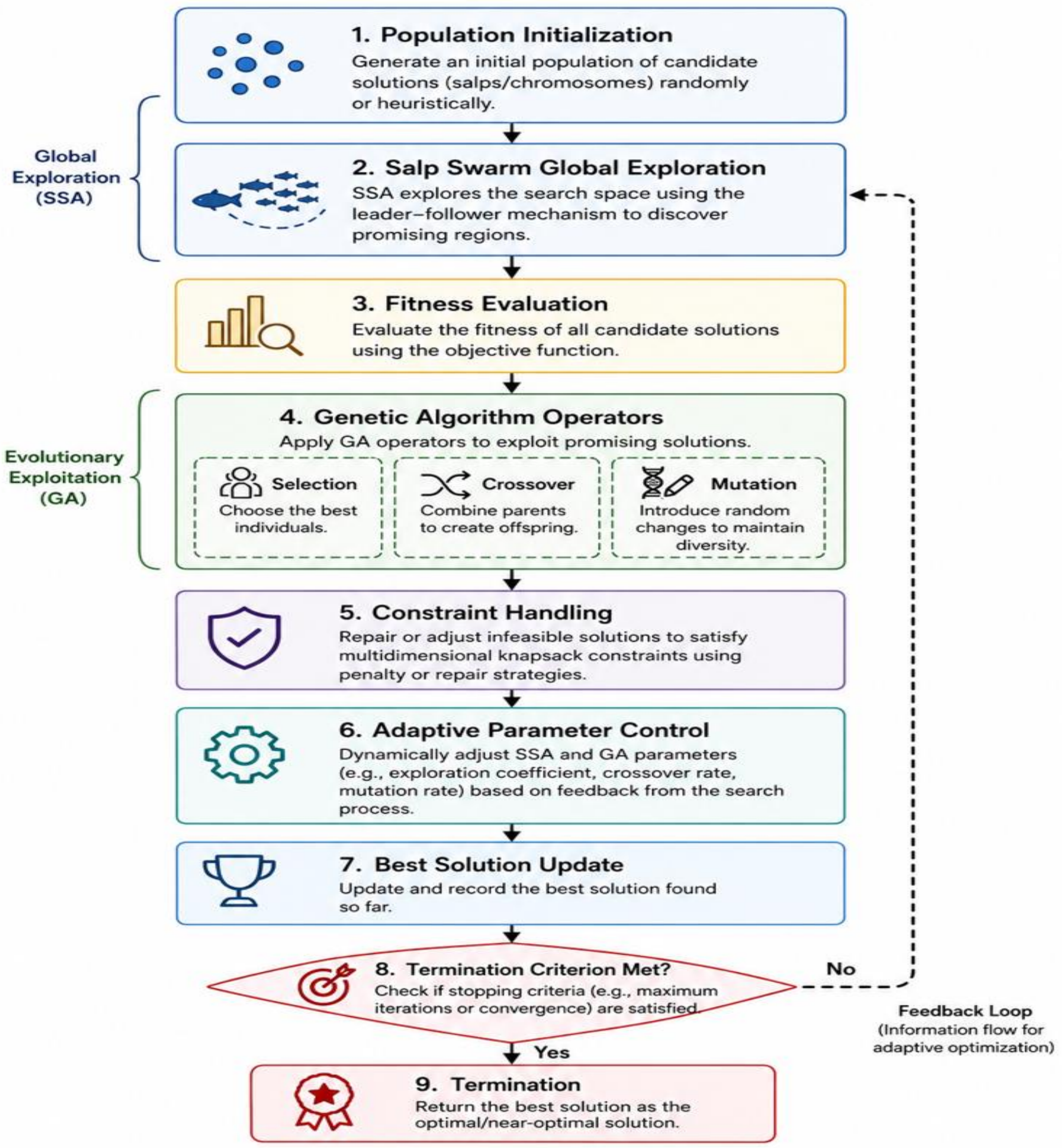

Figure 4. Conceptual Architecture of the Hybrid SSA–GA Optimization Framework.

While the theoretical foundations of hybrid SSA–GA optimization demonstrate why integrating evolutionary computation and swarm intelligence improves search performance, empirical evidence is required to validate these conceptual advantages. The next section synthesizes recent empirical studies, highlighting application domains, performance trends, benchmark comparisons, and remaining research gaps in hybrid SSA–GA optimization.

3.6. Empirical Trends and Emerging Directions in Hybrid SSA–GA Optimization

There is a growing trend of using hybrid metaheuristic algorithms, as people recognize that there is no single optimization algorithm that is effective in every problem domain. Empirical studies have been conducted in recent years, and the improvement in optimization performance that can be achieved by using the Salp Swarm Algorithm (SSA) with the Genetic Algorithm (GA) is generally better than that provided with either of the two algorithms alone, especially for high dimensional, nonlinear and constrained optimization problems. The advantages of swarm intelligence and evolutionary computation have been verified in various ways, such as an improvement in solution quality, convergence speed, search stability, and robustness, which are all commonly reported (Abdel-Basset et al., 2023; Li et al., 2024).

One of the most common approaches in the literature is the use of hybrid SSA–GA approaches for combinatorial optimization problems. They are such as multidimensional knapsack problem, job shop scheduling, travelling salesmen problem, vehicle routing problem, resource allocation problem, production planning etc. In all the applications, hybrid optimization methods consistently yield superior quality feasible solutions with fewer search iterations than traditional optimization methods. The complementary nature of the exploration mechanism provided by SSA and the evolutionary refinement provided by GA allows the algorithms to escape from the tendency to premature convergence and still perform efficient local search in the later optimization phases of the algorithms (Castelli et al., 2022).

In addition to combinatorial optimization, hybrid SSA–GA algorithms are also successfully applied in engineering optimization. Applications include optimization of structures, power system planning, scheduling of renewable energy sources, deployment of wireless sensor networks, optimization of cloud resources, and optimization of manufacturing processes. These studies illustrate that the hybrid algorithms are very effective in the solution of complex nonlinear optimization problems with several operational constraints and multiple objectives. They have a large solution space and high convergence speed, which is suitable for engineering systems that need fast and accurate decision-making (Lu et al., 2024).

Machine learning and AI is another new application that has been gaining traction. In recent years, hybrid metaheuristics are being used more and more in feature selection, hyperparameter optimization, training neural networks and optimizing deep learning models. In such applications, the SSA–GA algorithms are able to generate a more accurate predictive model than conventional search methods, because they can select the most informative features and optimize the model parameters better. With the increasing trend towards the convergence of optimization and AI, hybrid metaheuristics will remain significant in the design of intelligent decision support systems (DSs) (Li et al., 2024).

Empirical studies also show a strong trend towards adaptive hybrid optimization. Typically, the earlier developed hybrid models used constant crossover probabilities, mutation probabilities, and swarm parameters during optimization. Later studies propose self-adaptive mechanisms for parameter control that adapt search behavior as a function of the diversity of the population, convergence rate, or quality of the solutions. Adaptive optimization can enhance the robustness of the algorithms for the different instances of the problem and decreases the need to fine-tune their parameters by hand, which increases the practicality of using hybrid algorithms (Eiben & Smith, 2015).

Another is the embedding of learning mechanisms into the optimization in the hybrid approach. Many strategies have been combined with the SSA–GA framework to improve exploration, escape local optima and improve convergence: reinforcement learning, fuzzy logic, chaotic search, opposition-based learning, Lévy flight strategies, surrogate modelling. These smart extensions allow optimization algorithms to adapt their approach to previous search histories and dynamically change their behavior, marking a departure from traditional optimization techniques to adaptive and autonomous optimization systems (Abdel-Basset et al., 2023; Lu et al., 2024).

With all that said, there are some areas of current research that have yet to overcome the following limitations. Firstly, most studies for the proposed algorithms are validated using benchmark datasets while relatively few are tested by actual industrial problems. Therefore, little is known about the operational scalability, robustness, or computational efficiency. Second, there is often a focus on the accuracy of the optimisation and relatively little attention is given to computational complexity, memory usage, algoritm stability or reproducibility. Finally, the benchmark datasets vary and so do the stopping criteria and evaluation metrics, which makes it hard to directly compare competing algorithms.

One of the other weaknesses is the restricted focus on dynamic and uncertain optimization environments. While many SSA–GA applications make the assumption of fixed problem characteristics, in real optimization applications, problem characteristics are subject to change, with changing constraints, uncertain parameters and real-time decision requirements. The continuous evolution of the optimization environment is hence another potential research trajectory for developing hybrid algorithms.

Moreover, the ubiquity of explainable AI (XAI), as well as quantifiable metrics of algorithm interpretability, has come to the fore. The mechanisms of interactions between swarm intelligence and evolutionary operators in hybrid metaheuristics are still poorly understood and need to be further explored when these are used to find optimal solutions. Greater transparency and interpretability will increase user confidence and progress towards the broader adoption of hybrid optimization frameworks in key application areas like healthcare, finance, transportation and autonomous systems.

In summary, existing empirical evidence shows that hybrid SSA–GA optimization is a well-developed and competitive method to handle complex optimization problems. However, future progress will require further developments for adaptability, scalability, interpretability and validation in real-world contexts. These advancements will further reinforce the use of hybrid metaheuristic optimization as one of the enabling technologies in intelligent resource allocation and complex decision making.

The conceptual and empirical evidence presented in this review demonstrates that hybrid SSA–GA optimization has evolved into a robust framework for addressing complex combinatorial optimization problems. However, the literature also reveals fragmented theoretical perspectives and varying implementation strategies. To consolidate these insights, the next section proposes a unified conceptual framework that integrates the key components, optimization mechanisms, and research directions underpinning hybrid SSA–GA optimization for the Multidimensional Knapsack Problem.

4.0. A Unified Conceptual Framework for Hybrid Salp Swarm–Genetic Algorithm Optimization

A Unified Conceptual Framework for Hybrid Salp Swarm–Genetic Algorithm Optimization.

The reviewed literature showcases the development of hybrid optimization from the common combination of metaheuristic algorithms to increasingly intelligent algorithms that can adapt to the complexity of the optimized environment. However, there are different architectures, parameter adaptation strategies, and constraint-handling mechanisms in use in existing studies, which leads to a lack of theoretical uniformity and standardization. To fill this void, this review suggests a new conceptual framework called Unified Hybrid Salp Swarm–Genetic Algorithm (UHSSA–GA) incorporating the main ideas that are recurrently found in the literature into a unified framework for solving the Multidimensional Knapsack Problem (MKP).

The optimized framework presented is based on five interlinked optimization stages. The first step is to model the problem and represent the solution, in which decision variables, objective function and several resource constraints are formally defined. Effective chromosome or binary representations are used to encode candidate solutions to ensure that they are feasible and can be processed via efficient evolutionary operations.

The second stage is the initialization of the population and global exploration. A starting population of initial candidate solutions are created randomly or via heuristic initialization methods to aid in the quality of early solutions. The Salp Swarm Algorithm subsequently conducts the global exploration process using its adaptive leader–follower approach, which enables the search process to explore various parts of the solution space with almost no risk of premature convergence.

In the third stage, the emphasis is on evolutionary exploitation. The operators of GA such as selection, crossover and mutation are used to prune the candidate solutions identified during SSA exploration. These operators enhance the search within good areas, refine the quality of solutions, and maintain valuable genetic information through the generations. By incorporating SSA exploration into the exploitation process, the optimization process can be designed to keep an effective balance between diversification and intensification.

In the fourth stage, adaptive optimization mechanisms are introduced. The framework does not depend on static control parameters; instead, it features adaptable parameter tuning, diversity monitoring, and constraint handling. The population diversity, convergence behaviour and feasibility rate are monitored constantly and mutation probabilities, crossover rates, exploration coefficients and penalties are adjusted automatically as the search process progresses. This is an adaptive feedback mechanism which will improve the robustness of its optimization over different kinds of optimization problems with different complexity.

Solution evaluation and learning are the fifth steps. Objective function values, along with other performance measures (convergence rate, computational efficiency, feasibility rate, robustness, stability of solution) are used to evaluate candidate solutions. Optimization results can then be used to guide optimization of parameters or learning procedures in subsequent cycles, establishing a continuous optimization process.

One special aspect of the proposed framework is the incorporation of feedback connections between the different stages in the optimization process. While exploring, exploiting, and evaluating are distinct activities, information can be passed around continuously throughout the optimization cycle. During solution evaluation, performance indicators are produced that affect exploration and exploitation actions in turn, allowing the algorithm to react to the changing optimization conditions. This ability to learn in an iterative fashion is a feature of adaptive and intelligent optimization systems that have been developed more recently.

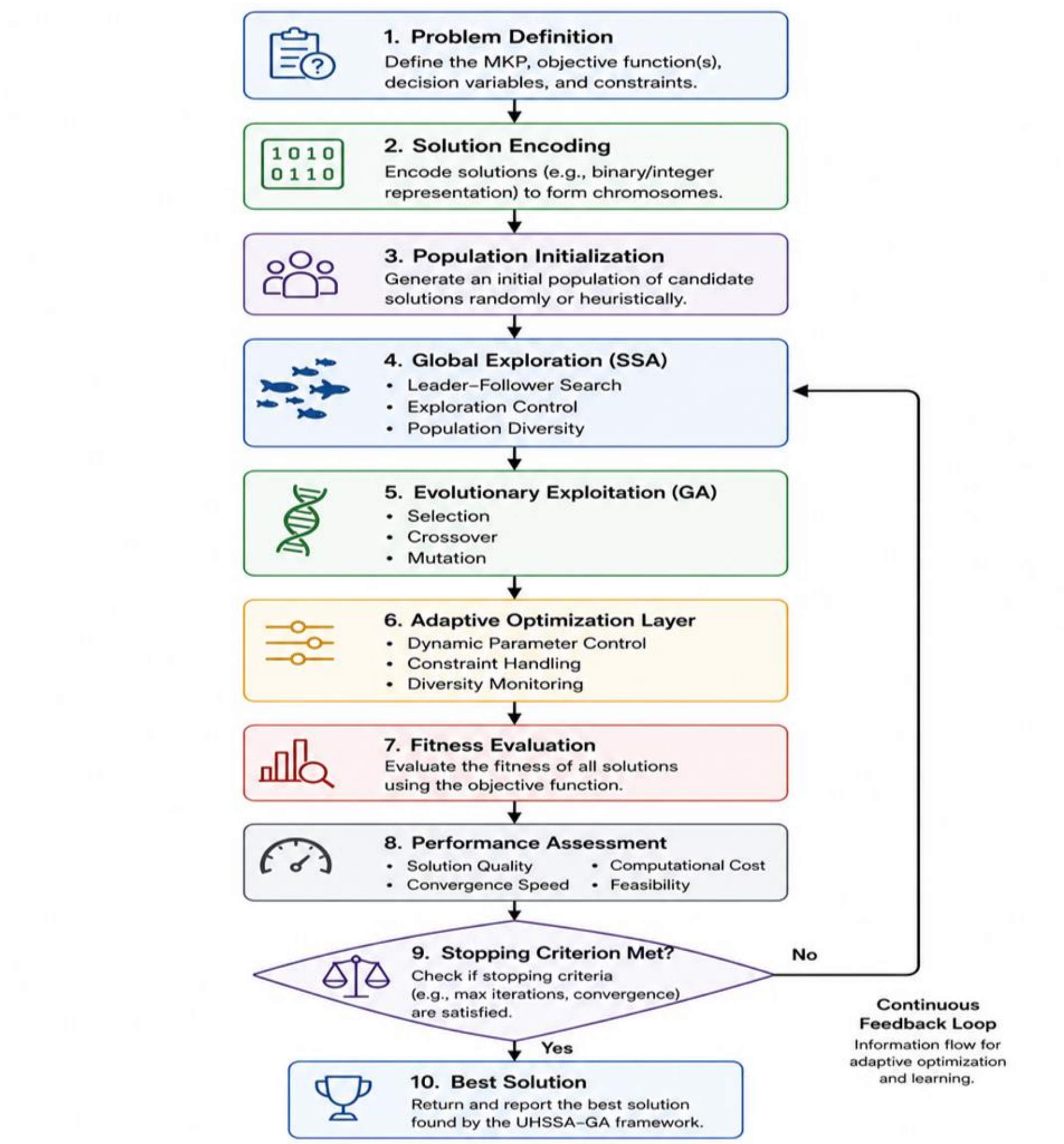

Figure 5. Proposed Unified Hybrid Salp Swarm–Genetic Algorithm (UHSSA–GA) Conceptual Framework.

A further feature of the framework is that future optimization problems are increasingly uncertain, have dynamically changing constraints, and involve high dimensional search spaces. As a result, it offers a flexible architecture to support new technologies such as reinforcement learning, surrogate-assisted optimization, explainable artificial intelligence, distributed computing, and digital twin environments. These extensions place the framework within the context of scalable foundation for future research of hybridized metaheuristics.

Overall, the proposed Unified Hybrid Salp Swarm – Genetic Algorithm Framework combines the main concepts found throughout this review into a unified theoretical model. The framework combines exploration, exploitation, adaptive parameter control, constraint handling and continuous learning to offer a structured approach to the development of scalable, robust, and computationally efficient optimization algorithms that can solve the Multidimensional Knapsack Problem and other complex combinatorial optimization problems.

The proposed framework integrates the conceptual and empirical insights identified throughout the review into a unified architecture for hybrid optimization. The final section discusses the theoretical contributions, practical implications, study limitations, and future research directions arising from this synthesis before presenting the overall conclusions.

5.0. Discussion

This review paper combined the conceptual background, theoretical development and recent advances of the hybrid Salp Swarm–Genetic Algorithm (SSA–GA) optimization method to solve the Multidimensional Knapsack Problem (MKP). The analysis shows how the computational complexity of the MKP has made the path to the use of intelligent metaheuristic algorithms for finding good solutions in reasonable computation time, gradually moving from the exact optimization methods. In this context, the hybrid optimization method of Salp Swarm Algorithm and the Genetic Algorithm has been seen as one of the most effective optimization approaches that combine global search with local search.

The review also emphasizes the key points that the success of hybrid SSA–GA optimization comes from the complementary qualities of the algorithms. Both SSA and GA enhance the exploration process by other means; the former is done by its adaptive leader–follower search mechanism while the latter is done by the systematic improvement of candidate solutions by evolutionary operators, which are systematic operators for enhancement. Both SSA and GA improve the exploration process in other ways; the former through its adaptive leader–follower search mechanism, the latter through its systematic improvement of candidate solutions by systematic operators for improvement, called evolutionary operators. They have been integrated to solve common optimization problems, such as premature convergence, loss of population diversity and slow convergence in high dimensional search spaces. As a result, hybrid SSA–GA frameworks have been shown to be competitive in a variety of combinatorial and engineering optimization problems.

Although these developments have occurred, there are several research gaps revealed in the literature. Existing hybrid algorithms are mostly designed with fixed parameter values, which decreases their flexibility in various optimization contexts. Second, empirical assessments are still largely based on benchmarks, and there is comparatively little validation based on large-scale industrial or real-time optimization problems. Third, the performances of competing hybrid algorithms are not easily compared because of the lack of uniformity in benchmark datasets, stopping criteria, and performance measures. Lastly, although interpretable artificial intelligence is gaining significance in decision support systems, there is not much work dedicated to the transparency and explainability of algorithms.

To overcome these issues, future work needs to focus on the development of adaptive and self-learning hybrid optimization frameworks that are able to adapt exploration, exploitation and parameter settings dynamically according to the variations of problem characteristics. The inclusion of reinforcement learning, surrogate modelling, explainable AI, a digital twin and distributed computing architectures presents many opportunities to enhance scalability, robustness and computational efficiency. The validation should also be extended to real scenarios like cloud computing, smart manufacturing, sustainable supply chain management, transportation systems, etc., for future research and to show practical applicability in these applications that go beyond benchmark optimisation problems.

One important contribution of this review is the proposed Unified Hybrid Salp Swarm–Genetic Algorithm (UHSSA–GA) Framework, which combines the main concepts extracted in the literature throughout the process into a unified theoretical framework. The framework integrates structured problem modelling, adaptive global exploration, evolutionary local exploitation, intelligent parameter control, constraint handling and continuous performance feedback in a single comprehensive form which acts as a foundation for next-generation hybrid metaheuristic algorithms. It also provides a common conceptual frame of reference that could help in the development of more consistent algorithms, their comparative assessment, and knowledge-sharing across the broader combinatorial optimization community.

6.0. Conclusion

The hybrid SSA–GA optimization is a major step forward in the research of intelligent optimization. The combination of swarm intelligence with evolutionary computation offers an efficient way to solve the computational complexity problem of MDP and allows the use of increasingly complex optimization problems. With problems increasingly large, growing in dimensionality and uncertainty, hybrid adaptive metaheuristics will be increasingly important in intelligent decision-making. The conceptual framework and research program proposed here would offer a framework for further development in theory and practice and would help advance scalable, adaptive, and efficient optimization methodologies.

Declaration

Source of Funding

This research did not benefit from grant from any non-profit, public or commercial funding agency.

Competing Interests Statement

The authors have declared that no competing financial, professional or personal interests exist.

Consent for publication

The authors wrote and edited the manuscript and consented to the publication of this research work.

Availability of data and material

Not applicable for this study.

References

Abdel-Basset, M., Mohamed, R., Chakrabortty, R.K., Ryan, M.J., & Mirjalili, S. (2023). A comprehensive review of hybrid metaheuristic optimization algorithms: Recent advances, applications, and future directions. Archives of Computational Methods in Engineering, 30(2): 1065–1105. https://doi.org/10.1007/s11831-022-09828-4.

Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Computing Surveys, 35(3): 268–308. https://doi.org/10.1145/937503.937505.

Boussaïd, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. Information Sciences, 237: 82–117. https://doi.org/10.1016/j.ins.2013.02.041.

Bushaj, V., & Büyüktahtakın, İ.E. (2024). Advances in multidimensional knapsack problem optimization: Models, algorithms, and emerging applications. European Journal of Operational Research, 322(2): 385–406. https://doi.org/10.1016/j.ejor.2024.01.018.

Cacchiani, V., Caprara, A., & Toth, P. (2022). Knapsack problems: Recent developments and future perspectives. 4OR, 20(1): 1–38. https://doi.org/10.1007/s10288-021-00488-9.

Castelli, M., Silva, S., & Vanneschi, L. (2022). Hybrid metaheuristics for combinatorial optimization: Current trends and future challenges. Swarm and Evolutionary Computation, 71: Article 101067. https://doi.org/10.1016/j.swevo.2022.101067.

Coello Coello, C.A. (2002). Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art. Computer Methods in Applied Mechanics and Engineering, 191(11–12): 1245–1287. https://doi.org/10.1016/s0045-7825(01)00323-1.

Dellinger, J., Mavridou, T., & Zachariadis, E.E. (2022). Exact and heuristic algorithms for multidimensional knapsack problems: A computational review. Computers & Operations Research, 145: Article 105850. https://doi.org/10.1016/j.cor.2022.105850.

Eiben, A.E., & Smith, J.E. (2015). Introduction to evolutionary computing (2nd Ed.). Springer. https://doi.org/10.1007/978-3-662-44874-8.

Goldberg, D.E. (1989). Genetic algorithms in search, optimization, and machine learning. Addison-Wesley.

Holland, J.H. (1975). Adaptation in natural and artificial systems. University of Michigan Press.

Kellerer, H., Pferschy, U., & Pisinger, D. (2004). Knapsack problems. Springer. https://doi.org/10.1007/978-3-540-24777-7.

Li, X., Wang, Y., Zhang, H., & Chen, Z. (2024). Adaptive hybrid metaheuristic optimization for large-scale combinatorial problems: A review. Applied Soft Computing, 156: Article 111401. https://doi.org/10.1016/j.asoc.2024.111401.

Lu, Y., Zhao, J., Xu, L., & Wang, P. (2024). Intelligent hybrid optimization algorithms integrating machine learning and swarm intelligence: A comprehensive review. Expert Systems with Applications, 251: Article 123918. https://doi.org/10.1016/j.eswa.2024.123918.

Martello, S., & Toth, P. (1990). Knapsack problems: Algorithms and computer implementations. John Wiley & Sons.

Mirjalili, S., Gandomi, A.H., Mirjalili, S.Z., Saremi, S., Faris, H., & Mirjalili, S.M. (2017). Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. Advances in Engineering Software, 114: 163–191. https://doi.org/10.1016/j.advengsoft.2017.07.002.

Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization: An overview. Swarm Intelligence, 1(1): 33–57. https://doi.org/10.1007/s11721-007-0002-0.

Scherer, M., Klamroth, K., & Stiglmayr, M. (2024). Recent advances in combinatorial optimization under uncertainty: A review. European Journal of Operational Research, 323(1): 1–22. https://doi.org/10.1016/j.ejor.2024.02.011.

Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. Journal of Business Research, 104: 333–339. https://doi.org/10.1016/j.jbusres.2019.07.039.

Talbi, E.G. (2009). Metaheuristics: From design to implementation. John Wiley & Sons.

Wolpert, D.H., & Macready, W.G. (1997). No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation, 1(1): 67–82. https://doi.org/10.1109/4235.585893.

Yang, X.S. (2020). Nature-inspired optimization algorithms (2nd Ed.). Elsevier.