Stochastic local search: a state-of-the-art review

Muhamet Kastrati, Marenglen Biba
Department of Computer Science, University of New York Tirana, Albania

Article Info

Article history:

Received Nov 4, 2019
Revised Jul 12, 2020
Accepted Jul 27, 2020

Keywords:

Ant colony optimization
Evolutionary algorithms
Greedy randomize adaptive
Iterated local search
Stochastic local search

ABSTRACT

The main objective of this paper is to provide a state-of-the-art review, analyze and discuss stochastic local search techniques used for solving hard combinatorial problems. It begins with a short introduction, motivation and some basic notation on combinatorial problems, search paradigms and other relevant features of searching techniques as needed for background. In the following a brief overview of the stochastic local search methods along with an analysis of the state-of-the-art stochastic local search algorithms is given. Finally, the last part of the paper present and discuss some of the most latest trends in application of stochastic local search algorithms in machine learning, data mining and some other areas of science and engineering. We conclude with a discussion on capabilities and limitations of stochastic local search algorithms.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Muhamet Kastrati,
Department of Computer Science,
University of New York Tirana,
"Kodra e Diellit", Tirana, Albania.
Email: muhamet.kastrati@gmail.com

1. INTRODUCTION

Researchers are struggling to tackle combinatorial problems as the number of these is present in several branches in science and industry on which computational methods have been widely applied, including artificial intelligence, machine learning, data mining, operations research, bio-informatics and electronic commerce. Some of the most well-known combinatorial problems include finding shortest or cheapest paths in graphs, finding solutions for propositional formulae (SAT), scheduling, time-tabling, planning, optimizing resource allocation and other various domains [1]. Stochastic local search (SLS) has proved to be successfully and extensively used approach for solving combinatorial hard problems. SLS algorithms use randomization method during the generation or selection of candidate solutions [1]. These algorithms are commonly used for solving hard combinatorial optimization and decision problems. As presented by [1] some of the early and successfully applied examples of SLS techniques used for solving optimization problems includes the Lin-Kernighan algorithm, which is applied for solving the travelling salesman problem (TSP) [2], evolutionary algorithms [3] and simulated annealing [4] as well as some other successful examples of using these algorithms for solving NP-complete decision problems, including graph coloring problem (GCP) [5], the Satisfiability problem in propositional logic (SAT) [6, 7]. More recently, SLS algorithms have successfully been applied for problem-solving and for modeling in various areas of scientific and engineering problems.

The working principle in traditional local search algorithms when dealing with an instance of a combinatorial problem rely on the fact that search for solutions happen in the space of candidate solutions. Additionally, the local search process starts by moving from current solution to other solution in a neighborhood space of the candidate solutions, and decision on each search step has been made by using only existing information [1]. On the other side, SLS algorithms, when used for solving hard combinatorial problems are characterized by randomization, respectively, generation of initial solutions as well as decisions taken on each step is typically based on randomization. Furthermore, some SLS methods during the search process use some kind of memory techniques in order to keep a limited number of most recently visited candidate solutions [1]. As mentioned above several SLS algorithms have been effective approach used mainly to solve hard combinatorial problems, including SAT, MAX-SAT, TSP, Scheduling, Time-Tabling and Protein Folding. Some other useful progress regarding to SLS algorithms for SAT is done, covering algorithms of the GSAT, GWSAT, robust SLS algorithms for MAX-SAT (WalkSAT, IRoTS, GLSSAT and SAPS) and other hybrid SLS methods [1].

SLS algorithm engineering involves many challenges that the research community must address. They are a very strong supporter of many effective heuristics for solving hard computationally complex problems. Generally, engineering of these types of algorithms in manual way requires a very careful and to much effort in order to reach high and acceptable performance. On the other side, automatic algorithm configuration is an impressive approach introduced to improve the performance of algorithms for computationally hard problems.

A lot of work related to algorithm engineering has been done by research community with regard on automatic algorithm configuration and parameter tuning techniques. Examples of these include ParamILS algorithm introduced by Hutter *et al*. [8], automated algorithms configuration and parameter tuning presented by Hoos in [9], automatic design of hybrid SLS algorithms by Marmion *et al*., [10], grammar-based generation of SLS heuristics by Mascia *et al*. [11], permutation flowshop problems by Pagnozzi and Stützle [12], algorithm comparison by automatically configurable SLS frameworks by Mascia *et al*., [13], revisiting simulated annealing by Franzin and Stützle [14]. On the other side, Franzin *et al*., [15] studied the effect of transformations for numerical parameters, Franzin and Stützle [16] performed comparison of acceptance criteria in randomized local searches, Mu *et al*., [17] conducted a research on the impact of automated algorithm configuration etc.

In the last years, to much effort has been made by research community relating to the application of SLS in various domains of science and engineering. Some of the most successful examples of SLS application includes: combinatorial problems (SAT and TSP), artificial intelligence, machine learning and data mining, scheduling, time-tabling, protein folding and to many other areas. In this paper we have been much more focused on the research work conducted in the last five years in the area of SLS. We will start with the work of Zhou and Hu [18], who presented the gradient-based adaptive stochastic search (GASS) for non-differentiable optimization and latter Rosin [19], who presented how the unweighted SLS can be effective for random CSP benchmarks, Drugan [20], in which was presented stochastic Pareto local search for many objective quadratic assignment problem instances, Frohlich *et al*., [21] presented application of SLS for satisfiability modulo theories.

Another significant contribution to the field includes a paper by Boughaci *et al*., [22] with regard to the application of SLS for image steganography, Wang *et al*., [23] on estimation of the distribution algorithm with a SLS for uncertain capacitated arc routing problems, followed by Rezoug and Boughaci [24] dealing with integration of self-adaptive harmony search with a SLS for tackling knapsack problem. Furthermore, in the last two years there are several other studies conducted with regard to the SLS application, including Putikhin and Kascheev [25], they used SLS for solving SAT problems by extending continuous Boolean formulas, Chu *et al*., [26] presented neighboring variables based configuration checking in SLS for weighted partial maximum satisfiability, Luo [27], who applied stochastic iterative evolution CT reconstruction algorithm for limited-angle sparse projection data, Yu *et al*., [28] introduced the Thompson sampling for optimizing SLS. Oliveira *et al*., [29] studied analysis of the ACO algorithm for solving TSP, Paquete and Stützle [30] presented a review of SLS Algorithms for multi objective combinatorial optimization, Niu *et al*., [31] introduced a new SLS approach for computing preferred extensions of abstract argumentation, Santos *et al*., [32], who performed analysis of SLS methods for the unrelated parallel machine scheduling problem and Weise *et al*., [33], who showed an improved generic BET-AND-RUN strategy with performance prediction for SLS.

The number of papers published in last twenty years on some of the most-well known computer libraries (such as IEEE, Springer, Elsevier and ACM) better shows the latest trends direction and scientific significance of this field. SLS and its successful application has also attracted the attention of several scientific disciplines including computer science and their branches such as artificial intelligence, machine learning, data mining, economics and management, physics, chemistry and bio-informatics. There has already been one excellent book introduced by two pioneers of the field, Hoos and Stützle [1], which presents an excellent and comprehensive overview on SLS methods, techniques and algorithms application. In 2015, ten years later, again were Hoos and Stützle [34] who provide a general overview on SLS Algorithms.

This paper serves like a complementary one to those previously published, on the same time provides state-of-the-art review on SLS methods, algorithms and most recent development trends and application in science and industry. First, it gives a short overview on combinatorial problems and search paradigms and some other useful background notation. Then, SLS methods are presented from three kind of aspects (classes): the first class is known as “simple” SLS methods, as the term shows they are mainly based on simple search techniques, the second class comprises of hybrid SLS methods, which integrate several searching techniques and heuristics and the third class named as “population-based” SLS methods that proceeds with a set or population of candidate solutions. These SLS methods comprise a number of well-known techniques which mainly have been inspired by natural phenomena, and some of the most successful application of these methods in science and industry have been presented. Finally, besides the recent achievements, some future challenges and limitation are briefly discussed.

The rest of this paper is organized as the following: In the second section is given a brief overview of background theory starting with a general introduction of combinatorial problems, search paradigms and SLS techniques. The third section gives comprehensive review of the state-of-the-art in SLS algorithms, methods and techniques from its early stage of development to the present days. The fourth section presents the most recent application of SLS algorithms in machine learning, data mining and other areas in science and industry. The last section provides briefly some concluding remarks.

2. BACKGROUND AND NOTATION

In the following is given a short introduction in background theory and notation as it has been considered as more than necessary in order to facilitate understanding of the SLS methods, techniques and algorithms.

2.1. Combinatorial problems

Combinatorial problems can be categorized in two major groups: combinatorial decision problems and combinatorial optimisation problems. The two prototypical combinatorial problems widely known are SAT and TSP. The number of these problems is large, and they are present in several branches of computer sciences, economics and management, physics, chemistry, bio-informatics and other research areas in science and industry in which computational methods find applications. Some other well known combinatorial problems includes finding shortest round trips (TSP), solving propositional formulae (SAT), planning and scheduling, time-tabling, resource allocation, prediction of protein structures, etc.

2.2. Search paradigms

In general almost all computational approaches used to deal with hard combinatorial problems can be represented as search algorithms. The working principle employed in case of the search approach is to generate and evaluate candidate solutions in iterative way. On one hand, when combinatorial decision problems approach is used, the process of evaluating a candidate solution is related with the process of decision whether it represents an actual solution. On the other hand, when dealing with optimization problems, it generally implies determining the respective value of the objective function.

a. *Solution methods for combinatorial problems*-there has been a great interest and many efforts have been made to the design of combinatorial (optimization) problems, and a number algorithm for solving these kinds of hard and complex problems have been designed. In general, these algorithms are typically categorized as either exact or approximate algorithms. The first class is known as exact algorithms and comprises algorithms that guarantee to solve every finite size instance of a combinatorial optimization problem. While the other class comprises algorithms which make a trade-off between the guarantee of finding optimal solutions and getting good solutions in polynomial-time [35].
b. *Constructive algorithms*-generate initial solutions from scratch and later solution components are added to the initial solutions according to some rules until a solution is complete. This type of algorithms is considered to be the fastest approximate methods, however the solutions provided by them quite often is with lower quality compared with one provided by local search algorithms.
c. *Local Search*-the working principle of local search algorithms is as following, they start by generating an initial solution and then iterate from current solution to other candidate solution in a neighborhood space of the candidate solutions, by replacing actual solution by better one found [35].
d. *Perturbative (local) search*-when dealing with combinatorial problems candidate solutions consist of solution components and typical case is with assigning truth values to individual propositional variables when dealing with SAT problems. On the other hand, candidate solutions can easily be changed into new candidate solutions by applying one or more modification on respective solution components. This process of changing is known as the perturbing of a given candidate solution. According to [1] search algorithms which are based on this mechanism (technique) in order to generate the candidate solutions are called perturbative search methods.
e. *Constructive (local) search*-the generation of the candidate solutions in combinatorial problems are made by repeatedly extending partial solutions that can be defined as a search problem and the main

objective here is to get a ‘good’ candidate solution, where used for optimization problems, and the “goodness” correlate to the objective function. Algorithms based on this mechanism are called constructive search methods (also known as constructive heuristic methods).

2.3. Stochastic local search

The working principle in traditional local search algorithms when applied to solve combinatorial problem is characterized with fact that search for solutions happen in the space size of candidate solutions. In addition, the local search proceeds by first taking an generated initial candidate solution, and then is iterated from one candidate solution to another candidate solution within predefined neighborhood, and the decision on each step is taken based only on existing local information [1]. However, local search typically is characterized with two kind of problems such as: (i) getting stuck in local optima and (ii) being misguided by evaluation (objective) function.

3. SLS METHODS

In this part, are shortly presented some of the most prominent SLS methods, including their application to solve hard combinatorial problems.

3.1. Simple SLS methods

In order to ensure the search process to escape from a local optima (minimum), a number of SLS methods accept worsening moves. This part is devoted to these methods which are known as simple SLS methods, as these methods employ only one type of search techniques, on limited neighborhood sets [34].

a. *Randomized iterative improvement (RII)*-is simply based on iterative improvement extended with randomization. More precisely, in each step iteration is performed based on probability w_p, the selection of next searching position s' is done using uniformly at random within current neighborhood $N(s)$, known as uninformed random walk step, or in other circumstances $1 - w_p$ is used to perform an improvement step. Here, the w_p is called as walk probability or sometimes it is also known as noise parameter. One of the main advantages of RII is that they are easy to be implemented [1].

The successful application of RII algorithms have been proven in several optimizations and decisions problems. As presented by [34] in the 1990s are implemented some version of RII by using minor variations, in such cases the random iteration is defined with regard to the amount of constraint violations in place of choosing uniformly at random, changes those that enable RII algorithms to be one of the current state-of-the-art algorithms used to solve SAT problems and CSPs [34].

b. *Probabilistic iterative improvement (PII)*-unlike some other methods that accepts some worsening search steps, here this method is based on the idea of using some acceptance criteria which is based on probabilistic evaluation function. In contrast to RII, each step of PII requires two steps: the first step deals with selection of neighboring candidate solution s' in current neighborhood $N(s)$, which typically is done by using uniformly at random approach; the second step is used to determine whether to accept or not s' as the new candidate solution. In addition, for class of problems where minimization is required, the Metropolis conditions is used as acceptance probability [34].

$$p_{accept}(T;s;s') := \left(\begin{array}{ll} 1 & \text{if} g(s') < g(s) \\ exp(\frac{g(s')-g(s)}{T}) & \text{otherwise,} \end{array}\right.$$

where $p_{accept}(T;s;s')$ represents probability of acceptance and g denotes the evaluation function which should be reduced. The parameter T named as $temperature$ represents the impact of the probability whether to accept or not worsening search steps and is equivalent to constant-temperature defined to simulated annealing. It is worth mentioning that for value of $T = 0$, then, PII will effectively turns into an iterative improvement procedure, and for value of T=∞, then accomplishes uniform random walk [34].

c. *Simulated annealing (SA)*-is simple SLS method, which shares similar features with PII, apart from temperature parameter T that is modified at run time. This method is inspired by natural phenomenon analogy, respectively the slow cooling of solid materials. The parameter T (temperature) initially has the high value, which then is progressively decreased until the lowest temperature value is reached. High temperature values means that probability of accepting worsening candidate solutions will be high. Thus, as temperature decreases, the probability of accepting worsening candidate solutions decreases as well, which means that search process progressively begins to be greedy. Furthermore, for very low temperatures (small values of T), almost only neighbors that have better or at least equal value of evaluation to the current candidate solution can be accepted [34]. SA has been successfully and widely used to solve hard computational problems, by involving various forms of annealing and acceptance tests.

d. *Tabu search (ts)*-is another simple SLS method, which is characterized by explicit memory dependency during the search process [36]. In its simple form, known as simple tabu search, it performs iterative improvement from one potential solution s to an improved solution s' in the neighborhood of s until some stopping criterion has been satisfied. In order to avoid getting stuck on local optima and to ensure that all regions in their neighborhood have been explored, some kind of short-term memory has been used. In order to skip memorizing whole set of candidate solutions and explicitly forbidden these, here is assigned a tabu status to each component and is keept track of each solution component when it was last modified. TS algorithms even simple one has proved to perform quite good in several number of problems. However, its performance is strongly related on the tabu tenure settings. In order to escape the problem of finding fixed settings, which are suitable for a given specific problem, reactive tabu search mechanism has been introduced in order to modify the tabu tenure settings at run-time [34].

e. *Dynamic local search (DLS)*-in the previous section we have presented several techniques used for escaping from local optima by using minimally worsening steps in local search. Unlike other 'simple' SLS methods presented above, DLS [1] doesn't permit worsening search steps, but in contrast it updates the evaluation function during the local search in order to avoid getting stuck on local optima [34].

According to [34] the updated evaluation function g' is calculated as the sum $g'(s)$ and $penalty(i)$, that is:

$$g'(s) := g(s) + \sum_{i \in SC(s)} penalty(i) \tag{1}$$

$g'(s)$ represents the original evaluation function value, $penalty(i)$ denotes penalties for every solution component i, $SC(s)$ represents a set of solution components of s. At the beginning, all penalties have the value equal to zero. Depending on the penalty updating mechanism and choice of solution components that are used to adjust the penalties, there exist different variants of DLS. In the beginning, $u(i)$ is calculated for each i according to equation: $u(i) := gi(s)/(1 + penalty(i))$ where $gi(s)$ is used to measure the effect of i on the evaluation function and then penalties are increased for solution components that have maximal utility [34].

3.2. Hybrid SLS methods

There exist several more complex SLS methods known as hybrid methods, which integrate different types of search techniques (ex., construction search with perturbative local search) or involving other types of complex modifications of current candidate solutions, in order to generate effective initial starting point for following iterative improvement search [34]. The following section presents a short overview of the most prominent hybrid methods.

a. *Greedy randomized adaptive search procedures (GRASP)*-the working principle used by GRASP [37] is integration of randomized greedy constructive search with a successive perturbative local search The process of generating improved solutions by construction heuristics (greedy randomized procedure) is continually repeated until a termination criterion is met. The presence of the word *adaptive* specified in GRASP name, is to denote that hybrid search procedure involves an adaptive construction heuristic. In the similar way, the term randomized denotes that randomization is used, and it is realized by using the so-called restricted candidate list that keeps the best-scoring components depending on a heuristics function [34].

b. *Iterated greedy (IG)*-is another algorithm that belong to hybrid SLS methods. The working principle of IG is that it iteratively performs greedy construction heuristics in order to produce a sequence of candidate solutions that have high-quality. In this algorithm, the key principle is to switch between solution construction and destruction phases, while this ensures better performance through integration of two types of search techniques different to each other [34].

The basic principle applied to IG algorithm, was rediscovered a number of times and presented with different names, such as: ruin-and recreate, iterative flattening and iterative construction heuristic. IG algorithms have been successfully used for solving a number of problems and obtained results show that, especially when integrated with perturbative local search, they achieved impressive results on many optimization problems [38, 39].

c. *Iterated local search (ILS)*-is another successful hybrid SLS method, which is mainly known as hybrid between perturbation mechanism and local search algorithm. As presented in [34] an ILS algorithm comprises by four basic types of mechanisms: (i) mechanism used to generate an initial solutions (ex., construction heuristics), (ii) a subsidiary (perturbative) local search procedure, (iii) a perturbation procedure that performs modification to candidate solution and (iv) an acceptance criterion. ILS algorithms, especially its basic version, is characterized as fast and easily implemented. Furthermore, by applying some types of modification, ILS represents state-of-the-art method used to solve a wide range of combinatorial problems [34]. During the last years several versions of ILS have been rediscovered and found under different names, like large-step Markov chains [40] and chained local optimization [41], but also there exist some conceptual connection of the underlying algorithms with different types of variable neighborhood search, furthermore VNS and SVNS methods are considered as versions of ILS, which benefit by using the advantages of perturbation at run time [34].

3.3. Population-based SLS methods

These methods are considered more complex compare to other SLS methods, because they are established based on principle of using a set or population of candidate solutions.

a. *Ant colony optimization (ACO)*-is part of population-based SLS methods that has been widely and successfully employed to solve many complex combinatorial problems [42-44]. This algorithm relies on biological phenomena (ants), and here term artificial ants represents multi-agent methods whose inspiration comes from collective behaviors of real ant colonies. Typically, communication based on pheromones of the biological ant is the paradigm used. Combination of artificial ants with local search algorithms have been successfully used in several optimization tasks. Further details about these mothods can be found in [29, 44].

b. *Evolutionary algorithms (EAs)*-is one of the most widely used and succesfull population-based SLS method used for solving complex computational problems in recent years. EAs have been generally inspired from biological evaluation which characterized by three most well-known evolution mechanism such as mutation, recombination and selection. Another significant mechanism is evaluation function, which is known as fitness. These algorithms are also characterized with randomization process which is used to generate the initial set of candidate solutions, and then greedy construction heuristics that is mainly employed to seed the population. Later, this underlying population is subject of three most well known genetic-based mechanism known as mutation, recombination and selection. In general, EAs performance is strongly related with the right use of the evolutionary mechanism, due to this fact, too much research work has been done with regard to design and effective use of mutation and recombination mechanism [34]. EAs have been extenssively used to solve different kinds of real world problems and results obtained show that EAs achieved state-of-the-art result when applied to solve combinatorial related problems including finding short and implementation-friendly addition chains [45] and finding nash equilibrium in electricity markets [46, 47].

4. STATE-OF-THE-ART IN SLS

In this section, we have briefly presented a state-of-the-art review in SLS methods and its successful application in science and engineering.

4.1. Integration of systematic search and SLS

SLS and systematic search usually have been considered as two separate approaches applied to solve complex combinatorial optimization and decision problems. Due to their specific characteristics for a while these approaches have been considered more concurrent then complementary. However, during the last decade has been identified an increased attention to the exploring, design and development of hybrid methods, which involve integration of systematic search and SLS approaches with purpose enhancing the algorithms. These hybrid methods can be classified in two major classes depending on the role of combined techniques and the role they play. According to this categorization the first class comprises approaches where the systematic search techniques serves as master process while other procedure (SLS) is engaged to tackle any issue that can arise throughout the systematic search steps. In the second class, these roles have been changed, here SLS algorithm serves as the master process, while systematic search technique is used when dealing with specific tasks that arise during running time of SLS algorithm [34].

4.2. SLS algorithm engineering

The definition of general-purpose SLS methods is not simple as that of fully defined recipes: meaning that there are some open choices during the algorithms design, furthermore only proper combinations of these choices can help on designing of effective algorithms, which can be used for solving specific domain problems. As suggested by [34] more and more methodological research is needed to be undertaken towards an improved design and implementation of SLS algorithms. Although SLS algorithms engineering follows the motivation of algorithms engineering; they are mainly used to solve NP-hard problems that are characterized with complex and unpredictable behavior, additionally, the presence of stochasticity makes analysis of these algorithms more hard and complex.

4.3. Automatic configuration of SLS algorithms

As many other algorithms, the performance of SLS algorithms is strongly related to the number of design choices and parameter settings. Finding right design and configuration of SLS algorithms involves setting a large number of categorical, ordinal and numerical parameters. The main goal in designing of automatic configuration of SLS algorithms is to find the proper settings of these parameters in order to achieve the optimal performance [34]. According to [17] the automatic configuration of SLS algorithms is a powerful and widely used approach that has significant role in the design and development of algorithms that provide better performance in problem solving.

As we mentioned above the performance of SLS algorithms usually depends on the number of design choices and parameter tuning. In doing so, many researchers have been for a while unhappy with manual algorithm configuration, and in a number of fields they have introduced their approaches for automatic parameter tuning. Approaches for setting parameters have been presented and described by a number of authors, but we will describe some of the most prominent ones that have been published their research work in the last decade. Hutter *et al*., [8] in their study presented methods that set proper parameters automatically in order to enhance the performance on a given class of problem instances. Two years later, another extensive research was conducted by Hoos [9] on automated algorithms configuration and parameter tuning. In this study, they presented an extended introduction to the significant role that automated algorithms configuration and parameter tuning techniques play in the implementation of algorithms that have better performance. This study also gives a brief survey on the area of algorithm configuration and parameter tuning techniques.

Another study on automatic design of hybrid SLS algorithms has been introduced by Marmion *et al*., [10] who proposed a practical, unified structure that integrates several SLS methods. This approach is unified due to the fact that involves these metaheuristics into a single structure which can be separately instantiated and can be used to generate complex combinations and variants as well. Around the same time, grammar-based generation of SLS heuristics through automatic algorithm configuration tools has been introduced by Mascia *et al*., [11]. Authors proposed here a new approach that is based on using some sequence of categorical, integer, and real-valued parameters and automatic algorithm configuration in order to find out the algorithm that performs better for the specific problem at hand. There is another study conducted by Mascia *et al*., [13], concerning to the algorithm comparison by automatically configurable SLS frameworks, which is a case study using flow-shop scheduling problem (FSP). The obtained results showed that hybrid algorithms that are instantiated were able to match and improve over the best conventional SLS method.

Another research with regard to the impact of automated algorithm configuration on the scaling behavior of state-of-the-art inexact TSP solvers was presented by Mu *et al*., [17]. He investigated the effects of automatic algorithm configuration in regard to improving the performance of the two well known inexact solvers for the TSP, EAX and LKH. By using this new way of analyzing the empirical scaling of running time as a function of problem instance size, demonstrated that automated configuration has important effect on the scaling behavior of EAX.

Work on automated algorithm configuration and parameter tuning techniques has been also done by Franzin and Stützle [16] who have evaluated several acceptance criteria based on experimental work. They first made tuning of numerical parameters of the algorithms using automatic algorithm configuration approach for quadratic assignment problem and a permutation flowshop problem. Franzin *et al*., in [15] presented the effect of transformations of numerical parameters in automatic algorithm configuration. The authors here studied the impact of altering the sampling space of parameters in automated algorithm configurations. Franzin and Stützle [14] described SA as an ensemble of algorithmic components and describe SA variants from the literature within these components. They have also experimentally demonstrated the potential of this new approach on three well-known combinatorial optimization problems, such as quadratic assignment problem and two variants of the permutation flow shop problem. Pagnozzi and Stützle [12] presented an automatic design of hybrid SLS algorithms for permutation flowshop problems (PFSP). Here, they automatically generated a new state-of-the-art algorithm for some of the most widely studied variants of the PFSP.

4.4. Application of SLS algorithms

This part gives a brief overview of some successful applications of SLS algorithms used to solve optimization and search problems. We have categorized these examples in two major groups: application of SLS algorithms in machine learning and data mining and, application of SLS algorithms in other areas of science and engineering.

4.4.1. SLS in machine learning

In this section, are presented some of the most recent trends on application of SLS in the area of data mining, graph mining and machine leaning. Hossain *et al.*, [48] presented the application of SLS for pattern set mining. Here authors proposed the application of SLS for solving the pattern set mining, particularly, to the task of concept learning. They applied here a number SLS algorithms on a conventional benchmark problem instance for pattern set mining and the results obtained showed promising results for further exploration.

Brunato and Battiti [49] investigated SLS for direct training of threshold networks. In this study authors applied SLS algorithms for training neural networks by involving threshold activation functions, a novel technique known as binary learning machine (BLM). BLM works based on the principle of changing individual bits of each weight and selecting improving moves.

Brunato and Battiti in [50] presented a new algorithm based on multiscale SLS with binary representation for training neural networks. In this work they proposed a telescopic multiscale version of local search, where the number of bits was increased in an adaptive way, in this way leading to a faster search process and to local minima of better quality. Laachemi and Boughaci [51] presented an interesting work regarding Web service classification. In this paper, authors combined SLS with support vector machine (SVM) for Web services classification. This method was performed in two steps, first SLS meta-heuristic was used for feature selection, and later SVM algorithms was applied to do the classification task. Additionally, this method, which typically involves SLS and SVM for Web service classification was further validated by authors on the quality of web service (QWS) Dataset.

Nekkaa and Boughaci in [52] proposed a hybrid search method that integrates harmony search algorithms (HSA) with SLS for feature selection as a combinatorial optimization problem in classification. They introduced a novel selection strategy based on probability technique employed in HAS-SLS, which manages to select the appropriate solutions by using SLS to filter out irrelevant or redundant features, by providing a good trade-off between exploration and exploitation. Furthermore, the hybrid HAS-SLS is then integrated with a SVM classifier.

Farhi and Boughaci in [53] proposed use of SLS method for solving the frequent subgraph mining (FSM) problem. They introduced here the notion of diversification that comprise one of the most prominent approaches applied to solve hard optimization problems. The implementation and evaluation of proposed methods was tested on several types of data sets including synthetic and real-world one. At the same time this method was compared to other state-of-the-art methods currently used, including local search, GA and variable neighborhood search (VNS) algorithms. This method was able to efficiently discover diversified subgraphs in the search space through exploring new solutions by using randomness during the search process. The obtained results showed that SLS method produce competitive results and solutions found have better quality compared to those found by LS, GA and VNS algorithms.

Mafarja *et al.*, [54] proposed binary variants for the latest version of grasshopper optimization algorithms (GOA) used for feature selection. More precisely, GOA algorithms is applied to select optimal feature subset by removing irrelevant features, which latter will be used for classification purposes into the wrapper-based framework. Authors here introduced two mechanisms in order to design a binary version of GOA. The first mechanism, is one which is based on both Sigmoid and V-shaped transfer functions, and algorithms are named as BGOA-S and BGOA-V, respectively. The second mechanism involves a novel technique that integrates the best solution obtained so far. Additionally, a mutation operator is used to enhance the exploration phase in BGOA algorithm and is named as BGOA-M. These methods are evaluated on a considerably several benchmark data sets and obtained result showed that the performance achieved by BGOA and BGOA-M is superior when compared to performance obtained by other similar recently used techniques in the literature.

4.5. Other SLS application

In recent years, more and more efforts have been devoted to the application of SLS algorithms for solving of several computational hard optimization and search problems. Unlike previous section, here are briefly presented some of the recent successful application of SLS in other science and industry fields, including SAT, MAX-SAT, TSP, scheduling and routing problems. Boughaci *et al.*, [22] combined SLS meta-heuristics with the least significant bits (LSB) technique for image steganography. The authors in this study, implemented three methods named as LSB, LSB+LS and LSB+SLS, which achieved significant results by also demonstrating the benefits of the proposed method in image steganography. Drugan [20] presented the use of stochastic pareto local search (SprLS) for many-objective quadratic assignment problem (MOQAP) instances. Putikhin and Kascheev [25] presented a heuristic approach for finding initial values for SLS in solving SAT problem by using continuous extensions of Boolean formulas.

Wang *et al.*, [23] proposed a novel robust optimisation approach that involves estimation of the distribution algorithm (EDA) and SLS for tackling uncertain capacitated arc routing problems.

This method combines an EDA with a novel two stage SLS procedure to perform minimization of the maximal total cost over a set of different scenarios. Here, the SLS procedure is used to avoid extreme fitness values of unpromising moves in local search. The results obtained experimentally on two sets of benchmark problems showed that this algorithm outperforms state-of-the-art results.

Zhou and Hu in [18] proposed gradient-based adaptive stochastic search for non-differentiable optimization algorithm for solving general optimization problems with little structure. The proposed approach use random sampling technique to iteratively find high quality solutions from a parameterized distribution model over the solution space. Chu *et al.*, [26] proposed a new SLS algorithm called CCHNV (hard neighboring variables based configuration checking) for solving WPMS (weighted partial maximum satisfiability). Results obtained by a number of experiments on a broad range of WPMS instances showed that CCHNV performance outperforms the state-of-the-art SLS algorithms recently used to solve WPMS problem.

Luo [27] proposed a stochastic iterative evolution CT reconstruction algorithm for limited-angle sparse projection data. The working principle of this algorithm is as follow: a stochastic approach is applied for searching, and Markov Chain is used to predict iterative evolution model and accelerate the proposed algorithm's convergence. The experimental results obtained by the proposed algorithm in image reconstruction from limited-angle sparse projection data are promising and robustness.

Some other successful applications of SLS introduced recently include: analysis of SLS methods used for solving the unrelated parallel machine scheduling problem presented by Santos *et al*., [32], improved generic BET-AND-RUN strategy with performance prediction for SLS by Weise *et al*., [33]. On the other side, Lorenz and Wörz [55] demonstrated benefits of using the learned clauses on SLS, followed by He at al., [56] who presented his approach for solving floating-point constraint by using SLS, Caceres *et al*., [57] who proposed using of SLS in direct aperture optimisation problem in intensity modulated radiation therapy. Susan and Bhutani [58] introduced a novel memetic algorithm that incorporates greedy SLS mutation for course scheduling and Chu *et al*., [59] presented promising empirical investigation of using SLS for solving MAX-SAT problem.

5. CONCLUSION

Nowadays, there is great interest in research community and industry to solve complex combinatorial optimisation and decision problems, as the number of these is present in several branches in science and industry on which computational methods have been widely applied. SLS is very important and powerful tool used for solving hard combinatorial optimisation problems. As a new approach it is mainly characterized by randomization during the local search. SLS algorithms have been used for a while to tackle hard combinatorial problems as the number of these is present in several branches in science and industry including artificial intelligence, machine learning, data mining, operations research, bioinformatics and electronic commerce.

Various SLS methods have been employeed to solve combinatorial optimisation problems, but usually all these methods have been grouped in three major classes: the first class is known as "simple" SLS methods, as the term denotes they are mainly based on simple search techniques, the second class comprises of hybrid SLS methods, which integrate several searching techniques and heuristics and the third class which is more complex and is known as population-based SLS methods, which proceeds with a set or population of candidate solutions. These SLS methods comprises a number of well-known techniques mainly inspired by natural phenomena, which have been widely applied to various tasks.

Except the fact that these algorithms are widely used and are powerful tool in tackling hard combinatorial problems, manual configuration of these algorithms is complex and time consuming task. In order to overpass these problems a lot of research work has been done with regard to the automated algorithm configuration and parameter tuning techniques. Additionally, some of the latest successful application of SLS algorithms in machine learning, data mining and other fields in science and industry have been presented. We hope that issues discussed in this paper will push forward the discussion in the area SLS research, on the same time it may serve as complementary material for other researchers interested in area of SLS.

REFERENCES

[1] Holger H. Hoos and Thomas Stützle, "Stochastic local search: Foundations and applications," *Elsevier,* 2004.
[2] Shen Lin and Brian W. Kernighan, "An effective heuristic algorithm for the traveling-salesman problem," *Informs*, vol. 21, no. 2, pp. 498-516, 1973.
[3] Thomas Back, "Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms," *Oxford university press*, 1996.

[4] Scott Kirkpatrick, C. Daniel Gelatt, and Mario P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671-680, 1983.

[5] Alain Hertz and Dominique de Werra, "Using tabu search techniques for graph coloring," *Computing*, vol. 39, no. 4, pp. 345-351, 1987.

[6] Bart Selman, Hector J. Levesque, David G. Mitchell, et al., "A new method for solving hard satisfiability problems," In *Proceedings Tenth National Conference on Artificial Intelligence (AAAI-92)*, vol. 92, pp. 440-446, 1992.

[7] Bart Selman, Henry A. Kautz, and Bram Cohen, "Noise strategies for improving local search," In *AAAI-94*, vol. 94, pp. 337-343, 1994.

[8] Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown, and Thomas Stützle, "Paramils: an automatic algorithm configuration framework," *Journal of Artificial Intelligence Research*, vol. 36, pp. 267-306, 2009.

[9] Holger H. Hoos, "Automated algorithm configuration and parameter tuning," *Autonomous search*, pp. 37-71, 2011.

[10] Marie-Eléonore Marmion, Franco Mascia, Manuel López-Ibánez, and Thomas Stützle, "Automatic design of hybrid stochastic local search algorithms," In *International Workshop on Hybrid Metaheuristics*, vol. 7919, pp. 144-158, 2013.

[11] Franco Mascia, Manuel López-Ibáñez, Jérémie Dubois-Lacoste, and Thomas Stützle, "Grammar-based generation of stochastic local search heuristics through automatic algorithm configuration tools," *Computers & operations research*, vol. 51, pp. 190-199, 2014.

[12] Federico Pagnozzi and Thomas Stützle, "Automatic design of hybrid stochastic local search algorithms for permutation flowshop problems," *European Journal of Operational Research*, vol. 276, no. 2, pp. 409-421, 2019.

[13] Franco Mascia, Manuel López-Ibáñez, Jérémie Dubois-Lacoste, Marie-Eléonore Marmion, and Thomas Stützle, "Algorithm comparison by automatically configurable stochastic local search frameworks: A case study using flow-shop scheduling problems," In *International Workshop on Hybrid Metaheuristics*, vol. 8457, pp. 30-44, 2014.

[14] Alberto Franzin and Thomas Stützle, "Revisiting simulated annealing: A component-based analysis," *Computers & Operations Research*, vol. 104, pp. 191-206, 2019.

[15] Alberto Franzin, Leslie Pérez Cáceres, and Thomas Stützle, "Effect of transformations of numerical parameters in automatic algorithm configuration," *Optimization Letters*, vol. 12, no. 8, pp. 1741-1753, 2018.

[16] Alberto Franzin and Thomas Stützle, "Comparison of acceptance criteria in randomized local searches," In *International Conference on Artificial Evolution (Evolution Artificielle)*, vol. 10764, pp. 16-29, 2017.

[17] Zongxu Mu, Holger H. Hoos, and Thomas Stützle, "The impact of automated algorithm configuration on the scaling behaviour of state-of-the-art inexact tsp solvers," In *International Conference on Learning and Intelligent Optimization*, vol. 10079, pp. 157-172, 2016.

[18] Enlu Zhou and Jiaqiao Hu, "Gradient-based adaptive stochastic search for non-differentiable optimization," *IEEE Transactions on Automatic Control*, vol. 59, no. 7, pp. 1818-1832, 2014.

[19] Christopher D. Rosin, "Unweighted stochastic local search can be effective for random csp benchmarks*," arXiv preprint arXiv:1411.7480*, 2014.

[20] Madalina M. Drugan, "Stochastic pareto local search for many objective quadratic assignment problem instances," *2015 IEEE Congress on Evolutionary Computation (CEC)*, Sendai, pp. 1754-1761, 2015.

[21] Andreas Fröhlich, Armin Biere, Christoph M. Wintersteiger, and Youssef Hamadi, "Stochastic local search for satisfiability modulo theories," *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, pp. 1136-1143, 2015.

[22] Dalila Boughaci, Abdelhafid Kemouche, and Hocine Lachibi, "Stochastic local search combined with lsb technique for image steganography*," 2016 13th Learning and Technology Conference (L&T)*, Jeddah, pp. 1-9, 2016.

[23] Juan Wang, Ke Tang, Jose A. Lozano, and Xin Yao, "Estimation of the distribution algorithm with a stochastic local search for uncertain capacitated arc routing problems," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 1, pp. 96-109, 2016.

[24] Abdellah Rezoug and Dalila Boughaci," A self-adaptive harmony search combined with a stochastic local search for the 0-1 multidimensional knapsack problem," *International Journal of Bio-Inspired Computation*, vol. 8, no. 4, pp. 234-239, 2016.

[25] Nikita Putikhin and Nikolay Kascheev, "A heuristic to find initial values for stochastic local search in sat using continuous extensions of boolean formulas," *2017 IEEE East-West Design & Test Symposium (EWDTS)*, Novi Sad, pp. 1-4, 2017.

[26] Yi Chu, Chuan Luo, Wenxuan Huang, Haihang You, and Dongrui Fan, "Hard neighboring variables based configuration checking in stochastic local search for weighted partial maximum satisfiability," *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, Boston, MA, pp. 139-146, 2017.

[27] Lan Luo, Hongxia Gao, Yinghao Luo, and Yongfei Chen, "A stochastic iterative evolution ct reconstruction algorithm for limited-angle sparse projection data," *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Kansas City, MO, pp. 740-744, 2017.

[28] Tong Yu, Branislav Kveton, and Ole J. Mengshoel, "Thompson sampling for optimizing stochastic local search," In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, vol. 10534, pp. 493-510, 2017.

[29] Sabrina Oliveira, Mohamed Saifullah Hussin, Andrea Roli, Marco Dorigo, and Thomas Stützle, "Analysis of the population-based ant colony optimization algorithm for the tsp and the qap," *2017 IEEE Congress on Evolutionary Computation (CEC)*, San Sebastian, pp. 1734-1741, 2017.

[30] Luis Paquete and Thomas Stützle, "Stochastic local search algorithms for multiobjective combinatorial optimization," *Handbook of Approximation Algorithms and Metaheuristics: Methologies and Traditional Applications*, vol. 1, 2018.

[31] Dangdang Niu, Lei Liu, and Shuai Lü, "New stochastic local search approaches for computing preferred extensions of abstract argumentation," *AI Communications*, vol. 31, no. 5439, pp. 1-14, 2016.

[32] Haroldo G. Santos, Túlio A. M. Toffolo, Cristiano L. T. F. Silva, and Greet Vanden Berghe, "Analysis of stochastic local search methods for the unrelated parallel machine scheduling problem," *International Transactions in Operational Researc*h, vol. 26, no. 2, pp. 707-724, 2019.

[33] Thomas Weise, Zijun Wu, and Markus Wagner, "An improved generic bet-and-run strategy with performance prediction for stochastic local search," *AAAI Technical Track: Heuristic Search and Optimization*, vol. 33, no. 1, 2019.

[34] Holger H. Hoos and Thomas Stützle, "Stochastic local search algorithms: an overview," In *Springer Handbook of Computational Intelligenc*e, pp. 1085-1105, 2015.

[35] Thomas Stützle, "Local search algorithms for combinatorial problems," Darmstadt University of Technology PhD Thesis, 1998.

[36] Fred Glover and Manuel Laguna, "Tabu search," In *Handbook of combinatorial optimization*, pp. 2093-2229, 1998.

[37] Mauricio G. C. Resende and Celso C. Ribeiro, "Greedy randomized adaptive search procedures: Advances, hybridizations, and applications," In *Handbook of metaheuristics*, pp. 283-319, 2010.

[38] Rubén Ruiz and Thomas Stützle, "A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem," *European Journal of Operational Research*, vol. 177, no. 3, pp. 2033-2049, 2007.

[39] Hande Öztop, M. Fatih Tasgetiren, Deniz Türsel Eliiyi, and Quan-Ke Pan, "Iterated greedy algorithms for the hybrid flowshop scheduling with total flow time minimization," In *Proceedings of the Genetic and Evolutionary Computation Conferenc*e, pp. 379-385, 2018.

[40] Olivier Martin, Steve W. Otto, and Edward W. Felten, "Large-step markov chains for the traveling salesman problem," *Complex Systems*, vol. 5 no. 3, pp. 299-328, 1991.

[41] David Applegate, William Cook, and André Rohe, "Chained lin-kernighan for large traveling salesman problems," *INFORMS Journal on Computing*, vol. 15, no. 1, pp. 82-92, 2003.

[42] Marco Dorigo and Mauro Birattari, "Ant colony optimization," In *Encyclopedia of machine learning*, pp. 36-39, 2011.

[43] Marco Dorigo and Gianni Di Caro, "Ant colony optimization: a new meta-heuristic," In *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*, Washington, DC, USA, vol. 2, pp. 1470-1477, 1999.

[44] Marco Dorigo and Thomas Stützle, "Ant colony optimization: overview and recent advances," In *Handbook of metaheuristics*, pp. 311-351, 2019.

[45] Stjepan Picek, Carlos A. Coello, Domagoj Jakobovic, and Nele Mentens, "Finding short and implementation-friendly addition chains with evolutionary algorithms," *Journal of Heuristics*, vol. 24, no. 3, pp. 457-481, 2018.

[46] Forhad Zaman, Saber M. Elsayed, Tapabrata Ray, and Ruhul A. Sarkerr, "Evolutionary algorithms for finding nash equilibria in electricity markets," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 4, pp. 536-549, 2018.

[47] Luis Miguel Antonio and Carlos A. Coello Coello, "Coevolutionary multiobjective evolutionary algorithms: Survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 6, pp. 851-865, 2018.

[48] Muktadir Hossain, Tajkia Tasnim, Swakkhar Shatabda, and Dewan M. Farid, "Stochastic local search for pattern set mining," *The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014)*, Dhaka, pp. 1-6, 2014.

[49] Mauro Brunato and Roberto Battiti, "Stochastic local search for direct training of threshold networks," *2015 International Joint Conference on Neural Networks (IJCNN), Killarney*, pp. 1-8, 2015.

[50] Mauro Brunato and Roberto Battiti, "A telescopic binary learning machine for training neural networks," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 3, pp. 665-677, 2017.

[51] Abdelouahab Laachemi and Dalila Boughaci, "A stochastic local search combined with support vector machine for web services classification," *2016 International Conference on Advanced Aspects of Software Engineering (ICAASE)*, Constantine, pp. 9-16, 2016.

[52] Messaouda Nekkaa and Dalila Boughaci, "Hybrid harmony search combined with stochastic local search for feature selection," *Neural Processing Letters*, vol. 44, no. 1, pp. 199-220, 2016.

[53] Sidali Hocine Farhi and Dalila Boughaci, "Sls method for the frequent subgraph mining problem (fsm)," *2017 First International Conference on Embedded & Distributed Systems (EDiS)*, Oran,, pp. 1-6, 2017.

[54] Majdi Mafarja, Ibrahim Aljarah, Hossam Faris, Abdelaziz I. Hammouri, Al-Zoubi Ala'M., and Seyedali Mirjalili, "Binary grasshopper optimisation algorithm approaches for feature selection problems," *Expert Systems with Applications*, vol. 117, pp. 267-286, 2019.

[55] Jan-Hendrik Lorenz and Florian Wörz, "On the Effect of Learned Clauses on Stochastic Local Search," *arXiv preprint arXiv:2005.04022*, 2020.

[56] Shaobo He, Marek Baranowski and Zvonimir Rakamarić, "Stochastic Local Search for Solving Floating-Point Constraints," *International Workshop on Numerical Software Verification*, vol. 11652, pp. 76-84, 2019.

[57] Leslie Pérez Cáceres, Ignacio Araya, Denisse Soto and Guillermo Cabrera-Guerrero, "Stochastic Local Search Algorithms for the Direct Aperture Optimisation Problem in IMRT," *International Workshop on Hybrid Metaheuristics*, vol. 11299, pp. 108-123, 2019.

[58] Seba Susan and Aparna Bhutani, "A Novel Memetic Algorithm Incorporating Greedy Stochastic Local Search Mutation for Course Scheduling," *2019 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC)*, New York, NY, USA, pp. 254-259, 2019.

[59] Yi Chu, Chuan Luo, Shaowei Cai and Haihang You, "Empirical investigation of stochastic local search for maximum satisfiability," *Frontiers of Computer Science*, vol. 13, no. 1, pp. 86-98, 2019.

BIOGRAPHIES OF AUTHORS

Muhamet Kastrati is a PhD cand. at University of New York in Tirana with Master of Sciences in Computer Engineering from University Prishtina (2014). He obtained Diploma Degree in Faculty of Electrical and Computer Engineering from University of Prishtina (Kosovo) in 2007. His researches are in fields of Advanced Algorithms, Statistical Relational Learning, Machine Learning and Data Mining and Deep Learning.

Marenglen Biba Assoc. Prof at the University of New York in Tirana with PhD in Computer Sciences from University of Bari, Italy (2009). He obtained Laurea Degree (5-year) Cum Laude in Computer Science, specialization in Knowledge Engineering and Machine Learning, University of Bari, Italy in 2004. His researches are in fields of Artificial Intelligence, Machine Learning, Pattern Recognition, Data Mining, Computational Biology, Document Image Understanding, Information Extraction, Social Networks Analysis, and Natural Language Processing of Albanian.Further info on his homepage: http://www.marenglenbiba.net/