CHEMICAL ENGINEERING TRANSACTIONS

VOL. 114, 2024

A publication of AIDIC
The Italian Association of Chemical Engineering
Online at www.cetjournal.it

Guest Editors: Petar S. Varbanov, Min Zeng, Yee Van Fan, Xuechao Wang

ISBN 979-12-81206-12-0; **ISSN** 2283-9216

DOI: 10.3303/CET24114169

Metaheuristics in Logistics: Increasing the Efficiency of Algorithms by Defining Appropriate Parameter Settings

Tamara J. Bíró*, Péter Németh

Széchenyi István University, Hungary, 9026 Győr, Egyetem tér 1.
biro.tamara@sze.hu

Metaheuristic algorithms are well-researched and popular techniques in the field of optimization, which can solve complex tasks with a large number of instances with acceptable quality. They are extremely problem- and parameter-sensitive methods, so the exact definition of the necessary data and the testing of the appropriate parameters fundamentally determine the efficiency and performance of an algorithm. This is a time-consuming and expensive task. In many cases, when applying a metaheuristic, it works properly with the variables of a given task and there is no specific interval where a given algorithm can still be effective. To increase efficiency and reduce costs, the authors defined a general parameter definition by applying the Ant Colony Optimization algorithm applicable to the simple Traveling Salesman Problem with the number of cities n=50, where for values of 30 ≤ n ≤ 50, the defined parameter setting structure can be properly applied based on the results. The proposed parameter setting structure can work effectively not only for the task presented in the paper, but also for any similar task within the defined interval. In the case of tasks of a similar size, it is not necessary to experiment with the parameters to achieve the appropriate result, thereby reducing the optimization time and improving efficiency. The presentation of the set parameter setting scenarios and the obtained results all contribute to reducing the optimization time in the field of logistics as well. All of this can also help facilitate the practical application of metaheuristics in solving NP-hard tasks.

1. Introduction

There are countless methods to choose from to optimize a task, depending on how complex the given problem is. Exact methods can give exact results, but they cannot effectively handle problems with a large number of instances. On the other hand, metaheuristic solutions give an approximate optimal solution, but they can also be effectively used in complex NP-hard problems.

Few scientific works deal with the general comparison of methods that provide exact or definite optimal solutions and metaheuristic solutions, and each example is typical of the solution to a specific problem. Chandra et al. (2021), for example, compare the Branch and Bound (B&B) method with the Fruit Fly Optimization Algorithm (FOA) and the Artificial Atom Algorithm (A3) metaheuristics. In terms of processing time, the difference between the two methods is more than 12 days; however, if certain conditions are met, B&B performs better.

Applying a metaheuristic is a costly and time-consuming task. The scientific community has already proven that they can be applied well where exact methods are no longer able to provide results within an acceptable calculation time. Many metaheuristic algorithms have already been used in logistics, but there is little real use of these techniques. It is not enough to implement a method and accept the first result. It is necessary to test the parameters and determine what an acceptable setting is under the given conditions. In the paper by Chirwa et al. (2024), a good example of this can be seen, where it is properly described which are the appropriate parameter settings for a given task. This is a great help when optimizing similar tasks.

The primary goal of this paper is to find a generic parameter setting that can work effectively under specific conditions. This could increase efficiency and save time when applying and adapting an algorithm. The focus of the paper is the importance of fine-tuning the parameters, which can be seen as highlighted in Figure 1.

The paper is structured as follows. Section 2 is the literature review, section 3 describes the Travelling Salesman Problem, and section 4 presents the efficiency of the parameter tuning. In section 5, the results are summarized.

Paper Received: 20 May 2024; Revised: 23 July 2024; Accepted: 21 November 2024
Please cite this article as: Bíró T.J., Németh P., 2024, Metaheuristics in Logistics: Increasing the Efficiency of Algorithms by Defining Appropriate Parameter Settings, Chemical Engineering Transactions, 114, 1009-1014 DOI:10.3303/CET24114169

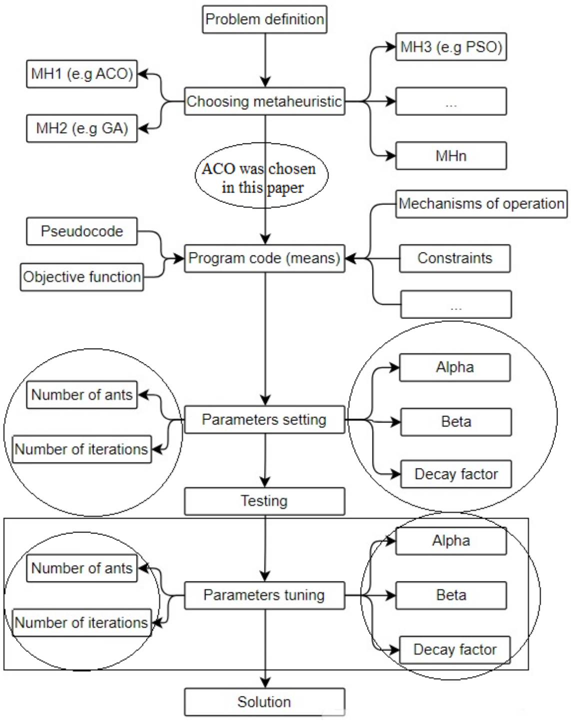

Figure 1: Flow chart: Application of metaheuristic optimization procedure

2. Literature review

Thousands of scientific papers deal with metaheuristic algorithms, as well as the Travelling Salesman Problem (TSP) and Ant Colony Optimization (ACO). Figure 2 clearly shows the literature statistics of the last five years from one database (so there are many more research materials than this) with the specified keywords. The problem is that it is difficult to find the answers to the questions that arise either on the practical side, i.e. in industry or in scientific communities, among the many papers. It is also due to this that the practical use of metaheuristics is small (Swan et al., 2022), although a lot of scientific work has been published in recent years on the excellent problem-solving ability of metaheuristic algorithms. However, most of these remain at the theoretical level and outside the scientific community, less commonly used techniques in practice.

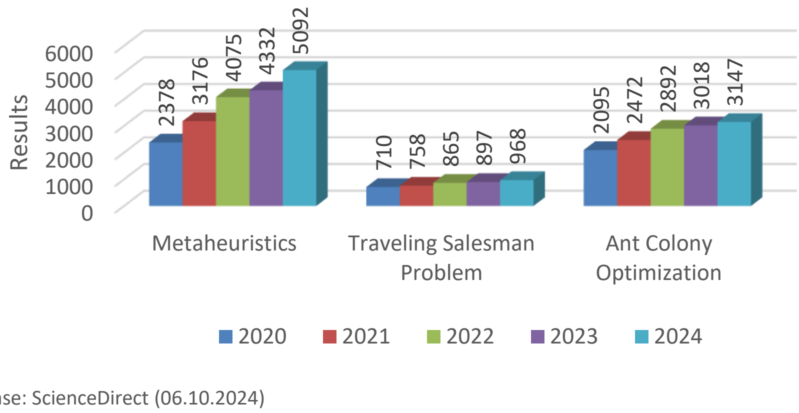

Figure 2: Literature analysis

The performance of metaheuristics and the accuracy of the results are greatly influenced by the appropriate choice of parameters (Joshi and Bansal, 2020). This is an extremely time-consuming and expensive task, as it is usually done experimentally (Shadkam, 2022). Many research works describe how important the parameter tuning of algorithms is, but most of the time, the process is not documented, and in many cases, the most favourable parameter settings for a given task are not summarised.

During the publication of a metaheuristic, countless data and information are indicated, all of which can help in adaptation, primarily for computer scientists. These include, for example, the pseudocode of the algorithm, the detailed description of the operating mechanisms, and the presentation of the most important parameters. With the help of expert knowledge that can be obtained from scientific works, a metaheuristic can most likely be used to solve various problems. However, this also requires the knowledge of specialists in a specific field.

In the field of logistics, metaheuristics are also often used, as many tasks are NP-hard. With the novel classification of metaheuristics and logistic tasks (Two groups have been distinguished: discrete and continuous. The tasks were classified according to the variables, and the metaheuristics according to the type of problems that have been solved efficiently in the literature); it is possible to select the algorithm with the highest probability of solving a given problem efficiently and to implement an optimisation method using expert knowledge available from pseudocode and papers. The combination of these two factors can ensure greater practical application of metaheuristic algorithms. To prove the validity and efficiency of a written method, it is necessary to solve the problem using an exact method since this is the basis for obtaining an exact solution to the problem, against which the result of the metaheuristic method can be compared. Then, the next important step is to adjust the parameters to improve the accuracy of the results of the optimisation procedure. Fine-tuning a technique requires the logistician's professional knowledge in the application of optimisation methods, and the cooperation of several experts in solving a problem is of paramount importance.

A well-researched and popular problem is the Traveling Salesman's Problem, which can not only be used to determine the shortest distance or the most optimal tour of given points but also has many other practical applications, such as production scheduling, material flow planning problems, waste collection management (Pop et al., 2024), picking problems, psychological data analysis, frequency assignment problem in a communication network, etc. (Jati et al., 2023).

TSP is a good choice because it is a combinatorial problem with discrete variables belonging to the NP-hard complexity class, but in the case of a small problem instance, it can be solved using an exact method, for example, linear programming, dynamic programming, and many other algorithms providing exact solutions. However, in multifactorial cases, metaheuristic solutions can provide an acceptable result. In this paper, the TSP is solved using the Ant Colony Optimization (ACO) algorithm in addition to the exact method, and the importance of parameter tuning for better results is shown. The different results obtained as a function of the parameters to solve a problem will be examined, and a general formula will be determined for the problem given the corresponding parameters within a specified interval. All this can increase efficiency in terms of time and costs, as well as sustainability in the field of logistics.

3. Traveling Salesman Problem

The Traveling Salesman Problem is a much-researched and one of the best-known combinatorial, NP-hard optimisation problems due to its wide range of applications. TSP can be formulated simply as follows: given a set of cities and the known travel cost, which can be, for example, distance, time, money, etc., between each pair of cities. The task of the agent is to visit each city exactly once and return to the starting point. For this, it is necessary to find the best solution, which is to minimise travel costs. The challenge of this task is to arrange the cities in the best possible order (Jati et al., 2023).

There are countless methods for solving TSP (Matai et al., 2010). One of the simplest deterministic solutions is a brute-force search, which explores all possible permutations of cities and calculates the total distance of each permutation. The goal is to find the permutation that results in the minimum total distance or least cost (optimal tour). Other possible exact solution algorithms that can solve a simple TSP are the Bellman–Held–Karp algorithm, Branch-and-Bound, Branch and Cut, etc. These methods can be used properly for small-scale problems since the calculation time would not be acceptable for a large-scale problem instance. "Researchers have already demonstrated that there is currently no scheme or algorithm for finding an exact solution in polynomial time." (Jati et al., 2023).

Due to the NP-hard complexity of the TSP, heuristic and metaheuristic algorithms are able to find a satisfactory, high-quality or near-optimal solution within an acceptable computation time.

TSP has already been solved with a lot of metaheuristics, the following list - without claiming to be complete - contains them: African Buffalo Optimization, Ant colony optimization, Artificial bee colony algorithm, Artificial Ecosystem Algorithm, Bean Optimization Algorithm, Bumble Bees Mating Optimization, Chicken swarm optimization, Clonal Selection Algorithm, Consultant-Guided search, Crystal Energy Optimization Algorithm, Discrete Bacterial Memetic Evolutionary Algorithm, Egyptian Vulture Optimization, Elephant Search Algorithm, Firefly algorithm, Fish swarm algorithm, Genetic algorithm, Golden ball, Harmony Search Algorithm, Honey-bees mating optimization algorithm, Hunting search algorithm, Hydrological cycle algorithm, Intelligent Water Drops Algorithm, Invasive Weed Optimization, Memetic Algorithm, Penguins Search Optimization Algorithm, Photosynthetic Learning Algorithm, River Formation Dynamics, Shuffled Frog Leaping Algorithm, Simulated annealing, Swallow Swarm Optimization Algorithm, Tabu Search algorithm, The scientific algorithms, Variable Neighborhood Descent Algorithm, Water Wave Optimization, Water-flow Algorithm (Ezugwu et al., 2021 and Kóczy et al., 2018).

4. The importance of parameter settings

Seeing the metaheuristics applied to TSP, the question may arise: which is the best and most efficient method? If an algorithm has already successfully solved a TSP, then what was the purpose of using other algorithms? There is no study that compares these results, highlighting the problem formulation and the required parameters. Expert knowledge should be gathered from individual articles, which can explain why which method works well. However, a complete comparison is almost impossible since the papers may lack the data necessary for a complete analysis. How many of the listed algorithms were/are used in practice to actually solve a task? It would be necessary to promote the more frequent use of metaheuristic algorithms in industrial practice since only a small percentage of the vast amount of theoretical knowledge and scientific material appears in real use.
Below, the authors examine the extent to which scientific papers and pseudocode of an algorithm can help on the practical side. A symmetric TSP (the distance between two cities is the same in both directions) was chosen as a test task, and Ant Colony Optimization (ACO) was chosen as a metaheuristic. First, it was examined whether a program could be created by taking into account the ACO pseudocode and its most important metaheuristic properties, and then it was tested on a specific TSP task with the objective function Eq(1). To determine whether the algorithm works properly, the tasks were solved using the exact method that provides accurate results and the ACO algorithm. The logic of the exact method is to find and explore all possible permutations of cities and calculate the total distance of each permutation. Goal: Find the permutation that results in the minimum total distance. For comprehensive proof, the study covers small, medium, and large numbers of problems. According to the authors' hypothesis, if the metaheuristic approach works properly, then only it will be able to provide results in the case of a large problem instance. In all cases, the possible solutions were examined using an exact solution method and the ACO metaheuristic approach (called ACO in the tables). The program codes were run in Python.
Table 1 shows which of the two solution methods is able to solve the problem properly based on increasing the number of cities (n), and then Table 2 shows the specific results. Based on these, it can be said that the exact method could no longer provide results within a foreseeable time at n=50 but performed better than the metaheuristic approach based on a run experiment at n=10. However, this does not mean that a better result cannot be achieved with ACO by changing the parameters.

$$\sum_{i=1}^{50}\sum_{j=1}^{50} d_{ij}x_{ij} \rightarrow min \tag{1}$$

Table 1: Running performance

Method	n=5	n=10	n=50
Exact	✓	✓	-
ACO	✓	✓	✓

Table 2: Running values

Method	n=5	n=10	n=50
Exact	160	1,050	-
ACO	160	1,105	3,315

One of the characteristics of metaheuristic algorithms is that it is necessary to experiment with the parameter settings to improve the results. The defining and most important parameters and operating mechanisms of ACO are as follows: The alpha and beta constants allow the user to control the relative importance of a trail versus its visibility (Bavey and Kollin, 2017). Pheromone trails are modified during the run of an ACO algorithm to bias the construction of new solutions. Pheromone updating usually consists of two complementary steps: pheromone evaporation and pheromone deposition. The general idea behind the two steps is to bias the pheromone trails to favour the formation of high-quality solutions (López-Ibáñez et al., 2018). In general, the larger the size of the ant colony, the stronger the search ability of the algorithm. Range of pheromone volatility factor (0, 1). If it is too small, the pheromone on the path will not evaporate in time, resulting in an excessive amount of pheromone on the path, which affects the convergence efficiency of the algorithm. If it is too large, the pheromone on the road cannot be maintained, and the ant colony loses the experience information of previous iterations (Yang et al., 2020). The authors defined the running time (t) as t < 1 min (this is the determining factor for the number of iterations). Different scenarios have been set up in order to get an approximate picture of the type of problem and the cohesion of the parameters providing a possible good solution. In each of the individual cases, n=50 cases were taken as a basis for the ACO approach. The primary goal is to find the minimum distance between cities so that each city is visited exactly once by the agent. The program was run 20 times in each scenario, providing valuable information. Table 3 shows the parameter settings of each scenario (S1-S12). Up to S5-S10, based on the results of the first 5 runs, it could be seen that the algorithm performs worse than in the first four scenarios, so they were not run 20 times. Table 4 shows the running results in the scenarios that showed promising results, highlighting the best and worst results.

Table 3: Parameter settings for scenarios

Scenario	num_ants	num_interations	alpha	beta	decay_factor
S1	10	100	1.0	2.0	0.1
S2	50	100	1.0	2.0	0.1
S3	50	200	1.0	2.0	0.1
S4	100	200	1.0	2.0	0.1
S5	50	200	0.5	5.0	0.01
S6	100	200	0.5	5.0	0.01
S7	50	200	0.1	5.0	0.01
S8	100	200	0.1	5.0	0.01
S9	50	200	1.0	2.0	0.01
S10	100	200	1.0	2.0	0.01
S11	50	200	1.0	2.0	0.5
S12	100	200	1.0	2.0	0.5

Table 4: Best and worst run results for different scenarios and average of 20 runs

Results	S1	S2	S3	S4	S11	S12
Best result	3,342	3,298	3,259	3207	3249	3,132
Worst result	3,710	3,560	3,428	3396	3469	3,369
Average	3,538.55	3,398.15	3,344.75	3,316.2	3,335.5	3,282.15

As can be seen in Figure 3, scenario 12 performed the best overall, based on which more favourable results can be obtained for solving a problem of similar size according to the following values (n represents the number of cities): Sn parameter settings: num_ants = 2n, num_iterations = 4n, alpha = 1.0, beta = 2.0, decay_factor = 0.5. After specifying the parameters that can be prescribed in general, it is also advisable to determine at what interval they work effectively. The worst (S1), the best (S12) scenario, and the parameters (Sn) that can be specified from the general description were compared on symmetric matrices with n=10, 20, 30, 40, and 50 values. The results are shown in Table 5.

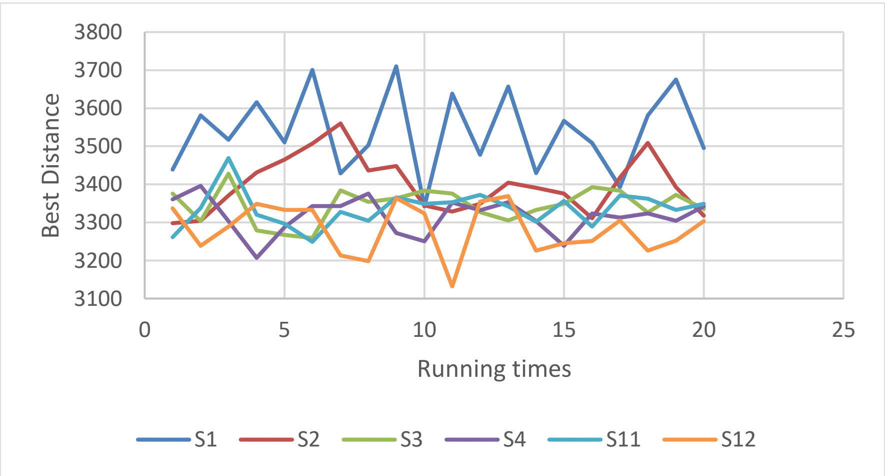

Figure 3: Graphical representation of the results of different scenarios

Table 5: Comparison of running values for scenarios S1, S12 and Sn

Scenario	M10x10	M20x20	M30x30	M40x40	M50x50
$S1_h$	1,100	2,241	2,553	3,190	3,710
$S1_s$	1,075	2,241	2,490	3,024	3,342
$S1_a$	1,091	2,241	2,517	3,134	3,534
$S12_h$	1,085	2,241	2,488	3,026	3,369
$S12_s$	1,070	2,241	2,475	2,989	3,132
$S12_a$	1,075	2,241	2,482	3,008	3,281
Sn_h	1,140	2,241	2,490	3,017	3,369
Sn_s	1,110	2,241	2,475	2,991	3,132
Sn_a	1,120	2,241	2,485	3,002	3,281

(Subscripts: h - the highest of the running values, s - the lowest of the running values, a - the average of all the running values)

The results show that the general parameter description structure defined for the task with n=50 cities still works properly and provides an efficient, optimal result in the event of a 40 % downward deviation.

5. Conclusions

Applying a metaheuristic algorithm in practice is an extremely time-consuming, expensive and complicated task. Although there are thousands of scientific works on the subject, the real, practical application of metaheuristics is small. The authors wanted to contribute to solving this deficiency with this article. The operation and accuracy of a metaheuristic is greatly influenced by the correct setting of the parameters. This is done experimentally. The authors defined a general parameter structure for solving a symmetric matrix-based TSP problem with ACO. In the cases with the following number of cities (n), the algorithm works efficiently with the following parameter settings: if $30 \leq n \leq 50$, then num_ants = 2n, num_iterations = 4n, alpha = 1.0, beta = 2.0, decay_factor = 0.5. The experimental results showed that the parameter settings determined based on the task presented in the paper give the best results, which contributes to the efficient solution of similar types of problems by reducing the total optimisation time. A further research goal is the examination of tasks with n>50 cities and the determination of additional general parameter structures.

References

Bavey A., Kollin F., 2017, Ant Colony Optimization Algorithms – Pheromone Techniques for TSP. <www.diva-portal.org/smash/get/diva2:1105900/FULLTEXT01.pdf>, accessed 07.10.2024.

Chandra A., Natalia C., Naro A., 2021, Comparative Solutions of Exact and Approximate Methods for Traveling Salesman Problem. Lámpsakos, 25, 1-12, DOI: 10.21501/21454086.3804.

Chirwa E.M., Molokwane P.E., Brink H.G., Wang Y.-T., 2024, Simulation of Cr(VI) Removal in a Biofilm Reactor System and Optimisation Using an Ecological Algorithm, Chemical Engineering Transactions, 110, 199-204.

Ezugwu A.E., Shukla A.K., Nath R., 2021, Metaheuristics: a comprehensive overview and classification along with bibliometric analysis. Artif Intell Rev, 54, 4237–4316.

Jati G.K., Kuwanto G., Hashmi T., Widjaja H., 2023, Discrete Komodo algorithm for traveling salesman problem. Applied Soft Computing, 139, 110219, ISSN 1568-4946.

Joshi S.K., Bansal J.C., 2020, Parameter tuning for meta-heuristics. Knowledge-Based Systems, 189, 105094.

Kóczy L.T., Földesi P., Tüű-Szabó B., 2018, Enhanced discrete bacterial memetic evolutionary algorithm - An efficacious metaheuristic for the traveling salesman optimization. Information Sciences, 460-461, 389–400.

López-Ibáñez M., Stützle T., Dorigo M., 2018, Ant Colony Optimization: A Component-Wise Overview. In: Martí, R., Pardalos, P., Resende, M. (Eds), Handbook of Heuristics. Springer, Cham, Switzerland.

Matai R., Singh S., Lal M., 2010. Traveling Salesman Problem: an Overview of Applications, Formulations, and Solution Approaches, in: Davendra, D. (Ed.), Traveling Salesman Problem, Theory and Applications. InTech, DOI: 10.5772/12909.

Pop P.C., Cosma O., Sabo C., Sitar C.P., 2024, A comprehensive survey on the generalized traveling salesman problem. European Journal of Operational Research, 314, 3, 819-835, ISSN 0377-2217.

Shadkam E., 2022, Parameter setting of meta-heuristic algorithms: a new hybrid method based on DEA and RSM. Environ Sci Pollut Res, 29, 22404–22426.

Swan J., Adriaensen S., Brownlee A.E.I., Hammond K., Johnson C.G., Kheiri A., Krawiec F., Merelo J.J., Minku L.L., Özcan E., Pappa G.L., García-Sánchez P., Sörensen K., Voß S., Wagner M., White D.R., 2022, Metaheuristics “In the Large”. European Journal of Operational Research, 297, 2, 393-406.

Yang L., Wang Y., Zhang J., 2020, Parameter Analysis and Simulation Experiment of Ant Colony Optimization on Small-scale TSP Problem. IOP Conf. Ser.: Mater. Sci. Eng., 768, 072095.