Tailoring a Red Deer Algorithm to Solve a Generalized Network Design Problem

Imen Mejri[a], Safa Bhar Layeb[b] and Emna Drira
LR-OASIS, National Engineering, School of Tunis, University of Tunis El Manar, Tunis, Tunisia

Keywords: Network Design Problems, Metaheuristics, Bio-inspired Algorithms, Red Dear Algorithm.

Abstract: This work investigates solving a challenging network design problem using the recently introduced evolutionary metaheuristic, namely the Red Deer Algorithm (RDA), that mimics the Scottish Red Deer's behavior during their breeding season. The RDA is tailored to solve a generalized network design problem that aims to design a network with minimal cost while satisfying several practical constraints. To assess the performance of this new bio-inspired metaheuristic on solving such NP-hard problem, computational experiments were conducted on Benchmark as well as real-world instances. Computational experimentation illustrates the accuracy of the RDA that outperforms all of the existing recent metaheuristics.

1 INTRODUCTION

Metaheuristics in general and bio-inspired metaheuristics in particular are still of interest to researchers as well as practitioners (e.g. Almufti et al., 2021; Ma et al., 2019) for solving real-world engineering design problems. The recent surveys of Swan et al. (2021) and Osaba et al. (2021) are dedicated to these cutting edges evolutionary computation techniques. Actually, there are several lately released metaheuristics that are increasingly applicable to solve NP-hard problems. Precisely, the Red Dear Algorithm (RDA) is a new nature-inspired metaheuristic which paradigm was first introduced by Fathollahi-Fard and Hajiaghaei-Keshteli(2016) and then well-engineered by Fathollahi-Fard et al. (2020a). The RDA mimics the mating behavior of Scottish Red Deer during their breading season. This new evolutionary bio-inspired metaheuristic has shown its performance for solving the vehicle routing problem, the travelling salesman problem and the single-machine problem, which are known as NP-hard problems (Fathollahi-Fard et al., 2020a).

Due to their high ability to deal with a wide range of real-world problems, Network Design Problems (NDPs) represent the largest category of combinatorial optimization in the fields of industries, logistics, telecommunications and energy (Mejri et al., 2021). In terms of graph theory, a NDP is modeled by a set of nodes representing for example power plants in an energy system (e.g. Singh et al.,2021) or warehouses in a distribution system (e.g. Layeb et al., 2018). Eventually, those nodes have to inter-communicate to exchange numerous types of merchandises, data, power, etc. In this purpose, edges link adequate pairs of nodes chosen properly. Those links are known as arcs generally characterized by capacities, variable costs, fixed costs, facilities, etc. Depending on the situation, an edge can model a road in transportation problems, a high voltage in energy systems, and so on (e.g. Mejri et al., 2019a; 2019b). To sum up, the objective of the network design problem consists on generating the optimal configuration that satisfies the maximum demands partially/totally with the minimum possible total system cost. According to the recent survey of Salimifard and Bigharaz (2020) about the classification of network design problems and their resolution approaches during the last two decades, exact methods are less and less used since 2004. However, the progress of metaheuristics is noticed during this period and is expected to continue to grow in the future as the size of the studied problems increases steadily.

[a] https://orcid.org/0000-0002-2182-8558
[b] https://orcid.org/0000-0003-2536-7872

Mejri, I., Layeb, S. and Drira, E.
Tailoring a Red Deer Algorithm to Solve a Generalized Network Design Problem.
DOI: 10.5220/0010689000003062
In *Proceedings of the 2nd International Conference on Innovative Intelligent Industrial Production and Logistics (IN4PL 2021)*, pages 32-39
ISBN: 978-989-758-535-7

This work investigates a challenging variant of NDPs, namely the Generalized Discrete Cost Multi-commodity Network Design Problem (GDCMNDP) (Khatrouch et al., 2019; Mejri et al., 2019c). Each edge is characterized by several bidirectional facilities with discrete prefixed costs and capacities. In telecommunication, facilities model a set of available technologies that provides high bandwidth connections, each transmitting services and/or information at different rates. Given the multicommodity flow demands to route partially or totally, the GDCMNDP mandates the installation of no more than one facility on each edge to minimize the sum of the fixed facility installation costs and the penalties of unrouted demands. Not surprisingly, the GDCMNDP is an NP-hard combinatorial problem. This highly challenging generalized network design problem was recently addressed by Khatrouch et al. (2019). The authors proposed three basic greedy heuristics as well as three metaheuristics, namely a Genetic Algorithm (GA), a Biogeography-Based Optimization method (BBO), and a hybrid Genetic Algorithm coupled with a Variable Neighborhood Search procedure (GA-VNS). Computational experiments highlights that the GA-VNS leads to the best performance. Besides, a stochastic version of the GDCMNDP, with uncertain amounts of flow demands, was effectively solved using a simulation-based optimization framework (Mejri et al., 2019c).

To the best of our knowledge, this work is the first to tailor the Red Deer Algorithm in order to solve a generalized Network Design Problem, and more precisely, the highly challenging GDCMNDP. Computational experiments on Benchmark instances and real-world instances from the literature reveal that the RDA generates good solutions within very fair computation times and outperforms the existing meta-heuristics (Khatrouch et al., 2019).

The rest of the article is organized as follows. A mathematical formulation of the Generalized Discrete Cost Multi-commodity Network Design Problem is detailed in Section II. Then, Section III presents the tailoring of the Red Deer Algorithm for solving the GDCMNDP. The computational results are detailed in Section IV. Finally, Section V draws conclusions and future research avenues for this work.

2 MATHEMATICAL MODEL

To model NDPs, combinatorial optimization formulations (e.g. Ennaifer et al., 2016) and in particular linear programming based models such as the path-based formulations (e.g. Mejri et al., 2019c; 2019d) and the flow-based formulations (e.g. Mejri et al., 2018; Layeb et al., 2017) are commonly used. More precisely, the GDCMNDP is defined on an undirected connected graph G=(V,E) where V is the set of nodes indicating all possible switching centers or customers, and E is the set of edges, corresponding to all possible optical fibers connections between the centers, in the field of telecommunication. To derive a valid mathematical model for the GDCMNDP, let's begin by introducing the following notations:

<u>Sets:</u>

V Set of nodes, indexed by i

E Set of edges, indexed by $e=\{i,j\}$, $i, j \in V$

A Set of arcs derived from E such that for each edge $e = \{i,j\} \in E$, two directed arcs are generated (i,j) and $(j,i) \in A$

K Set of distinct point-to-point multi-commodities, indexed by k

L_e Set of potential facilities representing types of optical fibers that can be installed on edge $e \in E$, indexed by l

<u>Parameters:</u>

c^l_e Fixed cost of installing facility l on edge e

u^l_e Bidirectional capacity of facility l on edge e

s_k Source node of commodity k

t_k Sink node of commodity k

d_k Flow demand of commodity k to route from s_k to t_k

p_k Service cost, representing penalty to be paid per unit of commodity demand not routed k

<u>Decision Variables:</u>

X^k_{ij} Continuous non-negative variable that reflects the amount of flow of commodity k circulating through arc (i,j)

Z_k Continuous non-negative variable that reflects the commodity demand not routed k

Y^l_e Binary variable that equals to 1 if facility l is installed on edge e, 0 otherwise

Using these notations, a Mixed Integer Linear Programming (MILP) formulation of the GDCMNDP can be stated as follows:

$$Minimize \sum_{e \in E} \sum_{l \in L_e} c_e^l Y_e^l + \sum_{k \in K} p_k Z_k \quad (1)$$

The objective function (1) minimizes the total installation fixed costs and penalties involved in unrouted multicommodity demands. It should be minimized subject to the following constraints:

$$\sum_{l \in L_e} Y_e^l \leq 1, \forall e \in E \quad (2)$$

For each edge, Constraints (2) impose the selection of a maximum of one facility.

$$\sum_{j:(i,j) \in A} X_{ij}^{\ k} - \sum_{j:(j,i) \in A} X_{ji}^{\ k} = \begin{cases} d_k - Z_k \ if \ i = s_k \\ 0 \ if \ i \in V, i \neq s_k, i \neq t_k, \forall k \in K \\ -(d_k - Z_k) \ if \ i = t_k \end{cases} \quad (3)$$

For each node and for each commodity, Constraints (3) compel the flow conservation principle.

$$\sum_{k \in K} X_{ij}^{\ k} + \sum_{k \in K} X_{ji}^{\ k} \leq \sum_{l \in L_e} u_e^{\ l} Y_e^l, \forall e = \{i, j\} \in E \quad (4)$$

For each edge, Constraints (4) force that the bi-directional flow through the two arcs to not surpass the installed capacity.

$$X_{ij}^{\ k} \geq 0 \quad \forall (i,j) \in A, \forall k \in K \quad (5)$$

$$Y_e^{\ l} \in \{0,1\}, \quad \forall e \in E, \forall l \in L_e \quad (6)$$

$$Z_k \geq 0 \quad \forall k \in K \quad (7)$$

Constraints (5)-(7) indicate the nature of the considered decision variables.

This yields to Model (1)-(7) that is clearly a valid mathematical formulation of the DCMNDP. Besides, it is a so-called compact MILP formulation, in terms of the combinatorial optimization. Thus, it is possible to solve it directly by a commercial MIP solver. It is worthy to mention that Model (1)-(7) will unsurprisingly become intractable in practice and could not solve the GDCMNDP to optimality, once the size of the instances increases. Therefore, we invoked the RDA metaheuristic to find good feasible solutions for this NP-hard problem.

3 RED DEER ALGORITHM

Recently addressed and well-established by Fathollahi-Fardet al. (2020a), the Red Deer Algorithm is a new nature-inspired population-based metaheuristic inspired from the Scottish Red Deer mating behavior during the breeding season. Precisely, from the end of September till the end of November begins the Scottish Red Deer breading season, called also the rut. During mid to late autumn, stags (Red Deer males) come back to the hinds (groups of female Red Deer) territory in order to be engaged in disputes with other males by starting to show their masculine power and display of dominance including roaring, parallel walks and fighting. The reproductive success in males is directly attached to their behavior during the mating season and also their physical strength citing the body size, the strength, the roaring and the development of antlers. At the end of a battle, some males can be seriously injured or can even die. After that the dominant male chases the week one and starts mating with the hinds. This is how the natural selection is generated.

As an evolutionary metaheuristic, the proposed RDA starts with an initial population of Red Deer candidates composed of the Red Deer males and Red Deer females (Fathollahi-Fard et al., 2020a). After the roaring and fighting phase between males also called 'intensification phase', males are divided into two groups: ‘male commanders’ (the strongest males to win the fights) and ‘stags’ (the chased males due to their weakness or injuries during fights) according to their strength and at this end ‘harems’ are formed. A harem is a group of hinds whose size is related to the commanders' abilities and power in the process of roaring, fighting and mating. Thus, the more the male is strong, the more the harem is large. Besides, the male commander of each harem is mating with α percent of hinds in his own harem and also with β percent of females of the closest harem to his territory which represents the so-called 'diversification phase'. Across the mating process, new candidates appear as offspring of the current candidates, i.e. the next generation.

Based on the research of Fathollahi-Fard et al. (2020a), we tailor the RDA to solve the

GDCMNDP. A pseudo-code of the proposed metaheuristic is presented in **Algorithm 1** below.

Algorithm 1: Red Deer Algorithm.

```
Set the parameters: MaxIt, nPop, nMale,
α, β, μ;
Initialize the Red Deer population;
Select nMale male Red Deer;
S_RDA= the best solution;
It=0;
While (It<MaxIt)
    For each Red Deer male
        Select the best male;
    endFor
    Form male commanders and stags;
   For each male commander
    Fight male commanders and stags;
    Update male commanders and stages' positions;
   endFor
    For each male commander
    Mate male commander with hinds of his harem;
       Mate male commander with hinds of another harem;
    endFor
   For each stag
    Mate stag with any hinds of any harem;
    endFor
    Select the next population using the elitism principle;
    UpdateS_RDA;
    It=It+1;
endWhile
Return S_RDA.
```

First, it is worthy to note that **Algorithm 1** is based on 6 input parameters to tune: *MaxIt*=the maximum number of iteration, *nPop*=the Red Deer population size, *nMale*=the number of Red Deer males, α=the percentage of mating inside a harem, β=the percentage of mating outside a harem, and μ=the percentage of male commanders.

Red Deer Representation:

Each Red Deer is represented by binary array that corresponds to the vector of the decision network variables $(Y_e^l)_{e\in E, l\in L_e}$ in Model (1)-(7). Obviously, each Red Deer should correspond to a feasible solution for the GDCMNDP. This condition is ensured while Constraints (2) are respected.

Initial Population Generation:

The initial Red Deer population is randomly generated of feasible candidates. During the initialization of the Red Deer population, it is simultaneously sorted in a descending order with respect to the fitness. Then, the first *nMale* candidates are selected as Red Deer males and the rest of the population is considered as Red Deer females, i.e. hinds.

Fitness Determination:

Similar to the genetic algorithm, the fitness function express the value of a Red Deer referencing to his grace, roaring power and fighting strength. In our case, it corresponds to the objective function (1) in Model (1)-(7). More precisely, it is calculated as the sum of the installation costs and the penalties of unrouted demands. Besides, let's precise that this amount of penalties is calculated through solving a routing problem represented as a linear programming model derived from Model (1)-(7) once the $(Y_e^l)_{e\in E, l\in L_e}$ vector is fixed.

Male Commanders' Selection:

For the Scottish Red Deer, intense roaring attracts hinds, so that some males are more successful than other in constructing harems due to their strength and capacity to win several fights. Thus, Red Deer males are divided according to their power into commanders which are the strongest males and the stags. The number of commanders is correlated to the μ parameter and expressed as

$$nMaleCom = Round\ (\mu.\ nMale) \tag{8}$$

Naturally, once the male commanders identified, the rest of the males are considered as stags.

Males' Fighting and Mating:

As in (Fathollahi-Fard et al., 2020a), the Red Deer males fight between each other based on the fitness function. As an issue of each fight, one of them can win the match. Thus, we update the male position according to the bubble sorting principle. The strongest male will take command of the harm while the looser will be chased away. The harems as the hinds mating selection are performed as proposed by Fathollahi-Fard et al. (2020a).

Next Population Selection:

As experienced by Yadav and Sohal (2017) as well as Khatrouch et al. (2019), the tournament selection appears to be better than the rank-based solution, since the tournament selection repetition is faster than the ranking section in generating the population of individuals. In the terms of convergence, the tournament selection shows more efficient results than the roulette wheel selection. Hence, the ability of reaching the maximum/minimum fitness with the lowest number of generations is the highest following the elitism selection technique. Thus the choice of the next generation, we use the Elitism selection according to the best ranked fitness.

4 COMPUTATIONAL EXPERIMENTATION

In order to evaluate the performance of the RDA in solving the GDCMNDP, the metaheuristic was implemented in C++ language on the Microsoft Visual C++ 2010 Express in concert with the commercial MILP solver, ILOG CPLEX 12.5. The code was run on a personal computer with8Go of RAM and a Core i7-7500U at 2.70 GHz.

The test-bed consists of two types of Benchmark instances. Six instances (MH01-MH06) from (Mrad and Haouari, 2008) and five real-world instances from telecommunication field: one instance denoted by (EON) as European Optical Network (Fumagalli et al., 1999) and four instances (NSFNET-NSFNET4) expressing the National Science Foundation Networks (Miyao and Saito, 1998). Table I displays the main characteristics of the considered instances and their data files are freely available at (Layeb, 2018).

Table 1: The Main Instances Characteristics.

Instance	$\|K\|$	$\|V\|$	$\|E\|$
NSFNet1	21	14	42
NSFNet2	21	14	42
NSFNet3	21	14	42
NSFNet4	21	14	42
EON	36	19	72
MH01	45	10	30
MH02	105	15	45
MH03	105	15	50
MH04	105	15	60
MH05	435	30	120
MH06	595	35	140

It is well-known that fixing the appropriate algorithmic parameters has a great influence on the efficiency and effectiveness of the algorithm performance. It is very often a trade-off between the quality of the solution and the required computation time. Therefore, an empirical experimentation has been conducted to fix the parameters values as: *MaxIt*=10, *nPop*=100, *nMale*=10, α=70% β=30%, and μ=rand[1,10]%. These settings yield to the best solutions within a CPU time compromise.

Table II reports the numerical results of the MILP Model (1)-(7), the proposed Red Deer Algorithm (RDA), and the three metaheuristics developed by Khatrouch et al. (2019): the Genetic Algorithm (GA), the Biogeography-Based Optimization method (BBO), and the hybrid Genetic Algorithm coupled with a Variable Neighborhood Search procedure (GA-VNS). Let's denote by **Sol***: the optimal solution of each instance found by the state-of-the-art MILP solver, **Time:** the CPU time in seconds needed to the convergence of each approach, **Gap:** the Gap en percentage between the found solution and the best solution divided by the best solution. In Table II, "-" indicates that the solver fails to find the optimal solution. Besides, it is noteworthy that the GA, the GA-VNS, and the BBO of Khatrouch et al. (2019) were run a machine with similar technique characteristics than this work. Therefore, the computation times reported in Table II could be comparable.

From Table II, we can derive several observations. Despite the sophisticated combinatorial optimization tools built into commercial MILP solvers, Model (1)-(7) become intractable as the instance size increases. It could not provide optimal solutions for instances with more than 30 nodes. However, all the meta-heuristics could solve all the tested instances.

Moreover, the Biogeography-Based Optimization method lags far behind all the other metaheuristics. It reveals very poor performance both in terms of the solution quality and the high computation times. Actually, the BBO requires significant computational time while showing weak performance. In other words, the BBO requires much more computational effort to find solutions for the GDCMNDP than the other evolutionary population-based metaheuristics.

The Genetic Algorithm performs reasonably well with an average gap of 1.7%. Besides, Strengthening the GA with the Variable Neighborhood Search procedure has enhanced its performance only in terms of quality solution. In fact, the average gap of the GA-VNS is about 1.4% while its average CPU time has increased by 167 % compared to the CPU time of the basic GA.

Table 2: Numerical Results.

Instance	Model (1)-(7)		GA (Khatrouch et al., 2019)		GA-VNS (Khatrouch et al., 2019)		BBO (Khatrouch et al., 2019)		RDA	
	*Sol**	*Time*	*Gap*	*Time*	*Gap*	*Time*	*Gap*	*Time*	*Gap*	*Time*
NSFNet1	13	100.6	0.0%	465.9	0.0%	523.4	0.0%	3235.8	0.0%	255.6
NSFNet2	21	0.6	0.0%	321.6	0.0%	344.3	0.0%	244.7	0.0%	304.6
NSFNet3	15	10.7	6.6%	239.8	6.6%	331.5	0.0%	2303.5	6.1%	305.7
NSFNet4	18	6.8	0.0%	384.2	0.0%	487.4	0.0%	2354.0	0.0%	1976.1
EON	14	114.4	7.1%	984.2	7.1%	1113.7	21.4%	4563.7	7.6%	126.7
MH01	993	11.2	1.2%	989.3	1.2%	1564.0	1.2%	5277.8	0.6%	136.6
MH02	2444	182.3	0.8%	1795.9	0.8%	2224.4	7.4%	6084.2	0.0%	243.3
MH03	3167	117.4	0.0%	2404.9	0.0%	3328.4	21.1%	8635.2	0.0%	286.3
MH04	3481	29.2	0.0%	3482.3	0.0%	6930.0	43.7%	9301.1	0.0%	550.8
MH05	-	-	0.3%	6782.5	0.0%	8170.4	100.0%	57857.4	0.0%	2177.3
MH06	-	-	2.8%	8265.3	0.0 %	18679.9	114.4%	110910.5	0.2%	3026.2
Average			**1.7%**	*2374.2*	**1.4%**	*3972.5*	**28.1%**	*19160.7*	**1.3%**	*853.6*

Table II shows that the Red Deer algorithm finds the best solutions for 9 out of the 11 tested instances. The proposed Red Deer Algorithm seems capable of exploring the research space effectively in order to identify almost the optimal solution. Actually, the RDA lightly outperforms the GA-VNS in term of quality solution as the average gap of the RDA is about 1.3%. However, although the RDA finds solutions very similar to those of the GA-VNS, the RDA converges in extremely short computation times and mostly unrivaled by those of the GA-VNS. Indeed, the average CPU time required by the RDA is about 21% of that required by the GA-VNS.

Thus, the Red Deer algorithm shows very promising performance that can be further extended by extensive computational experiments on additional benchmark instances of the GDCMNDP as well as other variants of the NDPs.

5 CONCLUSIONS

The scope of this work is to tailor the recently introduced evolutionary Red Deer Algorithm to solve the challenging Generalized Discrete Cost Multi-commodity Network Design Problem effectively. This generalized network design problem has a significant spectrum of applications, especially in distribution, logistics and telecommunications, with a deep business impact on the network companies. Moreover, the GDCMNDP is known to be an NP-hard combinatorial optimization problem and its resolution still remains a hard task for researchers as well as practitioners, especially when the size of the instances increases. We have investigated the Red Deer Algorithm and implemented it adequately to handle the particular features of network design problems. The numerical results of the computational experiments on Benchmark instances illustrate the effectiveness of the Red Deer Algorithm when compared with the metaheuristics from the existing literature. Actually, the Red Deer Algorithm generates high-quality solutions to large-size instances in very reasonable computation times.

These promising results encourage going further in investigating the Red Deer Algorithm characteristics in order to enhance its performance. As research avenues for future work, we suggest improving the proposed approach by decreasing its input parameters as recently proposed in (Fathollahi-Fard et al., 2020b). Having fewer parameters to control seems to lead to deeper phases of intensification and research that allow the best solution to be found more efficiently.

REFERENCES

Almufti, S. M., Marqas, R. B., Othman, P. S., & Sallow, A. B. (2021). Single-based and Population-based Metaheuristics for Solving NP-hard Problems. *Iraqi Journal of Science*, *62*(5), 1710-1720.

Ennaifer, N. B., Layeb, S. B., & Zeghal, F. M. (2016, April). On lower bounds computation for the Discrete Cost Multicommodity Network Design Problem. In *2016 International Conference on Control, Decision and Information Technologies (CoDIT)* (pp. 511-516). IEEE.

Fathollahi-Fard, A., &Hajiaghaei-Keshteli, M. (2016). Red Deer Algorithm (RDA); a new optimization algorithm inspired by Red Deers' mating. In *International conference on industrial engineering, IEEE.,(2016 e)* (pp. 33-34).

Fathollahi-Fard, A., Hajiaghaei-Keshteli, M., &Tavakkoli-Moghaddam, R. (2020a). Red deer algorithm (RDA): a new nature-inspired meta-heuristic. *Soft Computing*, 1-29.

Fathollahi-Fard, A. M., Ahmadi, A., &Sajadieh, M. S. (2020b). An Efficient Modified Red Deer Algorithm to Solve a Truck Scheduling Problem Considering Time Windows and Deadline for Trucks' Departure. *Evolutionary Computation in Scheduling*, 137-167.

Fumagalli, A., Cerutti, I., Tacca, M., Masetti, F., Jagannathan, R., &Alagar, S. (1999, March). Survivable networks based on optimal routing and WDM self-healing rings. In *IEEE INFOCOM'99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320)* (Vol. 2, pp. 726-733). IEEE.

Khatrouch, S., Layeb, S. B., &Siala, J. C. (2019). Bio-inspired metaheuristics for the generalised discrete cost multicommodity network design problem. *International Journal of Metaheuristics*, *7*(2), 176-196.

Layeb, S. B., Heni, R., &Balma, A. (2017, May). Compact MILP models for the discrete cost multicommodity network design problem. In *2017 International Conference on Engineering & MIS (ICEMIS)* (pp. 1-7). IEEE.

Layeb, S. B., Jaoua, A., Jbira, A., &Makhlouf, Y. (2018). A simulation-optimization approach for scheduling in stochastic freight transportation. *Computers & Industrial Engineering*, *126*, 99-110.

Layeb, S.B. (2018) GDCMNDP Instances [online] https://www.researchgate.net/publication/322446665_GDCMNDP_Instances.

Ma, H., Shen, S., Yu, M., Yang, Z., Fei, M., & Zhou, H. (2019). Multi-population techniques in nature inspired optimization algorithms: A comprehensive survey. *Swarm and evolutionary computation*, *44*, 365-387.

Mejri, I., Layeb, S. B., &Zeghal, F. (2021). A Survey on Network Design Problems: Main Variants and Resolution Approaches. *European J. Industrial Engineering, to appear.*

Mejri, I., Layeb, S. B., & Mansour, F. Z. (2019, April). Enhanced Exact Approach for the Network Loading Problem. In *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)* (pp. 970-975). IEEE.

Mejri, I., Layeb, S. B., &Zeghal Mansour, F. (2019). Formulations and Benders Decomposition based Procedures for the Discrete Cost Multicommodity Network Design Problem. *International Journal of Computing and Digital Systems*, *8*(6), 659-668.

Mejri, I., Layeb, S. B., Haouari, M., & Mansour, F. Z. (2019). A simulation-optimization approach for the stochastic discrete cost multicommodity flow problem. *Engineering Optimization*, *52*(3), 507-526.

Mejri, I., Haouari, M., Layeb, S. B., & Mansour, F. Z. (2019). An exact approach for the multicommodity network optimization problem with a step cost function. *RAIRO-Operations Research*, *53*(4), 1279-1295.

Mejri, I., Layeb, S. B., & Mansour, Z., F. (2018, November). Solving the Discrete Cost Multicommodity Network Design Problem to Optimality. In *2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)* (pp. 1-5). IEEE.

Mrad, M., &Haouari, M. (2008). Optimal solution of the discrete cost multicommodity network design problem. *Applied Mathematics and Computation*, *204*(2), 745-753.

Miyao, Y., & Saito, H. (1998). Optimal design and evaluation of survivable WDM transport networks. *IEEE Journal on Selected Areas in Communications*, *16*(7), 1190-1198.

Osaba, E., Villar-Rodriguez, E., Del Ser, J., Nebro, A. J., Molina, D., LaTorre, A., Suganthan, P. N., Coello, C. A., & Herrera, F. (2021). A Tutorial on the Design, Experimentation and Application of Metaheuristic

Algorithms to Real-World Optimization Problems. *Swarm and Evolutionary Computation*, 100888.
Salimifard, K., &Bigharaz, S. (2020). The multicommodity network flow problem: state of the art classification, applications, and solution methods. *Operational Research*, 1-47.
Smith, J. (1998). *The book*, The publishing company. London, 2nd edition.
Singh, A., Sharma, S., & Singh, J. (2021). Nature-inspired algorithms for wireless sensor networks: A comprehensive survey. *Computer Science Review*, *39*, 100342.
Swan, J., Adriaensen, S., 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. (2021). Metaheuristics "In the Large". *European Journal of Operational Research, in press*.
Yadav, S. L., &Sohal, A. (2017). Comparative study of different selection techniques in genetic algorithm. *International Journal of Engineering, Science and Mathematics,6*(3), 174-180.