A Novel Heuristic Optimization Algorithm for Solving the Delay-Constrained Least-Cost Problem

Amina Boudjelida(✉) and Ali Lemouari(✉)

Laboratory of Mathematics and Applied Mathematics (LMAM), University of Jijel, Jijel, Algeria
boudjelidaamina@gmail.com, lemouari_ali@yahoo.fr

Abstract. The quality of the multicast routing service (QoS) is an NP multi-objective optimization problem. It is one of the main issues of transmission in communication networks that consist of concurrently sending the same information from a source to a subset of all possible destinations in a computer network. Thus, it becomes an important technology communication. To solve the problem, a current approach for efficiently supporting a multicast session in a network is establishing a multicast tree that covers the source and all terminal nodes. This problem can be reduced to a minimal Steiner tree problem (MST), which aims to look for a tree that covers a set of nodes with a minimum total cost that has been proven to be NP-complete. In this paper, we propose a novel algorithm based on the greedy randomized search procedure (GRASP) for the Delay-Constrained Least-Cost problem. Constrained with the construction and improvement phase, the proposed algorithm makes the difference in the construction phase through using a new method called EB heuristic. The procedure was first applied to improve the KMB heuristic in order to solve the Steiner tree problem. Obtained solutions were improved by using the tabu search method in the enhancement process. Computational experiments on medium-sized problems (50–100 nodes) from literature show that the proposed metaheuristic gives competitive results in terms of cost and delay compared to the proposed results in the literature.

Keywords: Multicast routing · DCLC · QOS · Optimization · Heuristic · Metaheuristic

1 Introduction

Depending on the number of destinations, network routing can be categorized into three basic types: unicast (one-to-one), broadcast (one-to-all) and multicast (one-to-many). The multicast communication model has been defined firstly in [1]. Multicast, or selective streaming, is a communication approach for the

É. Renault et al. (Eds.): MLN 2020, LNCS 12629, pp. 349–363, 2021.
https://doi.org/10.1007/978-3-030-70866-5_23

transmission of datagrams to a group of zero or more hosts identified by a single destination group address [1]. The notion of a group is often associated with multicast communications. A host group is a set of network entities sharing a common identifying multicast address, each member of this group received any data packets addressed to this multicast address by senders (sources) that may or may not be members of the same group and have no knowledge of the groups' membership [2]. This definition implies that, from the sender's point of view, this model reduces the multicast service interface to a unicast one, moreover, multicast routing can utilize network resources more efficiently, as a data packet traverses each link only once, and some of the links are shared [2,3].

In underlying computer networks, the rapid development of real-time multimedia technologies such as videoconferencing, distance learning and coordination require strict quality of service conditions such as cost, bandwidth, packet loss rate, delay, and delay jitter. The stringent delay constraint is enforced on multimedia traffic to guarantee that audio and video data are transformed smoothly to the audience [22]. Besides, multicast is assigned to a condition in which the sender wants to send its data packets to a group of networks or receivers that actually form a multicast group. It is obvious that the benefits of this task include less wastage of bandwidth and network resources, parallelism in the network, transmitter load and reduced network traffic.

Multicast routing based on QoS is an serious issue for network research as well as for high-efficiency and performance networks in future generations. Multicast routing based on the quality of service, therefore, it aims to find an optimized multicast routing tree in order to meet the service quality limitations, it is an NP-complete problem. Establishing a multicast tree could solve multicast routing problems. One of the most important issues of implementing multicast services is the type of tree structure designed to ensure increased quality and efficiency of the multicast tree. There are abundant methods for solving problems related to multicast routing. Among those methods are the exact methods that seek to find optimal solutions for the problems. Other methods are the approximate methods, where one was satisfied with the good quality solutions, without guaranteeing optimality to the port of reduced computation time. In return, the disadvantage is having no information on the quality of the solutions obtained. Thus, hybrid methods combine exact methods and/or approximate methods to create new methods that have given rise to a pseudo-class of methods. There are two ways to solve this problem: An optimal solution at the final moment and An optimal close solution by a heuristic algorithm. The first solution is an optimal solution, but the complexity of the NP-Complete problems makes it impracticable. On the other hand, the second method is a possible way. Thus, the routing algorithm has a significant impact on the development and performance of computer communication networks.

In this paper, our objective is to find solutions to the problems related to the Qos multicast routing, particularly the Delay-Constrained Least-Cost routing problem. Mathematically, the problems can be considered as Steiner tree problems (PST) in a graph. Hence, we propose a novel approach based on Randomized Search Procedure (GRASP) metaheuristic, using the EB heuristic for construction phase, and the Tabu Search algorithm (TS) as a local search procedure used to improve the solution. The remainder of this paper is structured as follows. Section 2 outlines the multicast routing and reviews the existing related work. Section 3 discusses the QoS multicast routing problem. Section 4 defines the Delay-Constrained Least-Cost routing problem. Section 5 describes the proposed method for solving DCLC problem. Section 6 shows simulation and tested experiments. Section 7 summarizes the main contributions and results of this paper.

2 Multicast Routing: Related Work

Since the 1990s, the rapid evolution of numerous real-time multimedia applications has been stimulating the demand for QoS based multicast routing in the underlying computer communication networks. The main goals of QoS multicast routing are to efficiently allocate network resources, balance the network load, reduce congestion hot spots and provide adequate QoS guarantees for end-users of multimedia applications [3].

The traditional unicast model is extremely inefficient for the group-based applications (videoconferencing, shared workspaces, distributed interactive simulations (DIS), software upgrading, and resource location) since the same data is unnecessarily transmitted across the network to each receiver, these applications require the underlying network to satisfy certain quality of service (QoS) multicast communication [2,4]. These QoS requirements include the cost, delay, delay variation, lost and hop count, etc. [4]. The multicast model was proposed to reduce the many unicast connections into a multicast tree for a group of receivers [2]. The phenomenal growth of group communications and quality of service (QoS) aware applications over the Internet has accelerated the need for scalable and efficient network support [2].

Multicasting has emerged as one of the most focused areas in the field of networking. As the technology and popularity of the Internet have grown, applications that require multicasting are becoming more widespread, where information needs to be sent to multiple end-users at the same time in the underlying computer networks [5]. Another interesting recent development has been the emergence of dynamically reconfigurable wireless ad hoc networks to interconnect mobile users for applications ranging from disaster recovery to distributed collaborative computing [5]. In this context, self-organized wireless mobile nodes that share a common wireless channel can work without the support of fixed infrastructure or centralized administration [6]. Multicasting is more complex than in wired networks, the main constraints in these networks are bandwidth limitation and unpredictable host mobility. The challenge is to propose multi-hop routes for multicast routing protocols [7], multi-hopping is usually required due to limited transmission power where each node participates in the network as both host and a router [6].

Many works were carried in the last decade for QoS multicast routing problem such as a method based on genetic algorithms (GA) proposed by Haghighat et al. [8]. In this algorithm, the connectivity matrix of edges is used for genotype representation the matrix where it tells whether or not a specific edge connects the pair of nodes. The initial population is based on the randomized depth-first search algorithm. Also, different heuristics are proposed for reproduction process. The proposed GA-based algorithm overcomes existing algorithms such as BSMA heuristic [9] is the best deterministic Delay Constraint Low-Cost, Wang-GA [10] used for solving the Bandwidth-Delay-Constrained Least-Cost problem which accepted non-uniform and real-valued delay bounds and used the mutation operations to convert the algorithm from a local optimal point into the global solution, moreover this algorithm used a pre-processing mechanism to decrease the search space. In addition, several heuristics have been developed with the GRASP heuristic by N. Skorin-Kapov et al. [11] to solve the Delay-Constrained Multicast Routing (DCMR), however, the neighborhood of the TS-CST algorithm used in the local search procedure proved too restrictive and take both the cost and the time frame into consideration. Zhang et al. [12] proposed a method for least-cost QoS multicast routing based on genetic simulated annealing algorithm NGSA, the genetic algorithm and simulated annealing algorithm are combined to improve the computing performance in this method.

In the last recent years, as in [13], a multi-objective differential evolution algorithm named as MOMR-DE proposed using the modified crossover and mutation operators to build the shortest-path multicast tree. Constraint handling scheme is used to handle QoS constraints. Furthermore, ranking technique, fast non-dominated sorting process and crowding distance sorting process were combined together in order to select the elitism and preserve the diversity of the solutions. The last year, Zhang et al. [14] Combine the solution generation process of Ant Colony Optimization (ACO) algorithm with the cloud model (CM). The cloud model enhances the performance of the ACO algorithm by adjusting the pheromone trail on the edges.

Recently, Askari et al. [23] proposed an algorithm named EMSC to resolve the problem of jointing multicast routing, scheduling, and calling admission control in MRMC-WMNs. Accordingly, this is an efficient cross-layer algorithm that jointly considers the minimum-cost minimum-interference path selection, the QoS requirement of each path in tree construction, and the minimum number of occupied time slots. In a recent paper by Hassan et al. [27], and based on ant colonies, a multi-objective algorithm is developed to construct a multicast tree for data transmission in a computer network. This algorithm simultaneously optimizes the cost, delay and hop (total weight) of the multicast tree. additionally, a novel encoding-free non-dominated sorting genetic algorithm (EF-NSGA) has recently been presented in [25] in order to optimise the Application Layer Multicast (ALM) routing problem as a multi-objective optimization problem to minimize the tree. For achieving encoding-free, genotypes are directly represented as tree-like phenotypes in this algorithm. Accordingly, the genetic operators acting on genotypes, like crossover and mutation, need to be redesigned to adapt the tree-like genotypes. Furthermore, in [26], the authors proposed a discrete artificial fish school algorithm (DAFSA) to optimize multiple co-existing multicast routing problems in a link-capacitated network as a whole rather than sequentially optimize them each in isolation in order to avoid the link-congestion and minimizing their overall tree cost as well.

3 QoS Multicast Routing Problem

The QoS multicast routing problem concerns the search of optimal routing trees in the distributed network, where messages or information are sent from the source node to all destination nodes while meeting all QoS requirements. A common approach constructs a multicast tree structure which covers the source and all terminals nodes, is to bring the problem towards the minimal Steiner tree problem (MST), which aims to look for a tree that covers a set of nodes with a minimum total cost. The Constrained Steiner tree is a well-known NP-complete problem.

Let us represent a network with a graph G = (V, E), where V is a set of n nodes and E a set of m edges. e(i, j) ∈ E a link that associates a weighting function f_W (e) between nodes i and j. The link is bidirectional, i.e. the existence of a link e(i, j) from node i to node j implies the existence of another link e'(j, i). Due to the asymmetric nature of computer networks, it is possible that f_W (e) ≠ f_W (e').

Let us define s a node called the destination node, R a set of nodes where R ⊆ V − {s} is the destination nodes that receive a data stream from source node s. The set R called multicast groups or terminal nodes, relay nodes which are intermediate hops on the paths from the source to destinations. The rest of the paper uses the following notations [4]:

- (i ,j) ∈ E the link from node i to node j, i, j ∈ V.
- $c_{ij} \in \Re^+$ the cost of link (i, j).
- $d_{ij} \in \Re^+$ the delay of link (i, j).
- $z_{ij} \in \Re^+$ the capacity of link (i, j), measured in Mbps.
- $t_{ij} \in \Re^+$ the current traffic of link (i, j), measured in Mbps.
- s ∈ V the source node of a multicast group.
- R ⊆ V-s the set of destinations of a multicast group.
- r_d ∈ R the destinations in a multicast group.
- $\phi \in \Re^+$ the traffic demand (bandwidth requirement) of a multicast request, measured in Mbps.
- T (s, R) the multicast tree with the source node s spanning all destinations r_d ∈ R.
- $p_T(s, r_d)$ ⊆ T(s, R) the path connecting the source node s and a destination r_d ∈ R.
- d($p_T(s, r_d)$) the delay of path $p_T(s, r_d)$.

4 Delay-Constrained Least-Cost Multicast Routing Problem (DCLC)

A variety of Quality of Service (QoS) constraints have been established in real-life applications. That is, cost, packet loss ratio, use of links, bandwidth, delay variation, etc. Multicast Routing Problems (MRPs) based on QoS become much more complicated multi-objective problems when various conflicting objectives are considered simultaneously, these problems are named a Multi-Objective Multicast Routing Problem (MMRP). The problems consist of finding a multicast tree noted T that minimiz the following objectives:

$$min\ Z = C(T) + DM(T) + \alpha(T) + DA(T) \tag{1}$$

where: C(T): the cost of the multicast tree:

$$C(T) = \phi \sum_{(i,j)\in T} c_{i,j} \tag{2}$$

DM(T): the maximal end-to-end delay:

$$DM(T) = Max\left\{d(p_T(s, r_d))\right., r_d \in \Re \tag{3}$$

$$d(p_T(s, r_d)) = \sum_{(i,j)\in p_T(s,r_d)} d_{ij}, r_d \in \Re \tag{4}$$

α(T): the maximal link utilization:

$$\alpha(T) = Max\left\{\frac{\varnothing + t_{ij}}{z_{ij}}\right\}, (i, j) \in T \tag{5}$$

DA(T): the average delay:

$$DA(T) = \frac{1}{|\Re|} \sum_{r_d\in R} d(p_T(s, r_d)) \tag{6}$$

The MMRP is subject to a link capacity constraint as follows:

$$\varnothing + t_{ij} \leqslant z_{ij}, \forall (i, j) \in T(s, \Re) \tag{7}$$

The DCLC problem is a particular case of MMRP problem. It concerns only the two QoS requirements the cost and delay of the multicast tree. The DCLC multicast routing problem is equivalent to the Delay-Constrained Steiner tree (DCST) problem, which is also NP-complete [15]. The objective of the Delay-Constrained Steiner Tree (DCST) Problem is constructing a multicast tree T such that the tree delay is within the delay bound, and the tree cost is minimized.

The end-to-end delay from the source to each destination is the sum of delays along the path, it plays a key role in obtaining feasible solutions in search algorithms, the smaller the delay bound is, the tighter the problem is constrained [3,4]. The objective function of the DCLC can be rewritten as follow:

$$Min\ C(T)\ \ | Delay(T) \leqslant \Delta, T \in T(s, \Re) \tag{8}$$

The inhibiting factor in the Delay-Constrained MRP is the value of Δ, which is the delay bound. This means that the smaller the delay bound is, the stronger is the constraint [11]. We note that in the majority of the literature papers and this one, the same delay bound is applied to all destinations. Otherwise different applications may have different upper bound for each destination.

5 Resolution Methodologies

Based on Greedy Randomized Adaptive Search Procedure (GRASP) combined with the EB heuristic as the initial solution, and the tabu search procedure as the local search method, an algorithm for solving the multicast routing problem have been proposed in this paper. The proposed algorithm is dedicated to the problem of single-objective multicast routing, is reduced to the MStTG problem, where the only constraint is the cost. Furthermore, this algorithm was adapted for resolving the problem of multi-objective multicast routing DCLC, where both constraints: cost and delay are considered.

In order to resolve the multicast routing problem (MRP), we propose in this paper a novel one-off optimization approach based on the GRASP metaheuristic. The greedy randomized adaptive search procedure (GRASP) is a metaheuristic proposed by Feo et al. [24], it consists of an iterative randomized sampling technique in which each iteration provides a solution to the problem. The best incumbent solution overall GRASP iterations is kept as the final result. The procedure builds a solution based upon two phases: a construction phase and a local search phase.

5.1 The Construction Phase

The KMB heuristic [16], proposed by Kou, Markovsky and Bermann, is one of the most efficient approximate method that used to construct the Steiner tree covering all terminals nodes, the problem is well known as an NP-hard. The method based on two main algorithms allowing the development of the shortest paths and minimal spanning tree. There is a practical interest in this heuristic due to its simple implementation, however, it is rarely enough to apply an algorithm such as KMB directly to a multicast routing problem [17]. The produced results by the KMB are not necessarily minimal.

In 2016, M. Fujita et al. have proposed a Steiner tree construction heuristic to improve the KMB algorithm [18]. The heuristic is called Edge Betweenness (EB) (Algorithm 1.1) based on information derived from the edges of the graph and has good performance for various types of Steiner tree problems.

Algorithme 1.1 : EB heuristic.

```
Input: an undirected distance graph G;
Step1. Construct the G' graph from a given network using Eq. 10;
Step2. Construct the complete undirected graph $G_1$;
Step3. Find the minimal spanning tree ($MST_1$) according to Prim;
Step4. The paths in $MST_1$ are replaced by those from the original
    network to construct the Steiner tree;
Step5. Remove from the tree all branches that contain only non-terminals;
Output: a Steiner tree for G;
```

The EB heuristic is based on maximizing the use of low weight edges. This reduces the cost of the Steiner tree result. Therefore, in [18], the authors introduced a parameter allowing to calculate the rate of use of the edges; this parameter is called Edges Betweenness and it represents the centrality of the edges in the network. If an edge is used by many paths between nodes, then that edge has high betweenness [18]. The edge betweenness of the edge e is defined as follows:

$$eb(e) = \frac{\sum_{s=1}^{|V|} \sum_{g=1, g \neq s}^{|V|-1} \frac{I(s,g)}{n(s,g)}}{(|V|-1)(|V|-2)/2} \tag{9}$$

where $|V|$ is the number of nodes, s is a starting node of the shortest paths, g is a terminating node of the shortest paths, I(s, g) is the number of shortest paths between s and g through the edge e and n(s, g) is the number of shortest paths between s and g. M. Fujita et al. [18] define a new cost for the edge as follows, by using the edge betweenness:

$$C_{new} = C(e) - \alpha * eb(e) \tag{10}$$

In Eq. 10, C (e) is the given cost of the edge e, eb (e) is the edge betweenness of the edge e, and α is a controlling parameter that determines the priority between cost and betweenness of the given edge. By using the new cost, an edge is susceptible to be included in the Steiner tree, if it has a low cost and a high edge betweenness. The heuristic gives better results than the KMB algorithm.

The construction phase constructs a solution in two steps. First, in our proposed algorithm (Algorithm 1.4), the EB heuristic is employed to create a so-called restricted candidate list (RCL) of elements that can be added to a partial solution. Second, the elements are randomly selected according to the P factor to create a feasible solution while P $\in$]0,1[is the percentage of components included in the candidate list (RCL) at each iteration, which the factor allows to diversify the initial solution generated.

5.2 The Local Search Phase

Once a solution is obtained in the construction phase, our TS algorithm is applied to improve the current solution with a best solution reached in neighborhood. The metaheuristic Tabu search (TS) is a global optimization method guides a local heuristic search procedure to explore the solution space beyond local optima using intensification and diversification strategy. The heuristic tries to avoid trapping into local optima by using a special memory called tabu list. Any solution which has been recently selected from the best in neighborhood is put into a tabu list so that it becomes (taboo) for a short period of time, depending on the length of the list. The process minimizes the chance of cycling in the same solution and therefore creates more chances of improvement by moving into other space region.

Our procedure TS starts using the EB heuristic as the initial solution, it is done after a fixed number of iterations or a maximum number of continuous iterations without improvement of the best-known solution. The current solution is improved using the best in neighborhood. Furthermore, three movements are used to construct a neighborhood: random movement, path movement, and node movement:

a. Random move (random position - random path): this movement consists of randomly removing a path between two terminals from its position and replacing it with another path that does not belong to the tree.
b. Steiner node move: a no-Steiner node is a node that does not belong to the multicast tree. Moving the Steiner node is a basic transformation that switches the status of one of the elements of the current solution from a Steiner node to a no-Steiner node or vice versa [19]. This movement is an exchange between two non-terminal nodes, the first node is a worse node (a non-Steiner node), but the second is a better node that minimizes the cost of the movement (a Steiner node). This movement was used in the TS-based algorithm for the problem of multicast routing at Delay Constrained and Least-Cost proposed by Skorin Kapov et al. [20].
c. Path move (random position - selected path): this movement is a special case of random movement, it was used in the TS-based algorithm for the problem of multicast routing at DCLC proposed by Youssef et al. [21]. In this paper, to find the alternative path, Dijkstra algorithm is used to calculate the path length between nodes, then Prim algorithm is used for Steiner tree construction. The neighborhood structure based on "Delete and Add" operations inspired from [21] have been chosen for generating neighbors. This movement is a path change operation using two data structures, the solution is encoded as an array of $|M|$ elements, the first represents the current solution, while the second represents a secondary solution. Furthermore, this move consists of inserting the selected path in a random position, the path is defined between a source s and the destination d terminal nodes. The choice of the source s at each iteration is random, and the destination d is one of the other multicast group nodes. At each iteration, we randomly delete one superpath from the encoding of the current solution and then generate different feasible solutions by adding superpaths from the secondary solution which can be at a low cost.

In the context of our paper, the proposed algorithm (Algorithm 1.4) take into account the end-to-end delays in order to solve the DCLC problem.

Algorithme 1.4 : The proposed algorithm.

```
Input: Graph G;
$C_T$ ← the set of edges in the complete graph of the terminal nodes; // the
  costs of these edges are constructed using the Edge Betweenness method
P ← percentage of components to include each iteration; // P ∈ ]0,1[
n ← length of time to do TS;
Best ← initially the best solution is the resulting tree of the $C_T$ ;
repeat
    /* Construction of the initial solution */
    S {}; // Our candidate solution
    repeat
        Sol_$C_T$ ← components in $C_T$ − S arranged in ascending order
          which could be added to S;
        if ( Sol_$C_T$ is empty) then
            S {}; //Try again
        else
            RCL ← the P % lowest cost components in Sol_$C_T$; //
              restricted candidate list
            S ← S ∪ s; // s is component chosen uniformly at random
              from RCL
            Sol_$C_T$ ← Sol_$C_T$ − s;
    until S is a complete solution (Includes all terminal nodes);
    /* Local search */
    for (a period of time n) do
        S' = TS (G, S);
        if ((( cost (S')< cost (Best) )& &( Delay (S')< $\Delta$ )) || (( cost
          (S')= cost (Best))& &( Delay (S')< Delay(Best)))) then
            Best=S';
until Best is the ideal solution or we have run out of time;
Output: Best solution found;
```

At each iteration of our algorithm, we generate a new feasible solution, so the initial solution used for local search is changing at each iteration in order to avoid being trapped into a local optimum. Moreover, to prevent the proposed algorithm stuck into local optima, Edge Betweenness heuristic with α factor is incorporated to our algorithm. Concerning the EB heuristic, the control parameter α, whose optimal value depends on the topological characteristics of the considered network, determines the priority between the cost of the edge and the cost edge betweenness. The value of α can change the cost of the edges, and this changed the Steiner tree result. It is our mechanism for resetting the initial solution. When creating the RCL, a function can be used to evaluate the quality of the elements. According to the percentage factor P, only qualified elements are included in the RCL. Looking for a best solution of the exploration of space continues until a criterion is met, the best solution obtained returned as the final one of our proposed algorithm. This procedure is repeated several times until the ideal solution is achieved or following a maximum number of iterations. Beside that, the ideal solution is the optimal one of benchmark instances from the Steiner library.

6 Experimental Results

To properly test and evaluate the implemented algorithms, we used the SteinLib instances library [29], which is a library of test instances of different sizes for the Minimum Steiner Tree Problem in Graphs. We generate class B instances and transform them into non-oriented graphs. Thus, we will notice that the difficulty of an instance of the problem does not depend only on the size of the problem, there are relatively small instances that are still difficult to resolve.

Table 1. Results of proposed algorithms.

N	OPT	KMB	EB	MRP	DCLCΔ1	DCLCΔ2
B01	82	82^*	82^*	**82***	82^*	**82***
B02	83	90	85	**83***	–	–
B03	138	140	139	**138***	138^*	**138***
B04	59	64	59^*	**59***	59^*	**59***
B05	61	62	62	**61***	61^*	**61***
B06	122	128	127	**123**	123	**122***
B07	111	111^*	111^*	**111***	111^*	**111***
B08	104	104^*	104^*	**104***	104^*	**104***
B09	220	223	221	**220***	220^*	**220***
B10	86	98	98	**86***	86^*	**86***
B11	88	92	90	**88***	88^*	**89**
B12	174	174^*	174^*	**174***	174^*	**174***
B13	165	175	175	**170**	170	**170**
B14	235	238	238	**238**	238	**237**
B15	318	325	322	**318***	318^*	**318***
B16	127	137	137	**127***	127^*	**127***
B17	131	134	133	**132**	132	**133**
B18	218	223	223	**220**	220	**221**

(with DCLC $\Delta_1 = \infty$ and $\Delta_2 = 1.1 * \text{Delay(OPT)}$).

Based on proposed algorithm, combined with the EB and TS methods, an approach is proposed in this paper for the first time to solve the DCLC problem. In Table 1, we presente the final results for the proposed algorithms. Computational results demonstrate the effectiveness of our proposed algorithm for this problem. OPT, denotes the cost of the optimal solution for each unconstrained Steiner tree instance, and values marked with * indicate optimal solutions.

From Table 1, we can observe that the KMB heuristic has given good results (B01, B07, B08, B12), but it does not ensure optimal solutions for all instances. It is found that the EB method has obtained improved solutions than the KMB algorithm, which implies that several edges are shared by the paths in the Steiner tree obtained using the new cost calculated from the cost and the edge betweenness. Concerning the EB heuristic, the control parameter α, in which optimal value depends on the topological characteristics of the considered network, determines the priority between the cost of the edge and the cost edge betweenness. Also, we consider that the maximum cost of the edge is an important factor in determining the optimal value of α. It was necessary to determine a good balance between parameter α and the number of iterations of the proposed algorithm, for at each iteration, the value of α can change the cost of the edges, and thus changed the Steiner tree result. Recall that candidates in the restricted candidate list (RCL) are chosen according to the cost of adding them to the existing tree in ascending order.

In the column MRP, we represent the results based on our proposed algorithm for solving the single-objective multicast routing problem (reduced to the MStTG problem). The proposed algorithm gave the optimal solution to 13 test problems for the MStTG problem. Therefore, the results provide a good indication of the competency of the proposed algorithm compared to the KMB and EB. However, some instances take a bit of time to run. The initial solution is based on EB heuristic and the choice of position in this neighborhood that minimizes cost. The mechanism consists of prohibiting the return to the last positions explored. The “Size of the taboo list” parameter can range between [10, 50], and the number of iterations in our first tests can go up to 1000 iterations. Through the random part of the initial solution, based Edge Betweenness heuristic allows to differ the solutions generated, but they are still of good quality since the random choice is made among a set of good candidates. The movements using in the TS metaheuristic, applies to the realizable solution to see if it is still possible to improve this solution.

Regarding QoS multicasting with a bounded end-to-end delay, to evaluate the performance of the proposed algorithm for solving the DCLC problem, a large number of simulations have been conducted on the benchmark Steinb instances where end-to-end delay is added. The results obtained are presented in the columns DCLC in Table 1. This algorithm have been executed with a sufficiently high value of the delay bound Δ so as not to act as a constraint (The link delay cannot be set to ∞). That means simulating the MStTG problem. Also, we set the delay bound $\Delta_2 = 1.1 * \text{Delay}(\text{OPT})$, where OPT is the multicast tree of the optimal solution with the minimal cost and delay.

Lastly, to evaluate the performance of the proposed algorithm, we compare the results obtained with other existing in the literature which are implemented GRASP algorithm. In Table 2, we compare the obtained results with two algorithms: GRASP−CST (Greedy Randomized Adaptive Search

Procedure−Constrained Steiner Tree) proposed in the literature [11], and GRASP−VND (Greedy Random Adaptive Search Procedure−Variable Neighborhood Search) mentioned in [28] in terms of the average tree cost for this set of benchmark problems. All experiment results demonstrate that the proposed algorithm has the overall best performance compared with other existing results in the literature through the use of EB heuristic combained with the movements of tabu searh in the implementation of our proposed algorithm.

Table 2. Ccomparison of the proposed algorithm with GRASP−CST and GRASP−VNS.

N	DCLCΔ2	GRASP−CST	GRASP−VNS
B01	**82***	82	82
B02	–	83	83
B03	**138***	138	138
B04	**59***	59	59
B05	**61***	62	61
B06	**122***	124	122.2
B07	**111***	111	111
B08	**104***	104	104
B09	**220***	221	220
B10	**86***	86	86.5
B11	**89**	88	88
B12	**174***	174	174
B13	**170**	165	169.5
B14	**237**	235	235
B15	**318***	318	319.5
B16	**127***	132	127
B17	**133**	131	131.5
B18	**221**	219	218.5

(with DCLC Δ_2 = 1.1 * Delay(OPT)).

7 Conclusion and Future Work

In this paper, we try to find optimal solutions for single-objective multicast routing problem and multi-objective multicast routing problem, where both constraints: cost and delay are considered. To achieve this goal, we investigate algorithms such as heuristic EB and Tabu Search. Our proposition is based on the Greedy Randomized Search Procedure using the EB heuristic to construct the initial solution, then we used the movements of Tabu Srearch to diversify the research space. The chosen algorithms are adapted, implemented, and tested via an extensive set of experiments on a number of benchmark instances from the Steiner library. The results obtained show that the EB algorithm gives improvements to the KMB algorithm. Thus, the proposed method gives good results by contributing to EB heuristic and the movements of TS. After, we compared these algorithms to the optimal solutions from the Steiner library in order to study the efficiency of these algorithms to optimize the total cost of the constructed Steiner tree. Finally, we test our proposed algorithm on the DCLC problem and compare the results obtained with other existing one in the literature. The results provide a good indication of the competency of our proposed algorithm. In our future work, we intend to investigate the influence of our algorithm for solving MRPs with a wider range of real-world features. It is also interesting to extend our algorithms to solve other multi-objective optimization problems with reduced execution time.

References

1. Diot, C., Levine, B.N., Lyles, B., Kassem, H., Balensiefen, D.: Deployment issues for the IP multicast service and architecture. IEEE Netw. **14**(1), 78–88 (2000)
2. Striegel, A., Manimaran, G.: A survey of QoS multicasting issues. IEEE Commun. Mag. **40**(6), 82–87 (2002)
3. Xu, Y.: Metaheuristic approaches for QoS multicast routing problems. Ph.D. thesis, University of Nottingham Nottingham (2011)
4. Qu, R., Xu, Y., Kendall, G.: A variable neighborhood descent search algorithm for delay-constrained least-cost multicast routing. In: Stützle, T. (ed.) LION 2009. LNCS, vol. 5851, pp. 15–29. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-11169-3_2
5. Lee, S.J., Gerla, M., Chiang, C.C.: On-demand multicast routing protocol. In: 1999 IEEE Wireless Communications and Networking Conference, WCNC (Cat. no. 99TH8466), vol. 3, pp. 1298–1302. IEEE (1999)
6. Boppana, A.: A scalable simplified multicast forwarding for mobile ad-hoc networks (2011)
7. Chelius, G., Fleury, É.: Performance evaluation of multicast trees in adhoc networks (2002)
8. Haghighat, A.T., Faez, K., Dehghan, M., Mowlaei, A., Ghahremani, Y.: GA-based heuristic algorithms for QoS based multicast routing. Knowl.-Based Syst. **16**(5–6), 305–312 (2003)
9. Parsa, M., Zhu, Q., Garcia-Luna-Aceves, J.J.: An iterative algorithm for delay-constrained minimum-cost multicasting. IEEE/ACM Trans. Netw. **6**(4), 461–474 (1998)
10. Zhengying, W., Bingxin, S., Erdun, Z.: Bandwidth-delay-constrained least-cost multicast routing based on heuristic genetic algorithm. Comput. Commun. **24**(7–8), 685–692 (2001)
11. Skorin-Kapov, N., Kos, M.: A grasp heuristic for the delay-constrained multicast routing problem. Telecommun. Syst. **32**(1), 55–69 (2006). https://doi.org/10.1007/s11235-006-8202-2
12. Zhang, L., Cai, L.B., Li, M., Wang, F.H.: A method for least-cost QoS multicast routing based on genetic simulated annealing algorithm. Comput. Commun. **32**(1), 105–110 (2009)

13. Wei, W., Qin, Y., Cai, Z.: A multi-objective multicast routing optimization based on differential evolution in MANET. Int. J. Intell. Comput. Cybern. **11**(1), 121–140 (2018)
14. Zhang, X., Shen, X., Yu, Z.: A novel hybrid ant colony optimization for a multicast routing problem. Algorithms **12**(1), 18 (2019)
15. Matta, I., Guo, L.: QDMR: an efficient QoS dependent multicast routing algorithm. J. Commun. Netw. **2**(2), 168–176 (2000)
16. Kou, L., Markowsky, G., Berman, L.: A fast algorithm for Steiner trees. Acta Informatica **15**(2), 141–145 (1981). https://doi.org/10.1007/BF00288961
17. Oliveira, C.A., Pardalos, P.M.: Mathematical Aspects of Network Routing Optimization. Springer, New York (2011). https://doi.org/10.1007/978-1-4614-0311-1
18. Fujita, M., Kimura, T., Jin'no, K.: An effective construction algorithm for the Steiner tree problem based on edge betweenness. J. Sig. Process. **20**(4), 145–148 (2016)
19. Ghaboosi, N., Haghighat, A.T.: Tabu search based algorithms for bandwidth-delay-constrained least-cost multicast routing. Telecommun. Syst. **34**(3–4), 147–166 (2007). https://doi.org/10.1007/s11235-007-9031-7
20. Skorin-Kapov, N., Kos, M.: The application of Steiner trees to delay constrained multicast routing: a tabu search approach. In: 2003 Proceedings of the 7th International Conference on Telecommunications, ConTEL 2003, vol. 2, pp. 443–448. IEEE (2003)
21. Youssef, H., Al-Mulhem, A., Sait, S.M., Tahir, M.A.: QoS-driven multicast tree generation using tabu search. Comput. Commun. **25**(11–12), 1140–1149 (2002)
22. Zhang, K., Wang, H., Liu, F.: Multicast routing for delay and delay variation bounded Steiner tree using simulated annealing. In: Proceedings of the 2005 IEEE Networking, Sensing and Control, pp. 682–687. IEEE (2005)
23. Askari, Z., Avokh, A.: EMSC: a joint multicast routing, scheduling, and call admission control in multi-radio multi-channel WMNs. Front. Comput. Sci. **14**(5), 1–16 (2020). https://doi.org/10.1007/s11704-019-8199-9
24. Feo, T.A., Resende, M.G.: Greedy randomized adaptive search procedures. J. Global Optim. **6**(2), 109–133 (1995). https://doi.org/10.1007/BF01096763
25. Liu, Q., Tang, R., Ren, H., Pei, Y.: Optimizing multicast routing tree on application layer via an encoding-free non-dominated sorting genetic algorithm. Appl. Intell. **50**(3), 759–777 (2019). https://doi.org/10.1007/s10489-019-01547-9
26. Liu, Q., Ren, H.P., Tang, R.J., Yao, J.L.: Optimizing co-existing multicast routing trees in IP network via discrete artificial fish school algorithm. Knowl.-Based Syst. **191**, 105276 (2020)
27. Hassan, M., Hamid, A., Alkinani, M.: Ant colony optimization for multi-objective multicast routing. Comput. Mater. Continua **63**(3), 1159–1173 (2020)
28. Xu, Y., Qu, R.: A GRASP approach for the delayconstrained multicast routing problem. In: Proceedings of the 4th Multidisplinary International Scheduling Conference (MISTA4), Dublin, Ireland, pp. 93–104 (2009)
29. Koch, T., Martin, A., Voß, S.: SteinLib: an updated library on steiner tree problems in graphs. In: Cheng, X.Z., Du, D.Z. (eds.) Steiner Trees in Industry. COOP, vol. 11, pp. 285–325. Springer, Boston (2001). https://doi.org/10.1007/978-1-4613-0255-1_9