Solving the Delay-Constrained Least-Cost routing problem using Tabu Search with Edge Betweenness

1ˢᵗ Amina Boudjelida
Laboratory of Mathematics and Applied Mathematics
University of Jijel
of Mathematics and Applied Mathematics (LMAM) Jijel, Algeria
boudjelidaamina@gmail.com

2ⁿᵈ Ali Lemouari
Laboratory of Mathematics and Applied Mathematics
University of Jijel
of Mathematics and Applied Mathematics (LMAM) Jijel, Algeria
lemouari_ali@yahoo.fr

Abstract—**Multicast routing consists of concurrently sending the same information from a source to a subset of all possible destinations in a computer network thus becomes an important technology communication. To solve the problem, a current approach for efficiently supporting a multicast session in a network consists of 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, the problem is NP-hard. In this paper, we investigate metaheuristics approaches for the Delay-Constrained Least-Cost (DCLC) problem, we propose a novel algorithm based on Tabu Search procedure with the Edge Betweenness (EB). The EB heuristic used first to improve KMB heuristic, able to measure the edge value to being included in a given path. The obtained solution improved using the tabu search method. The performance of the proposed algorithm is evaluated by experiments on a number of benchmark instances from the Steiner library. Experimental results show that the proposed metaheuristic gives competitive results in terms of cost and delay compared to the optimal results in Steiner library and other existing algorithms in the literature.**

Index Terms—**Multicast Routing, DCLC, QOS, Optimization, Heuristic, Metaheuristic**

I. 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 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, all receiving 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].

As a major Internet activity, multicast is an appropriate medium for multimedia technologies, such as videoconferencing, distance learning and coordination, which require strict quality of service conditions such as bandwidth, delay, delay variation, etc. 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.

The subject of multicast routing on QoS is an important issue for network research and a serious problem for future generations of high efficiency and performance networks. Multicast routing based on the quality of service, therefore, aims to find an optimized multicast routing tree, in order to meet service quality limitations, it is an NP-complete problem. Establishing a multicast tree could solve multicast routing problems. One of the most important issues for 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 these 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. The disadvantage is to have in return 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 side, 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 multicast routing, the Delay-Constrained Least-Cost routing problem. Mathematically, the problems can be considered as Steiner tree problems (PST) in a graph. We propose a tabu search as a metaheuristic based on the EB heuristic as a initial solution and the movements of tabu search algorithm used to improve the solution. The remainder of this paper is structured as follows. Section II outlines the multicast routing and reviews the existing related work. Section III discusses the QoS multicast routing problem. Section IV defines the Delay-Constrained Least-Cost routing problem. Section V describes the proposed method for solving DCLC problem. Section VI shows simulation and tested experiments. Section VII summarizes the main contributions and results of this paper.

II. 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 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], Sun-GA [10] and Wang-GA [11]. In addition, several heuristics have been developed with the GRASP heuristic by N. Skorin-Kapov et al [25] to solve the Delay-Constrained Multicast Routing (DCMR). Experiments on maximum nodes sized problems (100 nodes) from benchmark of the literature and comparison with existing algorithms have been done. Tseng et al [13] proposed an ant colony-based algorithm to solve the degree and delay-constrained broadcasting problem with minimum-cost. The algorithm consists of three mechanisms: the spanning tree construction, the local and global updating rules of the pheromone trail. Such a communication scheme can be regarded as a broadcasting problem with degree and delay constraints in the overlay network. Zhang et al [14] proposed a method for least-cost QoS multicast routing based on genetic simulated annealing algorithm NGSA, the genetic algorithm et simulated annealing algorithm are combined to improve the computing performance in this method. Recently as in [15], a multi-objective differential evolution algorithm named as MOMR-DE is proposed using the modified crossover and mutation operators to build the shortest-path multicast tree. Constraint handling scheme is used to handle QoS constraints. In addition, 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. In the last year, Zhang et al [16] 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. In a recent paper by Hassan et al [24], 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.

III. 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.
- $c_{ij} ∈ \Re^+$ the cost of link (i, j).
- $d_{ij} ∈ \Re^+$ the delay of link (i, j).
- $z_{ij} ∈ \Re^+$ the capacity of link (i, j), measured in Mbps.
- $t_{ij} ∈ \Re^+$ 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.
- $\phi ∈ \Re^+$ 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 $p_T(s, r_d)$.

IV. 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 minimizing the following objectives:

$$min\, Z = C(T) + DM(T) + \alpha(T) + DA(T) \qquad (1)$$

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

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

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

$$DM(T) = Max\,\{d(p_T(s, r_d))\,, r_d \in \Re \qquad (3)$$

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

α(T): the maximal link utilization:

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

DA(T): the average delay :

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

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

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

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 [17]. The objective of the Delay-Constrained Steiner Tree (DCST) Problem is to construct 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:

$$Min\, C(T) \mid Delay(T) \leqslant \Delta, T \in T(s, \Re) \qquad (8)$$

Where Δ is the delay bound, 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.

V. Resolution Methodologies

Based on the tabu search procedure combined with the EB heuristic as the initial solution, two algorithms for solving the multicast routing problem have been proposed in this paper. The first algorithm TSEB-MRP is dedicated to the problem of single-objective multicast routing, where the only constraint is the cost. Therefore, the second algorithm TSEB-DCLC is inspired by our first algorithm TSEB-MRP it is intended for the problem of multi-objective DCLC problem, where both constraints: cost and delay are considered.

A. Edge Betweenness Heuristic

The KMB heuristic [18], proposed by Kou, Markovsky and Bermann, is one of the most efficient approximate methods 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 [19]. The

produced results by the KMB heuristics are not necessarily minimal.

M. Fujita et al have proposed a Steiner tree construction heuristic to improve the KMB algorithm [20]. The heuristic is called Edge Betweenness (EB) (Algorithm .1) based on information derived from the edges of the graph and has good performance for various types of Steiner tree problems. The EB heuristic is based on maximizing the use of low weight edges. This reduces the cost of the Steiner tree result. Therefore, in [20], the authors introduced a parameter allowing to calculate the rate of use of the edges, this parameter is called Edges Betweenness, 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 [20]. The edge betweenness of the edge e is defined as follows:

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

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 [20] define a new cost for the edge as follows, by using the edge betweenness:

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

In (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 conventional KMB algorithm.

Algorithm .1: EB.

Input: an undirected distance graph G;
Step1. Construct the G' graph from a given network using (10);
Step2. Construct the complete undirected graph G1;
Step3. Find the minimal spanning tree (MST1) according to Prim;
Step4. The paths in MST1 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.;

B. Tabu Search Algorithm

The metaheuristic Tabu search (TS) is a global optimization method guides a local search procedure to explore the solution space beyond local optima using intensification and diversification strategy. The method 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 putted into a tabu list so that it becomes 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. We apply the TS in order to resolve the Delay Constrained and Least-Cost problem. The procedure TSEB-MRP 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 removing randomly 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 [21]. 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 [22].

c. Path move (random position - selected path): this 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 [23]. 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 [23] have 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, we will propose an algorithm TSEB-DCLC (Algorithm .3) based on the previously defined methods TS and EB (TSEB-MRP) for solving the DCLC problem. For this purpose, these algorithms will be adapted to take into account the end-to-end delays. In our proposed algorithm, the initial solution S is generated according to the EB heuristic. At each iteration, we generate a random number

---

Algorithm .3: TSEB for DCLC problem.

l ← length of taboo list;
n ← number of movements;
S ← the initial candidate solution is the tree resulting
 from the heuristic **EB**;
nbrIterMax ← total number of iterations;
nbrIterLocal ← 0;
Best ← S;
L ← { }; // L: taboo list with a maximum length equal
 to l

repeat
  if *(length (L) > l)* **then**
    Remove old element from L;

  β ← random(0,1);
  if *($\beta \in [0, \frac{1}{3}[$)* **then**
    R ← Random_move (S, L) ;
  else
    if *($\beta \in [\frac{1}{3}, \frac{2}{3}[$)* **then**
      R ← Steiner_node_move(S, L);
    else
      R ← Path_move (S, L);

  for *(n time competition)* **do**
    W ← Path_move (S, L);
    if *((((cost (W)< cost (R))& &(Delay (W)<*
    Δ)) || ((cost (W)= cost (R))& &(Delay
    (W)< Delay(R)))& & (W $\notin$ L)) **then**
      R ← W;

  if *(R$\notin$L)* **then**
    S ← R;
    Insert (R ,L);
  if *(((cost (s)< cost (Best))& &(Delay (s)< Δ))*
  || ((cost (s)= cost (Best))& &(Delay (W)<
  Delay(Best)))) **then**
    Best ← S, nbrIterLocal ← 0;
  else
    nbrIterLocal ← nbrIterLocal+1;
  // Resetting the initial solution
  if *(nbrIterLocal = nbrIterMax)& & (Best is not*
  the ideal)) **then**
    γ ← random(0,1);
    if *($\gamma \in [0, \frac{1}{3}[$)* **then**
      $\alpha = 0.3$;
    else
      if *($\gamma \in [\frac{1}{3}, \frac{2}{3}[$)* **then**
        $\alpha = 0.6$;
      else
        $\alpha = 0.9$;
    S = **EB**(G, α);

until *Best is the ideal solution or we don't have the*
time;
Return Best;

β in order to find a neighbor to S using Random, Steiner or Path mouvement, it is applied to diversify the incoming solution to the local search. Then, the solution obtained R entered into competition with another random local solution W in order to exploit the local search space . Furthermore, to prevent the TSEB-DCLC algorithm stuck into local optima, EB 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. 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 solution of benchmark instances from the Steiner library.

VI. EXPERIMENTAL RESULTS

To properly test and evaluate the implemented algorithms, we used the SteinLib instances library, 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. For the DCLC problem, we generate instances class B in the SteinLib, because, there are no benchmark problem instances for DCLC problem in the literature. Based on Tabu Search method, combined with

TABLE I
RESULTS OF PROPOSED ALGORITHMS ON CLASS B INSTANCES
COMPARED TO $TS-CST$ AND $SSPR-TS$.

N	OPT	KMB	KMB_{EB}	$TSEB_{MRP}$	$TSEB_{\Delta 1}$	$TSEB_{\Delta 2}$	TS_{CST}	$SSPR_{TS}$
B01	82	82*	82*	**82***	82*	**82***	82	82
B02	83	90	85	**83***	-	**-**	83	86.3
B03	138	140	139	**138***	138*	**138***	138	138
B04	59	64	59*	**59***	59*	**59***	59	59
B05	61	62	62	**61***	61*	**61***	76	61
B06	122	128	127	**123**	123	**124***	126	122.1
B07	111	111*	111*	**111***	111*	**111***	111	111
B08	104	104*	104*	**104***	104*	**104***	104	104
B09	220	223	221	**220***	220*	**220***	231	220
B10	86	98	98	**86***	86*	**86***	86	86
B11	88	92	90	**88***	88*	**88***	92	88
B12	174	174*	174*	**174***	174*	**174***	180	174
B13	165	175	175	**170**	170	**172**	165	169.2
B14	235	238	238	**235***	235*	**237**	239	258.8
B15	318	325	322	**318***	318	**318***	330	319.5
B16	127	137	137	**131***	131	**137**	149	133.5
B17	131	134	133	**131***	131	**132**	131	131
B18	218	223	223	**218***	218	**220**	219	128.2

(with $\Delta_1 = \infty$ and $\Delta_2 = 1.1 * Delay(OPT)$).

the Edge Betweenness heuristic, an approach is proposed in this paper for the first time to solve the DCLC problem. Computational results demonstrate the effectiveness of the proposed TSEB-DCLC algorithm for this problem. In TABLE I we presente the final results for the proposed algorithms. 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. In addition, we consider that the maximum cost of the edge is an important factor in determining the optimal value of α. OPT, denotes the cost of the optimal solution for each unconstrained Steiner tree instance, values marked with * indicate optimal solutions. All experiments have been implemented in JAVA.

From TABLE I, we can observe that the KMB heuristic has given good results, 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. In the column TSEB-MRP, we show that the results based on the TS method show good performance compared to the KMB and EB heuristics. The initial solution is based on EB heuristic and the choice of position in this neighborhood that minimizes cost. The mechanism consists in prohibiting the return to the last positions explored. The "Size of the taboo list" parameter can range between [10, 30], and the number of iterations in our first tests can go up to 1000 iterations. Through the random part of the initial solution, based EB algorithm, 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 method, applies to the realizable solution to see if it is still possible to improve this solution. In order to evaluate the performance of the proposed TSEB-DCLC algorithm, a large number of simulations have been conducted on the benchmark Steinb instances, then we show the results obtained in TABLE I for the DCLC problem where end-to-end delay is added. Those algorithms have been executed with a sufficiently high value of the delay bound Δ to not play the role of constraint (The link delay cannot really be set to ∞). The solutions obtained are the solutions for the single-objective case. In the last, we compare the obtained results with the algorithms: TS$-$CST (Tabu Search- Constrained Steiner Tree) proposed in the literature [25] , and SSPR$-$TS (Scatter Search and Path Reli- Tabu Search) mentioned in [26]. The results demonstrate the efficiency and effectiveness of our proposed algorithm for solving the multicast DCLC routing problem compared to other existing algorithms in the literature. Moreover, TSEB-DCLC produces results that are better than those of the literature. However, some instances take a bit of time to run.

VII. CONCLUSION AND FUTURE WORK

In this paper, a new heuristic named TSEB is introduced for single-objective multicast routing problem (MRP) and adapted for multi-objective DCLC problem where both constraints: cost and delay are considered. Our TSEB-MRP proposition is based on the TS procedure which uses the Edge Betweenness heuristic to construct the initial solution, and uses the movements of TS 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.

Furthermore, we compared these algorithms with the proposed TSEB-MRP method in order to study the efficiency of these algorithms in multicast routing to optimize the total cost of the constructed Steiner tree. In this study, the TSEB-MRP method gives good results by contributing to EB heuristic and the movements of TS. Finally, we test our proposal TSEB-DCLC on the DCLC problem, and we compare it to the optimal results and other existing algorithms in the literature. In our future work, we intend to investigate the influence of algorithm TSEB 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 by taking into account the reduction in execution time.

REFERENCES

[1] Christophe Diot, Brian Neil Levine, Bryan Lyles, Hassan Kassem, and Doug Balensiefen. Deployment issues for the ip multicast service and architecture. IEEE network, 14(1):78-88, 2000.

[2] Aaron Striegel and G Manimaran. A survey of qos multicasting issues. IEEE Communications magazine, 40(6):82-87, 2002.

[3] Ying Xu. Metaheuristic approaches for QoS multicast routing problems. PhD thesis, University of Nottingham Nottingham, 2011.

[4] Rong Qu, Ying Xu, and Graham Kendall. A variable neighborhood descent search algorithm for delay-constrained least-cost multicast routing. In International Conference on Learning and Intelligent Optimization, pages 15-29. Springer, 2009.

[5] Sung-Ju Lee, Mario Gerla, and Ching-Chuan Chiang. On-demand multicast routing protocol. In WCNC. 1999 IEEE Wireless Communications and Networking Conference (Cat. No. 99TH8466), volume 3, pages 1298-1302. IEEE, 1999.

[6] Aneesha Boppana et al. A scalable simplified multicast forwarding for mobile ad-hoc networks. 2011.

[7] Guillaume Chelius and Éric Fleury. Performance evaluation of multicast trees in adhoc networks. 2002.

[8] Abolfazl Toroghi Haghighat, Karim Faez, Mehdi Dehghan, Amir Mowlaei, and Y Ghahremani. Gabased heuristic algorithms for qos based multicast routing. Knowledge-Based Systems, 16(5-6):305-312, 2003.

[9] Mehrdad Parsa, Qing Zhu, and JJ Garcia-Luna-Aceves. An iterative algorithm for delay-constrained minimum-cost multicasting. IEEE/ACM transactions on networking, 6(4):461-474, 1998.

[10] Quan Sun and Horst Langendörfer. An efficient delay-constrained multicast routing algorithm. Journal of High Speed Networks, 7(1):43-55, 1998.

[11] Wang Zhengying, Shi Bingxin, and Zhao Erdun. Bandwidth-delay-constrained least-cost multicast routing based on heuristic genetic algorithm. Computer communications, 24(7-8):685-692, 2001.

[12] Nina Skorin-Kapov and Mladen Kos. A grasp heuristic for the delay-constrained multicast routing problem. Telecommunication Systems, 32(1):55-69, 2006.

[13] Sheng-Yuan Tseng, Chang-Chun Lin, and Yueh-Min Huang. Ant colony-based algorithm for constructing broadcasting tree with degree and delay constraints. Expert Systems with Applications, 35(3):1473- 1481, 2008.

[14] Li Zhang, Lian-bo Cai, Meng Li, and Fa-hui Wang. A method for least-cost qos multicast routing based on genetic simulated annealing algorithm. Computer Communications, 32(1):105-110, 2009.

[15] Wenhong Wei, Yong Qin, and Zhaoquan Cai. A multi-objective multicast routing optimization based on differential evolution in manet. International Journal of Intelligent Computing and Cybernetics, 2018.

[16] Xiaoxia Zhang, Xin Shen, and Ziqiao Yu. A novel hybrid ant colony optimization for a multicast routing problem. Algorithms, 12(1):18, 2019.

[17] Ibrahim Matta and Liang Guo. Qdmr: An efficient qos dependent multicast routing algorithm. Journal of Communications and Networks, 2(2):168-176, 2000.

[18] L Kou, George Markowsky, and Leonard Berman. A fast algorithm for steiner trees. Acta informatica, 15(2):141-145, 1981.

[19] Carlos AS Oliveira and Panos M Pardalos. Mathematical aspects of network routing optimization. Springer, 2011.

[20] Misa Fujita, Takayuki Kimura, and Kenya Jin'no. An effective construction algorithm for the steiner tree problem based on edge betweenness. Journal of Signal Processing, 20(4):145-148, 2016.

[21] Nejla Ghaboosi and Abolfazl T Haghighat. Tabu search based algorithms for bandwidth-delayconstrained least-cost multicast routing. Telecommunication Systems, 34(3-4):147-166, 2007.

[22] Nina Skorin-Kapov and Mladen Kos. The application of steiner trees to delay constrained multicast routing: a tabu search approach. In Proceedings of the 7th International Conference on Telecommunications, 2003. ConTEL 2003., volume 2, pages 443-448. IEEE, 2003.

[23] Habib Youssef, Abdulaziz Al-Mulhem, Sadiq M Sait, and Muhammad Atif Tahir. Qos-driven multicast tree generation using tabu search. Computer Communications, 25(11-12):1140-1149, 2002.

[24] Moatamad Hassan, Ahmed Hamid, and Monagi Alkinani. Ant colony optimization for multi-objective multicast routing. Computers, Materials & Continua, 63(3):1159-1173, 2020.

[25] Nina Skorin-Kapov and Mladen Kos. A grasp heuristic for the delay-constrained multicast routing problem. Telecommunication Systems, 32(1):55-69, 2006.

[26] Ying Xu and Rong Qu. A hybrid scatter search meta-heuristic for delay-constrained multicast routing problems. Applied Intelligence, 36(1):229-241, 2012.