Distributed Approximation Algorithms for Longest-Lived Multicast in WANETs with Directional Antennas

Song Guo, *Member, IEEE,* Victor Leung, *Fellow, IEEE,* and Xiaohong Jiang, *Senior Member, IEEE*

***Abstract*—We consider the lifetime optimization problem for multicasting in wireless ad hoc networks, in which each node is equipped with a directional antenna and has limited energy supplies. Several distributed algorithms proposed recently are especially beneficial to a resource-constrained wireless ad hoc network. In this paper, we propose a new distributed algorithm and investigate its theoretical performance compared to existing distributed algorithms. We use a graph theoretic approach to obtain the upper bound of the approximation ratio for a group of distributed algorithms. In particular, the derived upper bound in a closed form for each algorithm provides a sufficient condition to determine if the obtained solutions can reach optimum. Both theoretical and experimental performance analysis show that the new algorithm outperforms other proposals in terms of providing long-lived multicast tree.**

***Index Terms*—Wireless ad hoc network, approximation algorithm, multicast, directional antenna.**

I. Introduction

THERE is an increasing interest in WANETs (Wireless Ad Hoc Networks) in many application domains where instant infrastructure is needed and no central backbone system and administration exist. Each communicating node in these networks acts as a router in addition to a host in order to communicate with each other over a limited number of shared radio channels. A communication session can be achieved either through a single-hop transmission if the communicating nodes are close enough to each other, or by relaying through intermediate nodes otherwise. Energy conservation is of paramount importance for the wide deployment of WANETs in the forms of MANETs (Mobile Ad Hoc Networks) and WSNs (Wireless Sensor Networks) due to their potentially extensive civil and military applications. Multicasting plays an important role in typical WANETs where bandwidth is scarce and hosts have limited battery power. In addition, many routing protocols for MANETs need a broadcast / multicast as a communication primitive to update their states and maintain the routes between nodes. Multicast is also widely used in WSNs to disseminate information, *e.g.*, environmental changes, to other nodes in the network. Therefore, it is essential to develop efficient multicast protocols that are optimized for energy consumption. In particular, energy efficient communication in WANETs with directional antennas has received much more attention. This is because directional communications can save transmission power significantly by concentrating RF (Radio Frequency) energy only to where it is needed.

There are two energy-aware metrics and their corresponding problem formulations that have been most widely studied: (1) to minimize the energy consumption and (2) to maximize the lifetime. Both problems have received substantial attentions, *e.g.*, the work [1-5] for the first problem and [6-19] for the second. Although they are closely related, these two problems are not equivalent, *i.e.*, a multicast tree with minimum energy consumption cannot guarantee that it has the maximum lifetime. In order to maximize the multicast lifetime, a special attention should be taken to the impact of residual battery energy at each node [16, 17]. Therefore, we focus on the second problem in this paper.

Considering the resource limitation of WANETs, we present a distributed algorithm for this problem. We apply a graph theoretic approach to analyze its theoretical performance, in terms of approximation ratio, and our theoretical findings show that it is a constant-factor approximation algorithm, *i.e.*, its approximation ratios is bounded by a finite real number. This may help us understand the simulation results that significantly outperforms most of the centralized algorithms, *e.g.*, [16, 17]. Furthermore, a sufficient optimality condition is obtained from the upper bound of its approximation ration given in a close-form. By comparing results on both theoretical and simulation analysis for various algorithms, we also provide insights into the real performance behaviors of a group of distributed algorithms.

The rest of this paper is organized as follows. Section 2 gives a brief overview of related work. Section 3 describes our system model and problem formulation. Section 4 presents a new distributed algorithm for long-lived directional multicast communications. Section 5 studies its theoretical performance. Section 6 evaluates the real performance of a group of algorithms. Section 7 summarizes our findings. For the convenience of the readers, the notations used in this paper are listed in Table 1.

II. Related Work

When the optimization is on the multicast lifetime, the maximum-lifetime multicast tree algorithms are especially of interest. The lifetime of a multicast tree is typically defined as

Manuscript received June 15, 2009; revised November 4, 2009 and March 4, 2010; accepted March 7, 2010. The associate editor coordinating the review of this paper and approving it for publication was C. Yang.

S. Guo is with the School of Computer Science and Engineering, The University of Aizu, Aizu-Wakamatsu, Fukushima, Japan (e-mail: sguo@u-aizu.ac.jp).

V. Leung is with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada (e-mail: vleung@ece.ubc.ca).

X. Jiang is with the School of Systems Information Science, Future University Hakodate, Hakodate, Hokkaido, Japan (e-mail: jiang@fun.ac.jp).

Digital Object Identifier 10.1109/TWC.2010.07.090897

TABLE I: Notations

Notation	Description
A	an arc set corresponding to the unidirectional wireless communication link
$A(T_s)$	the arc set of a multicast tree T_s
C_X	all arcs crossing a node partition $(X, N-X)$ such that $s \in X$ and $D \not\subset X$
D	a set of destination nodes
G	a directed graph modeling the wireless network
M	a multicast group, $M = \{s\} \cup D$
N	a node set
$N(T_s)$	the node set of a multicast tree T_s
T_s	a multicast tree of $G(N, A)$ rooted at node s
T_s^i	an intermediate tree constructed by a distributed algorithm after the $(i+1)$-th node or equivalently the i-th arc $(i \geq 0)$ is added into the tree
m	the size of a multicast group M
n	the number of nodes in the network
p_{vu}	the RF power needed for the link from node v to node u, $p_{\min} \leq p_{vu} \leq p_{\max}$
r_{vu}	the distance between node v to node u
r_{vu}^i	the transmission range required at v to reach all its child nodes in T_s^i and a node u outside T_s^i
w_v^i	the node weight of v at the snapshot T_s^i
s	the source node of multicast group M
w_{vu}^i	the arc weight of (v, u) at the snapshot T_s^i
α	the propagation loss exponent
$\delta_d(T_s)$	the maximum arc weight of T_s in a network with directional antennas
$\delta_o(T_s)$	the maximum arc weight of T_s in a network with omni-directional antennas
ε_v	the residual energy of node v
φ_v	the antenna beamwidth applied by node v
φ_{vu}^i	the minimum beamwidth required at v to reach all its child nodes in T_s^i and a node u outside T_s^i
μ_Z	the upper bound of ρ_Z
ρ_Z	the approximation-ratio of the algorithm Z
τ_{vu}	the maximal lifetime of an arc $(v, u) \in A$ at a given energy supply
$\psi(C_X)$	the minimum weight of arcs in C_X
Ω_M	the family of all rooted multicast trees including nodes in M
$(\cdot)^*$	an optimal solution

the duration of the network operation time until the disconnection of the multicast tree due to the battery depletion. This optimization problem in networks with directional antennas has been studied in [13-19] and has been proven to be NP-hard [19]. The exact solution for such a difficult problem is presented in [18] based a MILP (mixed integer linear programming) formulation.

The RB-MIP/D-MIP (Reduced-Beam/Directional Multicast Incremental Power) algorithms proposed in [16, 17] extend the minimum-energy metric by incorporating residual battery energy based on the observation that long-lived multicast trees should consume less energy and should avoid nodes with small residual energy as well. Specifically, the cost of a link between nodes v and u is defined as $c_{vu} = p_{vu} \cdot (\varepsilon_v(0)/\varepsilon_v(t))^{\beta}$, where p_{vu} is the RF transmission power needed for the link from node v to node u, $\varepsilon_v(t)$ is the residual energy at node v at time t, and β is a parameter that reflects the importance we assign to the impact of residual energy. The work in [13] uses a different approach from [16, 17] with the lifetime as an explicit optimization objective. Similar to the greedy approach applied by Prim' algorithm, the proposed algorithm D-DPMT (Dynamic weight Directed Prime Multicast Tree) [13] increments a multicast tree by including one node at a time with proper antenna beam reconfiguration of its uplink node such that the included arc has the maximum lifetime at that iteration.

All these solutions are centralized, meaning that at least one node needs global network information in order to construct an energy efficient multicast tree. This may result in extreme and unacceptable requirements on memory and computation capacities for a resource-constrained wireless multihop network. Two distributed maximum-lifetime algorithms DMMT-OA (Distributed Min-Max Tree algorithm for Omnidirectional Antennas) and DMMT-DA (Distributed Min-Max Tree algorithm for Directional Antennas) have been proposed in [14] for directional communications. Their theoretical performance have been studied in [15]. Simulation results have also shown that these two distributed multicast algorithms for directional communications outperform other centralized multicast algorithms, *e.g.*, [13, 16, 17], significantly.

The advance of this work inspires us to further investigate the distributed solutions for this optimization problem. A careful observation on the DMMT-DA algorithm leads to a new distributed algorithm with improved performance. The distinct characteristics of our proposal and the contributions of the paper are summarized as follows.

1) The centralized algorithm [13] is not practical for resource-constrained wireless networks. Our proposed DMMT-DA-NC algorithm can be implemented in a distributed manner, which is particularly beneficial to such networks.
2) Different from the traditional link-centric based distributed algorithm [14], the proposed DMMT-DA-NC algorithm can avoid some solutions that may be deviated far from optimal.
3) The approximation-ration upper bound for the DMMT-DA-NC algorithm has been derived in a close-form by the first time in this paper. The theoretical performance for other algorithms given in [15] can be also achieved as special cases from this close-form.

III. System Model and Problem Statement

The high complexity of the longest-lived multicast problem in WANETs with directional antennas [19] may inhibit from providing optimal solutions for many network examples. As a result, most studies have focused on the *logical* problem of establishing energy-efficient structures for broadcast/multicast communications. Their approach is to assess the complexities one at a time and the study of underlying technologies, *e.g.*, TDMA (Time Division Multiple Access) and CDMA (Code Division Multiple Access) based systems, however, is not pursued at the same time. In other words, the multicast group is assumed to be assigned sufficient bandwidth and transceiver resources throughout the duration of the session. This paper shall further investigate this fundamental optimization problem along the same approach, under which a wireless ad hoc network can be modeled as a simple directed graph G with a finite node set N ($|N| = n$) and an arc set A corresponding to the unidirectional wireless communication links. Each node is equipped with a directional antenna, which concentrates RF transmission power to where it is needed.

We assume a widely used directional antenna propagation model [16, 17], in which the antenna at each node v can switch its orientation to any desired direction with transmission power uniformly distributed across its adjustable beamwidth φ_v between $\varphi_{\min}$ and 2π. The transmission power p_{vu} to support a link (v, u) separated by a distance r_{vu} ($r_{vu} > 1$) is therefore proportional to r_{vu}^{α} and φ_v with unit signal detection threshold, where the propagation loss exponent α typically takes on a value between 2 and 4. We further assume that any node $v \in N$ can choose its transmission power, strictly within some minimum and maximum finite levels $p_{\min}$ and $p_{\max}$, respectively, which are positive constant numbers. The transmission power p_{vu} thus can be expressed as follows.

$$p_{vu} = p(r_{vu}, \varphi_v) \tag{1}$$

$$p(r, \varphi) = \max\left\{p_{\min}, r^{\alpha} \cdot \frac{\varphi}{2\pi}\right\} \leq p_{\max} \tag{2}$$

We consider a source-initiated multicast with multicast members $M = s \cup D$ ($|M| = m$), where s is the source node and D is a set of destination nodes. All the nodes involved in the multicast form a multicast tree rooted at the node s, *i.e.*, a rooted tree T_s, with a tree node set $N(T_s)$ and a tree arc set $A(T_s)$. We define a rooted tree as a directed acyclic graph with a source node with no incoming arcs, and each other node v has exactly one incoming arc. A node with no out-going arcs is called a leaf node, and all other nodes are internal nodes. Let $\varepsilon = \{\varepsilon_v > 0 \,|v \in N\}$ be the energy supply associated with each node in G. The residual lifetime τ_{vu} of an arc $(v, u) \in A(T_s)$ is therefore ε_v / p_{vu}.

Let Ω_M be the family of all rooted multicast trees including nodes in M. The longest-lived multicast problem can thus be expressed as

$$\max_{T_s \in \Omega_M} \min_{(v,u) \in A(T_s)} \tau_{vu} = \left(\min_{T_s \in \Omega_M} \max_{(v,u) \in A(T_s)} 1/\tau_{vu}\right)^{-1}. \tag{3}$$

Note that if we assign the tree arc weight function w_{vu} as the reciprocal of the lifetime of the arc (v, u), *i.e.*,

$$w_{vu} = \frac{1}{\tau_{vu}} = \frac{p(r_{vu}, \varphi_v)}{\varepsilon_v}, \tag{4}$$

our optimization problem is equivalent to the *min-max tree problem*, which is to determine a directed tree T_s including all the multicast members (*i.e.*, $M \subseteq N(T_s)$) such that the maximum arc weight is minimized. The corresponding optimal solution is just the reciprocal of the lifetime of the longest-lived multicast tree. Given a multicast tree T_s, we use $\delta_o(T_s)$ and $\delta_d(T_s)$ to denote the maximum arc weight of the same tree in a network instance $G(N, A)$ equipped with omni-directional antennas and directional antennas, respectively, *i.e.*,

$$\delta_o(T_s) \equiv \max_{(v,u) \in A(T_s)} \{p(r_{vu}, 2\pi)/\varepsilon_v\} \tag{5}$$

$$\delta_d(T_s) \equiv \max_{(v,u) \in A(T_s)} \{p(r_{vu}, \varphi_v)/\varepsilon_v\} \tag{6}$$

The arc with the above weights (5) and (6) is called the *omni-directional* and *directional* bottleneck arc, respectively.

Although there are technically an infinite number of beam configurations for a directional antenna by setting various values for beamwidth and orientation (*i.e.*, the direction of antenna boresight), it is sufficient to determine the solution of our optimization problem under the configuration such that the antenna beam at any node v achieves minimum coverage for a subset S, $S \subseteq \{u|(v, u) \in A\}$, of all its neighboring nodes. The minimum coverage means that the beamwidth φ_v in (6) at node v should be set to the smallest possible angle in the range between $\varphi_{\min}$ and 2π to provide the beam-coverage for all nodes in S. In this way, the configuration of antenna beamwidth and orientation is equivalent to choose a subset of neighboring nodes to be covered. We consider $S = \{u_1, u_2, \cdots, u_k\}$. Let α_{xy} ($0 \leq \alpha_{xy} < 2\pi$) be the angle measured counter-clockwise from the horizontal axis to the vector $\langle x, y\rangle$, from node x pointing to node y. Without loss of generality, we assume that $\alpha_1 < \alpha_2 < \cdots < \alpha_k$ are all such angles of vectors $\langle v, u_i\rangle$ ($i = 1, \cdots, k$) in an increasing order, in which each child node u_i is covered by the antenna beam at v. The smallest beamwidth φ_v at v can be obtained in a straightforward manner as follows.

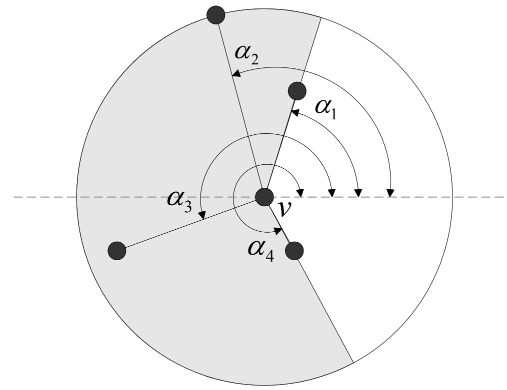


Fig. 1: An example to calculate the smallest beamwidth.

$$\max\left\{\varphi_{\min}, 2\pi - \max_{1 \leq i \leq k}\{\alpha_{i+1} - \alpha_i, 2\pi + \alpha_1 - \alpha_k\}\right\} \tag{7}$$

In the example of Fig. 1, node v has four child nodes ($k = 4$). Its smallest beamwidth thus can be obtained using (7) and denoted as the shaded area. The corresponding antenna orientation is just the central direction of the beam.

Note that although we can assign each parameter $\varphi_{\min}$ or $p_{\min}$ to be an arbitrarily small number in our model, such settings for both parameters are not allowed at the same time for the longest-lived broadcast problem. This is because otherwise each node would apply $\varphi_{\min}$ to reach just one downlink node and the final spanning tree becomes a Hamiltonian path, resulting in the tree lifetime arbitrarily large. Similar reasoning applies to the multicast case. In order to avoid such an infinite-lifetime problem, we constrain the following parameter to be a finite constant.

$$\mu_0 \equiv \min\left\{\frac{2\pi}{\varphi_{\min}}, \frac{p_{\max}}{p_{\min}}\right\} \tag{8}$$

Finally, we introduce several notations that will be used in the rest of the paper. We use $\Lambda_v^+(T_s)$ and $\lambda_v^+(T_s)$ to denote the child node set and the out-degree (*i.e.*, the number of child nodes) of node v in the tree T_s, respectively. Given a multicast

```
1:  Initialize i = 0, N(T_s^i) = {s}, and A(T_s^i) = ∅;
2:  repeat
3:      δ = min_{(v,u)∈C_{N(T_s^i)}} {w_{vu}^i, ∞};
4:      while ∃(v,u) ∈ C_{N(T_s^i)}, w_{vu}^i ≤ δ do
5:          i = i + 1;
6:          N(T_s^i) = N(T_s^{i-1}) ∪ {u};
7:          A(T_s^i) = A(T_s^{i-1}) ∪ {(v,u)};
8:      end while
9:  until M ⊆ N(T_s)
10: return the final multicast tree T_s by pruning from T_s^i.
```

Fig. 2: Framework of the distributed algorithms.

request (s, D), we use C_X to denote all arcs crossing a node partition $(X, N - X)$ such that the first node set X must include at least the source node s and the second node set $N - X$ must include at least one destination node in D, *i.e.*,

$$C_X \equiv \{(v,u) | v \in X \wedge u \in N - X \wedge s \in X \wedge D \not\subset X\} \quad (9)$$

Any C_X is referred as a cut of G with regards to a multicast request (s, D).

IV. Distributed Algorithms

In this section, we first briefly overview existing distributed algorithms DMMT-OA/DMMT-DA [14] and then present a new one motivated by an important observation that can improve the performance of DMMT-DA. All distributed algorithms shall be described in a formal manner, which helps us understand the theoretical performance analysis in latter sections.

A. Overview of Distributed Algorithms

Whenever there is a multicast session request (s, D) in the network but no route information is known, both DMMT-OA and DMMT-DA algorithms iterate a *Search-and-Grow* procedure to incrementally construct a multicast tree until the tree contains all the nodes in M. The Search phase of the procedure finds the minimum weight δ of arcs crossing the tree node set and the non-tree node set. In the subsequent Grow phase, the intermediate tree grows by including as many non-tree links as possible into the multicast tree such that the weight of each included link is equal or less than δ until no more such links can be found. The Search-and-Grow procedure starts from an initial tree, which contains the source node s only, and each iteration step makes the tree grow with one or more nodes to be included. After all destination nodes are included, the final multicast tree is achieved by pruning all unnecessary links, *i.e.*, the branches including non-member nodes only.

We use T_s^i to denote an intermediate tree constructed by a distributed algorithm after the $(i+1)$-th $(i \geq 0)$ node, or the i-th arc equivalently, is added into the tree. Similarly, we use w_{vu}^i to denote the weight of (v, u) at that moment. The framework of these distributed algorithms formulated in pseudo code is given in Fig. 2.

Note that the link weight function w_{vu}^i is calculated using variant methods in different algorithms. The DMMT-OA algorithm disregards the beamwidth at each node in the tree construction process, *i.e.*, assuming using omnidirectional antennas. The calculation of w_{vu}^i is defined below.

$$w_{vu}^i \equiv p(r_{vu}, 2\pi)/\varepsilon_v \quad (10)$$

After the final tree is constructed, each internal node set its antennas beamwidth to minimum value using (7). Unlike the *omni-directional* arc weights (10) used in the DMMT-OA algorithm, which remain unchanged throughout the execution of the algorithm, the DMMT-DA algorithm dynamically updates the *directional* weights w_{vu}^i at each step to reflect the changes of beamwidth as follows.

$$w_{vu}^i \equiv p(r_{vu}, \varphi_{vu}^i)/\varepsilon_v \quad (11)$$

$$\varphi_{vu}^i \equiv \min\left\{\varphi_v \,|\varphi_v \text{ covers nodes in } \{u\} \cup \Lambda_v^+(T_s^i)\right\} \quad (12)$$

As for the distributed implementation of Search-and-Grow based algorithms, a straightforward approach is the source routing, in which the global network state information is collected and sent back to the source node. The global network state information is, in general, at order of the total number of network links. In a wireless network, only relative position information can be obtained by the local measurements of range and bearing between each node and its neighboring nodes. Such link-related information sent back to the source will generate communication overhead up to $O(n^2)$. The $O(n)$ communication overhead could be possible but under a strong assumption that absolute nodal location information is available for each node. Furthermore, the source routing requires $O(n^2)$ memory space for maintaining the global network state information at one node (*i.e.*, the source node), which prohibits such approach from being used in large-scale resource-constrained wireless networks. For this reason, we have applied a different approach using the traditional hop-by-hop based Request-Reply message propagation [14]. In the Search Phase, each node v in the intermediate tree T_s^i will first calculate a local minimum weight w_v^i and send a Reply message with an aggregated value along the reverse tree-path back to the source. At node v, for example, such value is $\min_u\{w_u^i\}$ for any node u in the subtree of T_s^i rooted at v. If node v is a new multicast member that just joined the tree, it should also include its Id in its Reply message. Eventually, the source node will obtain from the Reply messages the value of δ, given in line (3) of the pseudo code, as well as a set of updated multicast members. If such set includes all the nodes in D, the Search-and-Grow procedure terminates with a constructed multicast tree. Otherwise, the source will check the value of δ. If $\delta < \infty$, the source will initiate a new Grow Phase by propagating the Request messages with the parameter δ over the multicast tree to include as many new arcs as possible that satisfy the condition in line (4) of the pseudo code. When the Grow operation completes at any node, *i.e.*, it cannot find any new qualified arc to be included, it then goes to the Search operation again as described earlier. On the other hand, if $\delta = \infty$, the source concludes that the multicast members are in disconnected network components and the Search-and-Grow procedure terminates with a failure.

As noted above, the major difference of Search-and-Grow based algorithm from Prim's algorithm is that it includes as many new nodes as possible at each iterative tree construction step. Some examples with multiple new node inclusions at one round of Search-and-Grow are given in [14]. In this way, the total number of Search-and-Grow rounds will be reduced significantly, resulting in a much low communication overhead. The message complexity of the Search-and-Grow based distributed algorithms has been comprehensively studied in [20], in which the simulation results show that an expected linear message complexity can be achieved under different network sizes and multicast group sizes.

B. A New Distributed Algorithm

As mentioned earlier, the DMMT-DA algorithm is one of the best solutions and especially beneficial to WANETs because of its distributed scheme. The earlier simulation studies have also shown that it outperforms DMMT-OA and other centralized algorithms, *e.g.*, in [13, 16, 17]. Its superiority inspires us for a further investigation on distributed solutions. Our case study on the DMMT-DA algorithm leads to the design of a new distributed algorithm that may provide even better performance.

A 5-node network instance is given in Fig. 3 with a source node s and a set of destination nodes a, b, c and d. The energy distribution is set as $\varepsilon_s = \varepsilon_c = \varepsilon_d > 0$ and $\varepsilon_a = \varepsilon_b = 0$. Note that the Euclidean distance between each pair of nodes is exactly indicated in Fig. 3. We consider a certain round of the *Search-and-Grow* procedure from an intermediate solution T_s^i ($i = 2$) obtained by the DMMT-DA algorithm with tree arcs (s, a) and (s, b) only. In the Search phase, variable δ is equal to $\min\{w_{sc}^i, w_{sd}^i\} = w_{sc}^i$ based on (11) and (12). This is because $w_{sc}^i < w_{sd}^i$ or equivalently $p(r_{sc}, \angle asc) < p(r_{sd}, \angle asd)$, in which the symbol $\angle xyz$ denotes the angle between the two rays of yx and yz. In the subsequent Grow phase, arcs (s, c) and (c, d) will be included into the tree. Finally, the multicast tree T_1 is achieved by DMMT-DA as shown in Fig. 3a with $\delta_d(T_1) = p(r_{sb}, \angle asc)/\varepsilon_s$. Now we consider an alternative arc (s, d) to be included into the tree in the same round of iteration and the final tree should be T_2 as shown in Fig. 3b with $\delta_d(T_2) = p(r_{sb}, \angle asd)/\varepsilon_s$. It is obvious that T_2 is a better solution, *i.e.*, $\delta_d(T_1) > \delta_d(T_2)$, because $\angle asc > \angle asd$. In other words, it may exist significant gap between the solution found by DMMT-DA based on the arc-weight and the optimum.

The above example motivates us to apply a node-centric approach, *i.e.*, to use a node-weight instead of an arc-weight defined in (10) or (11) as the criteria, to increment a multicast tree such that the performance of DMMT-DA could be improved. We propose a new algorithm, DMMT-DA-NC (DMMT-DA using the Node-Centric approach), for the min-max tree problem under the same framework except that line (4) is updated by line (4′) below.

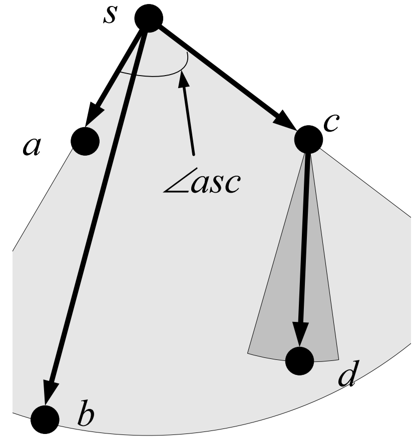
4′: **while** $\exists(v, u) \in C_{N(T_s^i)}, w_v^i = w_{vu}^i \leq \delta$ **do**

The variables w_v^i and w_{vu}^i in the DMMT-DA-NC algorithm are defined as follows.

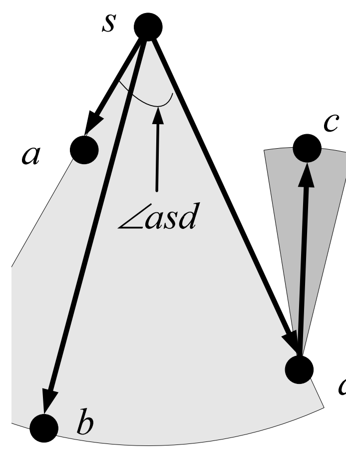
$$w_v^i \equiv \min\left\{w_{vu}^i \,\middle|\, u \in N - N(T_s^i)\right\} \quad (13)$$

$$w_{vu}^i \equiv p(r_{vu}^i, \varphi_{vu}^i)/\varepsilon_v \quad (14)$$

$$r_{vu}^i \equiv \max\left\{r_{vx} \,\middle|\, x \in \{u\} \cup \Lambda_v^+(T_s^i)\right\} \quad (15)$$



(a) $T_1 : \{(s, a), (s, b), (s, c), (c, d)\}$



(b) $T_2 : \{(s, a), (s, b), (s, d), (d, c)\}$

Fig. 3: An example to show that the result obtained by DMMT-DA can be improved using a node-centric approach.

As implied by the name of the algorithm, each node v maintains a node weight w_v^i ($0 \leq i \leq n-1$) as well at each step a tree is incremented. Note that the variable r_v^i denotes the longest Euclidean distance between node v and node x which is either the node u outside the tree T_s^i or any child node of node v already in the tree. In this greedy way, the tree incremental operation would lead to the lifetime of the resulting intermediate tree to be maximized over all possible choices of a non-tree node to be included into the tree.

Now we reconsider the example in Fig. 3 using the DMMT-DA-NC algorithm at the snapshot T_s^i with tree arcs (s, a) and (s, b) only. Note that at this moment, $r_{sc}^i = \max\{r_{sc}, r_{sa}, r_{sb}\} = r_{sb}$ and $r_{sd}^i = \max\{r_{sd}, r_{sa}, r_{sb}\} = r_{sb}$ are obtained from (15) and the corresponding arc weights are calculated using the new definition given in (14) as $w_{sc}^i = p(r_{sb}, \angle asc)/\varepsilon_s$ and $w_{sd}^i = p(r_{sb}, \angle asd)/\varepsilon_s$. Based on line (4′) for new arc inclusion criteria, we determine arc (s, d) to be chosen since $w_s^i = \min\left\{w_{sc}^i, w_{sd}^i\right\} = w_{sd}^i$ as shown in Fig. 3b. Comparing to the result in in Fig. 3a, we notice a substantial improvement.

Finally, we discuss the complexity of the distributed DMMT-DA-NC algorithm to conclude this section. Each node v requires $O(n)$ space for maintaining a table for all the nodes within its transmission range. Corresponding to each neighboring node u, an entry of the table records the geographical information of u, routing information (*e.g.*, node u is the parent or a child of v), status (*e.g.*, node u is in the tree or not), and the weight of arc (v, u) which should be updated dynamically in the tree construction. The initialization at each node takes time $O(n)$ for setting its neighboring table. The major computation in the tree construction is that when a tree node includes a new arc into the tree, it has to update the arc weight for each of its neighboring nodes outside the tree. For example, we consider node v at the snapshot T_s^i. It maintains the farthest child in $\Lambda_v^+(T_s^i)$ and the boundaries of the minimum beamwidth that can cover all its child nodes in $\Lambda_v^+(T_s^i)$. Then, for each neighboring node u outside the tree, the weight w_{vu}^i can be updated using (14), (15), and (12) in $O(1)$. Therefore, the overall time complexity involving one arc to be included is $O(n)$. By applying the hop-by-hop approach discussed in Section 4.1, the DMMT-DA-NC algorithm can achieve low communication overhead as well. Considering the low space/time/message complexity of the propoased distributed algorithm, we shall only focus on the analysis of its theoretical performance , in terms of algorithm approximation ratio, in the rest of the paper.

V. Theoretical Performance Analysis

We study the theoretical performance of the distributed algorithms in terms of approximation ratio. An algorithm for a problem has an approximation ratio of $\rho(n)$ if, for any input of size n, the expected cost c of the solution produced by the algorithm is within a factor of $\rho(n)$ of the cost c^* of an optimal solution: $\max\{c/c^*, c^*/c\} \leq \rho(n)$. The approximation ratio is the crucial metric that describes how close a heuristic algorithm performs to the optimal solution. In this section, we first provide some fundamental results that shall be used to derive the upper bound of the approximation ratio for each algorithm.

A. Fundamentals

Let δ_o^* and δ_d^* be the optimal solutions for the min-max tree problem under *omni-directional* and *directional* scenarios, respectively, *i.e.*,

$$\delta_o^* = \min_{T_s \in \Omega_M} \delta_o(T_s), \tag{16}$$

$$\delta_d^* = \min_{T_s \in \Omega_M} \delta_d(T_s). \tag{17}$$

Given a multicast tree T_s obtained by a distributed algorithm Z, its approximation ratio ρ_Z can be expressed as

$$\rho_Z = \delta_d(T_s)/\delta_d^*. \tag{18}$$

In order to study the lower bounds on both omni-directional and directional optimal solutions, we introduce *cut weight* $\psi(C_X)$, defined as the minimum omni-directional weight of arcs in C_X, *i.e.*,

$$\psi(C_X) \equiv \min_{(v,u)\in C_X} \{p(r_{vu}, 2\pi)/\varepsilon_v\}. \tag{19}$$

We further define an auxiliary function $K_{vu}(\varphi_1, \varphi_2)$ as follows.

$$K_{vu}(\varphi_1, \varphi_2) \equiv \frac{p(r_{vu}, \varphi_1)}{p(r_{vu}, \varphi_2)} \tag{20}$$

It is a straightforward exercise to show that $K_{vu}(\varphi_1, \varphi_2)$ is a non-decreasing function of φ_1 and a non-increasing function of φ_2. In particular, for any $(v, u) \in A$, it satisfies

$$\begin{cases} 1/\mu_0 \leq K_{vu}(\varphi_1, \varphi_2) \leq 1 & \varphi_1 \leq \varphi_2 \\ 1 \leq K_{vu}(\varphi_1, \varphi_2) \leq \mu_0 & \varphi_1 \geq \varphi_2 \end{cases} \tag{21}$$

Finally, we summarize the lower bounds of δ_o^* and δ_d^* in the following lemmas, whose strict proofs can be found in [15].

Lemma 1. If there exists a multicast tree in $G(N, A)$, then for any cut C_X

$$\delta_o^* \geq \psi(C_X). \tag{22}$$

Lemma 2. If there exists a multicast tree in $G(N, A)$, then the optimal solutions δ_o^* and δ_d^* satisfy

$$\delta_d^* \geq \delta_o^*/\mu_0. \tag{23}$$

B. Performance of DMMT-DA-NC

Now, we turn our attention to the most interesting and difficult task on deriving and evaluating the approximation ratio of the DMMT-DA-NC algorithm. Suppose that T_s is the final multicast tree obtained from the DMMT-DA-NC algorithm and the directional bottleneck arc (v, u) of T_s applies its beamwidth φ_v using (7). We further assume that arc (v', u') is the first one added into the tree in the same round of the Search-and-Grow procedure as arc (v, u) is included. Let T_s^j and T_s^i $(i > j)$ be the intermediate trees just before the arcs (v', u') and (v, u) to be included into the multicast tree, respectively. In particular, we shall give attention to the node partition $(X, N - X)$, $X \equiv N(T_s^j)$, and the arc $(x, y) \in C_X$ that satisfies $p(r_{xy}, 2\pi)/\varepsilon_x = \psi(C_X)$. Several equations regarding the arcs mentioned above have been observed as follows.

Observation 1. Because arc (v, u) is chosen to be included into the multicast tree, the condition in line $(4')$ of the pseudo code for the DMMT-DA-NC algorithm must be satisfied, *i.e.*,

$$w_v^i = w_{vu}^i. \tag{24}$$

Observation 2. Let δ be the minimum weight found just before arc (v', u') is added into the tree. We then have $\delta = \min\{w_{ab}^j \,\big|(a, b) \in C_X, X = N(T_s^j)\} \leq w_{v'u'}^j$. On the other hand, the condition in line $(4')$ on arc (v', u') implies $w_{v'}^j = w_{v'u'}^j \leq \delta$. Therefore, we can conclude that $\delta = w_{v'u'}^j$. Similarly, because arc (v, u) is also included in the same round of the Search-and-Grow procedure as arc (v', u'), the condition in line $(4')$ on arc (v, u) must be satisfied as well, *i.e.*, $w_v^i = w_{vu}^i \leq \delta$. By combining all inequalities above, we finally achieve

$$w_{v'}^j \geq w_v^i. \tag{25}$$

Observation 3. We reconsider the moment just before arc (v', u') is added into the tree. The node weight on v' and

x at that time satisfies $w^j_{v'} = \delta$ (discussed in Observation 2) and $\delta = \min\{w^j_{ab} \,|(a,b) \in C_X, X = N(T^j_s)\} \le \min\{w^j_{xb} \,|b : (x,b) \in C_X, X = N(T^j_s)\} = w^j_x$, respectively. We therefore obtain

$$w^j_{v'} \le w^j_x. \tag{26}$$

In order to find the approximation ratio upper-bound of the DMMT-DA-NC algorithm in a close-form, we use the above findings to derive $\delta_d(T_s)$ as follows.

$$\begin{aligned}\delta_d(T_s) &= p(r_{vu}, \varphi_v)/\varepsilon_v \\ &= K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot p(r_{vu}, \varphi^i_{vu})/\varepsilon_v \\ &\le K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot p(r^i_{vu}, \varphi^i_{vu})/\varepsilon_v \\ &= K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot w^i_{vu} \\ &= K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot w^i_{v} \\ &\le K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot w^j_{v'}\end{aligned}$$

Notice that $p(r_{vu}, \varphi^i_{vu}) \le p(r^i_{vu}, \varphi^i_{vu})$ is true in the above derivation because $p(r, \varphi)$ is a non-decreasing function of r and $r_{vu} \le r^i_{vu}$. The last two steps $w^i_{vu} = w^i_v$ and $w^i_v \le w^j_{v'}$ are also guaranteed by Observations 1 and 2, respectively. By applying Observation 3, we can further achieve

$$\begin{aligned}\delta_d(T_s) &\le K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot w^j_{v'} \\ &\le K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot w^j_{x} \\ &\le K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot w^j_{xy}.\end{aligned}$$

Now, two cases should be considered and the corresponding derivations under each case are conducted in the following.

Case (1) $r^j_{xy} = r_{xy}$: The arc-weight w^j_{xy} can be simply rewritten as

$$w^j_{xy} = p(r^j_{xy}, \varphi^j_{xy})/\varepsilon_x = p(r_{xy}, \varphi^j_{xy})/\varepsilon_x$$

Case (2) $r^j_{xy} = r_{xz_1}, z_1 \in \Lambda^+_x(T^j_s)$: As shown in Fig. 4, arc (x, z_1) is already in the tree T^j_s and we assume it was chosen to be included at the snapshot $T^{k_1}_s$ ($k_1 < j$). Because it is arc (x, z_1), instead of (x, y), that was chosen at that time, we can conclude that $w^{k_1}_{xz_1} = w^{k_1}_x \le w^{k_1}_{xy}$, or equivalently

$$p(r^{k_1}_{xz_1}, \varphi^{k_1}_{xz_1}) \le p(r^{k_1}_{xy}, \varphi^{k_1}_{xy}). \tag{27}$$

Furthermore, conditions $r^j_{xy} = r_{xz_1}$ and $k_1 < j$ imply

$$r^{k_1}_{xz_1} = r_{xz_1}. \tag{28}$$

Now the arc-weight w^j_{xy} under Case (2) can be rewritten as follows by combining (27) and (28).

$$\begin{aligned}w^j_{xy} &= p(r_{xz_1}, \varphi^j_{xy})/\varepsilon_x \\ &= K_{xz_1}(\varphi^j_{xy}, \varphi^{k_1}_{xz_1}) \cdot p(r_{xz_1}, \varphi^{k_1}_{xz_1})/\varepsilon_x \\ &= K_{xz_1}(\varphi^j_{xy}, \varphi^{k_1}_{xz_1}) \cdot p(r^{k_1}_{xz_1}, \varphi^{k_1}_{xz_1})/\varepsilon_x \\ &\le K_{xz_1}(\varphi^j_{xy}, \varphi^{k_1}_{xz_1}) \cdot p(r^{k_1}_{xy}, \varphi^{k_1}_{xy})/\varepsilon_x \\ &= K_{xz_1}(\varphi^j_{xy}, \varphi^{k_1}_{xz_1}) \cdot w^{k_1}_{xy}\end{aligned}$$

The obtained result $w^j_{xy} \le K_{xz_1}(\varphi^j_{xy}, \varphi^{k_1}_{xz_1}) \cdot w^{k_1}_{xy}$ in a recursive form allows us to derive $w^{k_1}_{xy}$ under two cases again in a similar manner: 1) $r^{k_1}_{xy} = r_{xy}$ or 2) $r^{k_1}_{xy} = r_{xz_2}$ as shown in Fig. 4. Such a process shall repeat until Case (1) is met.

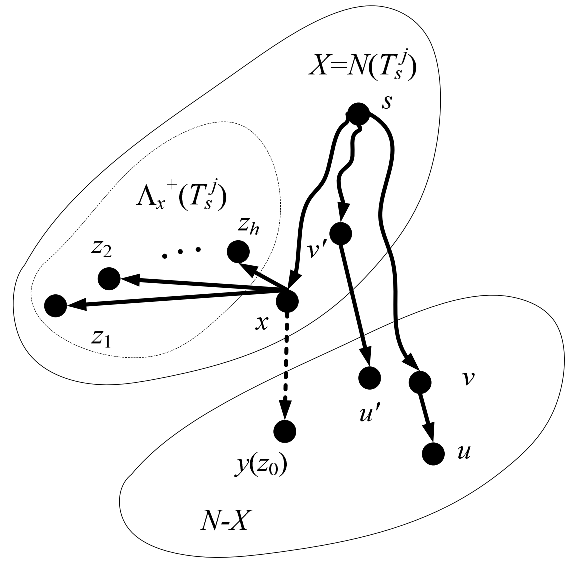


Fig. 4: Illustration for the approximation ratio derivation of the DMMT-DA-NC algorithm.

Without the loss of generality, we assume that Case (1) is met at the h-round ($0 \le h \le \lambda^+_x(T^j_s)$) of the above derivation iteration, *i.e.*,

$$\begin{cases} r^{k_\ell}_{xy} = r_{xz_{\ell+1}} & 0 \le \ell \le h-1 \\ r^{k_\ell}_{xy} = r_{xy} & \ell = h. \end{cases} \tag{29}$$

The boundaries of z_ℓ and k_ℓ are

$$z_0 = y, k_0 = j. \tag{30}$$

In the ℓ-round ($0 \le \ell \le h$) derivation, we assume that arc (x, z_ℓ) is chosen to be included into the intermediate tree $T^{k_\ell}_s$ ($k_h < k_{h-1} < \cdots < k_0$). The following equation will be eventually achieved.

$$w^j_{xy} \le H \cdot p(r^{k_h}_{xy}, \varphi^{k_h}_{xy})/\varepsilon_x = H \cdot p(r_{xy}, \varphi^{k_h}_{xy})/\varepsilon_x$$

The H-factor in the above expression is defined as follows.

$$H \equiv \begin{cases} 1 & h = 0 \\ \displaystyle\prod_{\ell=1}^{h} K_{xz_\ell}(\varphi^{k_{\ell-1}}_{xz_{\ell-1}}, \varphi^{k_\ell}_{xz_\ell}) & h \ge 1 \end{cases} \tag{31}$$

The arc-weight w^j_{xy} obtained so far by h iteration steps as shown in Fig. 4 and the results given in Lemmas 1 and 2 allow us to finalize the derivations of $\delta_d(T_s)$ as follows.

$$\begin{aligned}\delta_d(T_s) &\le K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot w^j_{xy} \\ &\le K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot H \cdot p(r_{xy}, \varphi^{k_h}_{xy})/\varepsilon_x \\ &= H \cdot K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot K_{xy}(\varphi^{k_h}_{xy}, 2\pi) \cdot p(r_{xy}, 2\pi)/\varepsilon_x \\ &= H \cdot K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot K_{xy}(\varphi^{k_h}_{xy}, 2\pi) \cdot \psi(C_X) \\ &\le H \cdot K_{vu}(\varphi_v, \varphi^i_{vu}) \cdot K_{xy}(\varphi^{k_h}_{xy}, 2\pi) \cdot \mu_0 \cdot \delta^*_d\end{aligned}$$

Finally, we summarize our findings for the DMMT-DA-NC algorithm in the following theorem.

Theorem 1. The DMMT-DA-NC algorithm is a constant-factor approximation algorithm with an approximation ratio $\rho_{\text{DMMT-DA-NC}}$ bounded by $\mu_{\text{DMMT-DA-NC}}$:

$$\mu_{\text{DMMT-DA-NC}} \equiv H \cdot K_{vu}(\varphi_v, \varphi_{vu}^i) \cdot K_{xy}(\varphi_{xy}^{k_h}, 2\pi) \cdot \mu_0. \quad (32)$$

C. Further Discussion

In this section, we extend the theoretical result for the DMMT-DA-NC algorithm to the other algorithms used the same Search-and-Grow framework.

Corollary 1. The DMMT-DA algorithm is a constant-factor approximation algorithm with an approximation ratio $\rho_{\text{DMMT-DA}}$ bounded by $\mu_{\text{DMMT-DA}}$:

$$\mu_{\text{DMMT-DA}} \equiv K_{vu}(\varphi_v, \varphi_{vu}^i) \cdot K_{vu}(\varphi_{xy}^j, 2\pi) \cdot \mu_0. \quad (33)$$

Proof: The DMMT-DA-NC algorithm will obtain the same solution as DMMT-DA if we force $r_{vu}^i = r_{vu}$ for any i and (v, u). Under this variation, Case (1) discussed in the last Section will be met when $h = 0$, resulting in $H = 1$. The corresponding approximation ratio upper-bound $\mu_{\text{DMMT-DA}}$ for the DMMT-DA algorithm can be obtained from the result of Theorem 1 by setting $H = 1$ and $h = 0$.

Corollary 2. The DMMT-OA algorithm is a constant-factor approximation algorithm with an approximation ratio $\rho_{\text{DMMT-OA}}$ bounded by $\mu_{\text{DMMT-OA}}$:

$$\mu_{\text{DMMT-OA}} \equiv K_{vu}(\varphi_v, 2\pi) \cdot \mu_0. \quad (34)$$

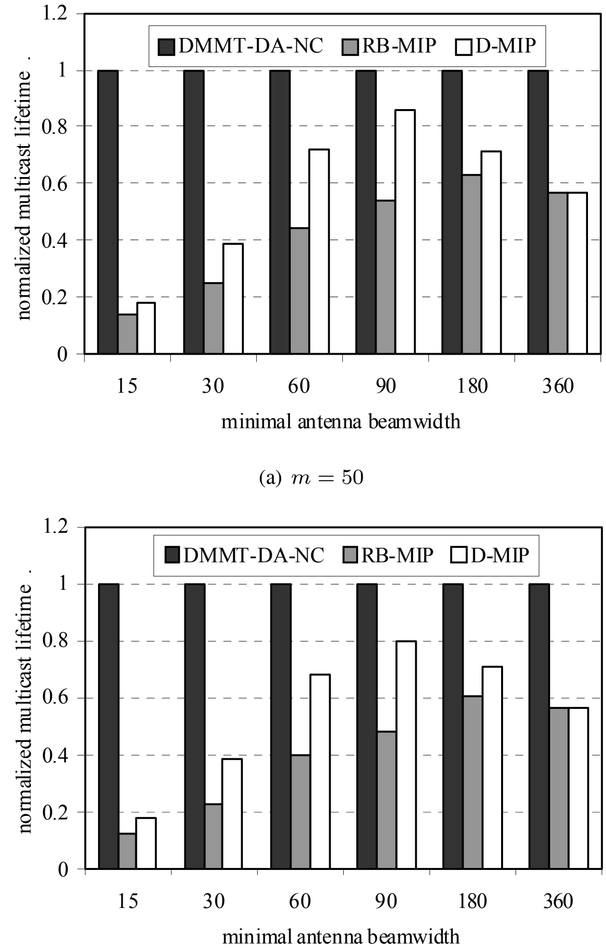
Proof: Similarly, the DMMT-DA algorithm will obtain the same solution as DMMT-OA if we force $\varphi_{vu}^i = 2\pi$ for any i and (v, u). The corresponding approximation ratio upper-bound $\mu_{\text{DMMT-OA}}$ for the DMMT-OA algorithm can be obtained from the result of Corollary 1 by setting $\varphi_{vu}^i = 2\pi$ and $\varphi_{xy}^j = 2\pi$.

The same theoretical results in [15] for the DMMT-OA and DMMT-DA algorithms are observed. These upper bounds given in the above Theorem and Corollaries for the distributed algorithms can be used to verify the optimal solutions as well. Once the sufficient optimality condition $\mu_Z = 1$ ($Z \in$ DMMT-OA, DMMT-DA and DMMT-DA-NC) is achieved; we can immediately conclude that the solution found by the distributed algorithm-Z is globally optimal.

VI. Experimental Performance Evaluation

We have performed a simulation study on a set of multicast lifetime optimization algorithms. In our simulation setup, a number of nodes are randomly distributed within a 10×10 square region for each network example. We specify a normally distributed energy supply with mean 500 and variance 200. Transmission power is restricted between $p_{\min}$ = 0.1 and $p_{\max}$ = 10, and a propagation-loss exponent of $\alpha = 2$ is assumed. The units of parameters are all consistent with each other. We randomly generated 100 different network examples, and we present here the average over those examples for all cases.

The first set of experiments is to evaluate our proposed DMMT-DA-NC algorithm by comparing it to the well-known heuristic algorithms RB-MIP and D-MIP [16, 17], for which we set the parameter $\beta = 2$ to emphasize the impact of residual energy in larger networks with 100 nodes. In order to facilitate the comparison of different algorithms over a wide range of network instances, we use the performance metric *normalized multicast lifetime* defined as the ratio of the tree lifetime obtained by a heuristic algorithm to the result obtained by DMMT-DA-NC.



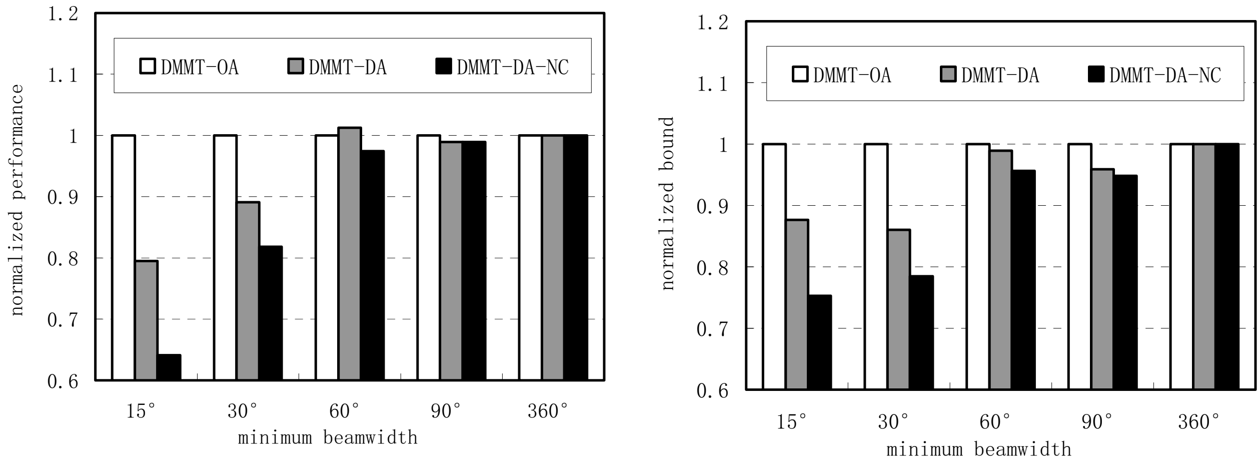
(a) $m = 50$

(b) $m = 100$

Fig. 5: Normalized multicast lifetime in 100-node networks.

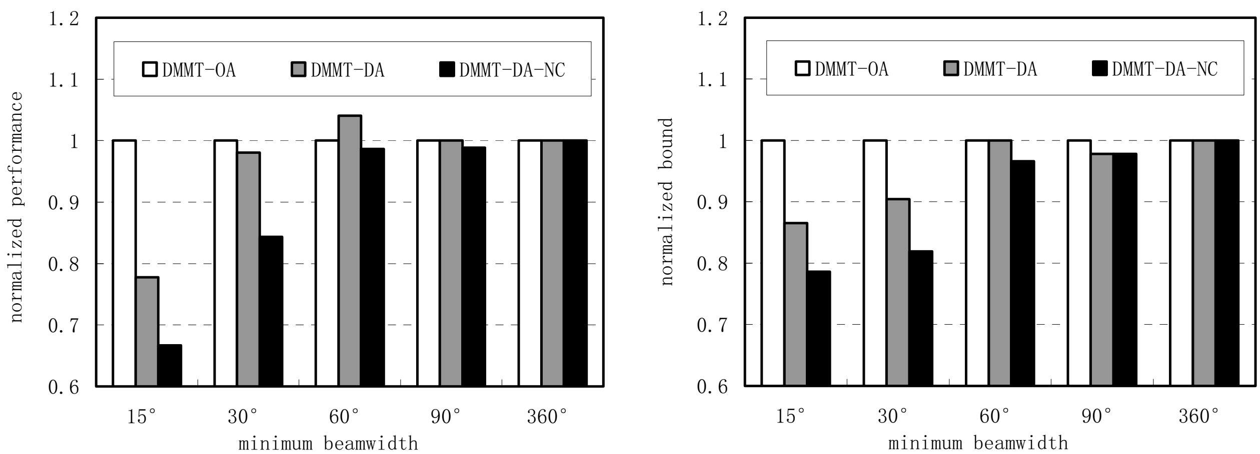
Figure 5 illustrates the mean values of normalized multicast lifetime as a function of the minimal antenna beamwidth under various multicast group sizes. The x-axis represents the minimum beamwidths $\varphi_{\min} = 15°, 30°, 60°, 90°$ and $360°$. Referring to the $m = 50$ multicast members in Fig. 5 (a), we observe that DMMT-DA-NC can provide over 50% and 40% longer lifetime, on average, compared to the RB-MIP and D-MIP algorithms, respectively. Similar results are observed in larger group sizes $m = 100$ in Fig. 5(b). In summary, the DMMT-DA-NC algorithm showis its superior performance that is guaranteed by its constant-factor approximation property.

In order to better understand the theoretical results we obtained in this paper, we then explore the relationship between the exact solution δ_Z obtained by algorithm-Z ($Z \in$



(a) width=3.35in (b) width=3.35in

Fig. 6: Normalized performance and normalized approximation-ratio upper bound as a function of the minimum beamwidths under multicast size 50.



(a) width=3.35in (b) width=3.35in

Fig. 7: Normalized performance and normalized approximation-ratio upper bound as a function of the minimum beamwidths under multicast size 100.

{DMMT-OA, DMMT-DA, DMMT-DA-NC}) and its corresponding approximation ratio upper bound μ_Z. We use the metric *normalized approximation ratio upper-bound* defined as $\mu_Z/\mu_{\text{DMMT-OA}}$ to study this relationship by comparing with the *normalized performance* $\delta_Z/\delta_{\text{DMMT-OA}}$ of this set of algorithms.

Fig. 6 depicts graphically the normalized performance and bound over different connected 100-node network topologies with the multicast size m = 50. The high correlation between values $\mu_Z/\mu_{\text{DMMT-OA}}$ and $\delta_Z/\delta_{\text{DMMT-OA}}$ as shown in Figs. 6a and 6b gives us insights to the real performance behavior of this group of distributed algorithms from a theoretical perspective. It means that the heuristic algorithm-Z with a tighter upper bound of the approximation ratio would have a better performance. This validates our analytical derivation of the approximation ratio upper bounds which allow to be used to evaluate the real performance of various heuristic algorithms. A similar observation can be made for the broadcasting scenarios m = 100 as shown in Fig. 7.

We also observe from both Figs. 6 and 7 that the new proposed distributed algorithm DMMT-DA-NC improves the other two algorithms significantly when the minimum beamwidth is small. In particular, such improvement is over 30% and 15% compared to DMMT-OA and DMMT-DA, respectively. On the other hand, once the minimum beamwidth increases (greater than 90°), all algorithms consistently converge to the same performance (the optimal solutions as proved in [14]), showing that the derived upper bounds μ_Z asymptotically approach to the exact approximation ratios δ_Z.

Finally, we would like to evaluate the absolute values of our derived upper bound for the DMMT-DA-NC algorithm. Its theoretical upper bound $\mu_{\text{DMMT-DA-NC}}$ is compared to the actual approximation ratio $\delta_{\text{DMMT-DA-NC}}$ calculated by (18), in which the optimal solutions are obtained using an optimization problem solver CPLEX [21] based on the MILP formulation given in [18]. Due to the computational expenses, only small

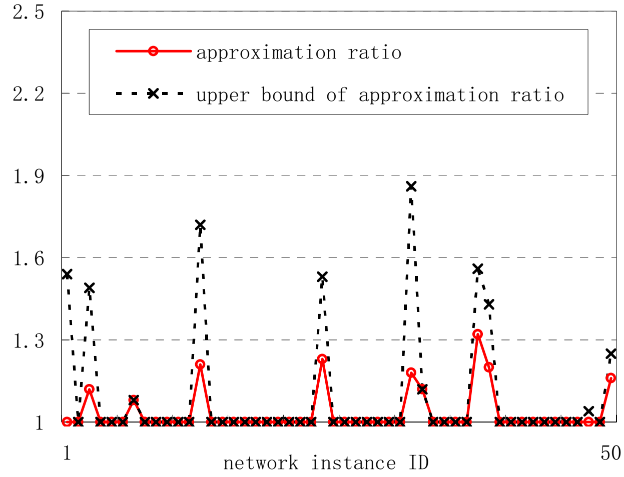


Fig. 8: The approximation-ratio and its theoretical upper bound for the DMMT-DA-NC algorithm in 20-node networks.

network examples (*e.g.*, n = 20) have been considered. The numerical results based on 50 randomly generated network examples with m = 20 and $\varphi_{min} = 60^{\circ}$ are graphically presented in Fig. 8. The solid line quantitatively indicates the performance deviation of DMMT-DA-NC from the optimal solution, which is about 3.24% in average.

We also observe that the validity of our derived bounds is always conserved. Over 80% of network instances, the derived upper bounds are exact the same as the actual approximation ratio, *e.g.*, when our distributed algorithm achieves an optimal solution as shown the overlapped points on x-axis. These results verify the sufficient optimality condition $\mu_{\text{DMMT-DA-NC}}$ = 1 that we have obtained from the theoretical analysis to estimate if a certain solution obtained by DMMT-DA-NC is exactly achieving the global optimum.

VII. Conclusion

In this paper, we have presented a new distributed long-lived multicast algorithm DMMT-DA-NC for wireless ad hoc networks using directional antennas. It distinguishes from and outperforms existing distributed algorithms DMMT-OA and DMMT-DA by exploring a node-centric approach. In order to better understand the superiority of our proposal, we study the theoretical performance, in terms of approximation ratio, of this group of distributed algorithms. In particular, we derive the upper bounds of their approximation ratio in a close form that can be used to evaluate how close a solution obtained by a distributed algorithm can achieve to the optimum. For example, the algorithm will perform exactly the same as the optimal one when its upper bound approaches to the minimal value 1. Furthermore, our experimental results show strong correlation between the upper bound and the average performance, *i.e.*, the smaller an upper bound is, the better performance the corresponding algorithm can achieve. More specifically, the proposed DMMT-DA-NC algorithm with the lowest upper bound shows the best performance in average, while the previous ones DMMT-OA and DMMT-DA are also constant-factor approximation algorithms.

References

[1] J. E. Wieselthier, G. D. Nguyen, *et al.*, "On the construction of energy-efficient broadcast and multicast trees in wireless networks," in *Proc. IEEE INFOCOM*, Tel-Aviv, 2000, pp. 585-594.

[2] J. Cartigny, D. Simplot, and I. Stojmenovic, "Localized minimum-energy broadcasting in ad-hoc networks," in *Proc. IEEE INFOCOM*, San Francisco, 2003, pp. 2210-2217.

[3] P. J. Wan, G. Calinescu, *et al.*, "Minimum-energy broadcast routing in static ad hoc wireless networks," in *Proc. IEEE INFOCOM*, Anchorage, 2001, pp. 1162-1171.

[4] P. J. Wan and C. W. Yi, "Minimum-power multicast routing in static ad hoc wireless networks," *IEEE/ACM Trans. Networking*, vol. 12, no. 3, June 2004, pp. 507-514.

[5] M. Cagalj, J.-P. Hubaux, and C. Enz, "Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues," in *Proc. ACM Mobicom*, Atlanta, 2002, pp. 172-182.

[6] L. Georgiadis, "Bottleneck multicast trees in linear time," *IEEE Commun. Lett.*, vol. 7, no. 11, pp. 564-566, Nov. 2003.

[7] I. Kang and R. Poovendran, "On the lifetime extension of energy-efficient multihop broadcast networks," *World Congress Computational Intelligence*, Honolulu, 2002.

[8] I. Kang and R. Poovendran, "Maximizing static network lifetime of wireless broadcast adhoc networks," in *Proc. IEEE ICC*, Anchorage, 2003, pp. 2256-2261.

[9] A. K. Das, R. J. Marks II, *et al.*, "MDLT: a polynomial time optimal algorithm for maximization of time-to-first-failure in energy-constrained broadcast wireless networks," in *Proc. IEEE Globecom*, San Francisco, 2003, pp. 362-366.

[10] M. X. Cheng, J. Sun, *et al.*, "Energy-efficient broadcast and multicast routing in ad hoc wireless networks," in *Proc. IEEE IPCCC*, Phoenix, 2003, pp. 87-94.

[11] B. Wang and S. K. S. Gupta, "On maximizing lifetime of multicast trees in wireless ad hoc networks," in *Proc. International Conf. Parallel Process.*, Kaohsiung, 2003, pp. 333-340.

[12] B. Floréen, P. Kaski, *et al.*, "Multicast time maximization in energy constrained wireless networks," in *Proc. Workshop Foundations Mobile Comput.*, New York, 2003, pp. 50-58.

[13] S. Guo and O. Yang, "Multicast lifetime maximization for energy-constrained wireless ad-hoc networks with directional antennas," in *Proc. IEEE Globecom*, Dallas, 2004, pp. 4120-4124.

[14] S. Guo, V. Leung, and O. Yang, "Distributed multicast algorithms for lifetime maximization in wireless ad hoc networks with omni-directional and directional antennas," *IEEE Globecom*, San Francisco, 2006.

[15] S. Guo, O. Yang, and V. Leung, "Approximation algorithms for longest-lived directional multicast communications in WANETs," in *Proc. ACM MobiHoc*, Montreal, 2007, pp. 190-198.

[16] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Energy-aware wireless networking with directional antennas: the case of session-based broadcasting and multicasting," *IEEE Trans. Mobile Comput.*, vol. 1, no. 3, 2002, pp. 176-191.

[17] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides, "Energy-limited wireless networking with directional antennas: the case of session-based multicasting," in *Proc. IEEE INFOCOM*, 2002, pp. 190-199.

[18] S. Guo and O. Yang, "Optimal tree construction for maximum lifetime multicasting in wireless ad-hoc networks with adaptive antennas," in *Proc. IEEE ICC*, Seoul, 2005, pp. 3370-3374.

[19] Y. Hou, Y. Shi, *et al.*, "Online lifetime-centric multicast routing for ad hoc networks with directional antennas," in *Proc. IEEE INFOCOM*, Miami, 2005, pp. 761-772.

[20] S. Guo, V. Leung, and O. Yang, "A scalable distributed multicast algorithm for lifetime maximization in large-scale resource-limited multihop wireless networks," in *Proc. ACM IWCMC*, Vancouver, 2006, pp. 419-424.

[21] CPLEX, http://www.cplex.com

Song Guo received the PhD degree in computer science from the University of Ottawa, Canada in 2005. Since then, he held a position with the Department of Electrical and Computer Engineering, the University of British Columbia, Canada on an NSERC (Natural Sciences and Engineering Research Council of Canada) Postdoctoral Fellowship. He is currently an Assistant Professor at School of Computer Science and Engineering, the University of Aizu, Japan. His research interests are mainly in the areas of protocol design and performance analysis for computer and telecommunication networks, presently focusing on modeling, analysis, cross-layer optimization, and performance evaluation of wireless ad hoc and sensor networks for reliable, energy-efficient, and cost effective communications.

Victor C. M. Leung received the B.A.Sc. (Hons.) and Ph.D. degrees, both in electrical engineering, from the University of British Columbia (UBC) in 1977 and 1981, respectively. He was the recipient of many academic awards, including the APEBC Gold Medal as the head of the 1977 graduating class in the Faculty of Applied Science, UBC, and the NSERC Postgraduate Scholarship. From 1981 to 1987, Dr. Leung was a Senior Member of Technical Staff and satellite systems specialist at MPR Teltech Ltd. In 1988, he was a Lecturer in Electronics at the Chinese University of Hong Kong. He returned to UBC as a faculty member in 1989, where he is a Professor and holder of the TELUS Mobility Research Chair in Advanced Telecommunications Engineering in the Department of Electrical and Computer Engineering. His research interests are in mobile systems and wireless networks. Dr. Leung is a Fellow of IEEE and a voting member of ACM. He is an associate editor of the IEEE TRANSACTIONS ON COMPUTERS, an editor of the *Journal of Communications and Networks*, and an editor of the *International Journal of Sensor Networks*.

Xiaohong Jiang received his B.S., M.S. and Ph.D degrees in 1989, 1992, and 1999 respectively, all from Xidian University, Xian, China. He is currently a full professor of Future University Hakodate, Japan. Before joining Future University, Dr.Jiang was an Associate professor, Tohoku University, from Feb.2005 to Mar.2010. He was an assistant professor in the Graduate School of Information Science, Japan Advanced Institute of Science and Technology (JAIST), from Oct.2001 to Jan.2005. Dr. Jiang was a JSPS (Japan Society for the Promotion of Science) postdoctoral research fellow at JAIST from Oct.1999-Oct. 2001. He was a research associate in the Department of Electronics and Electrical Engineering, the University of Edinburgh from Mar.1999-Oct.1999. Dr. Jiangs research interests include optical switching networks, routers, network coding, WDM networks, VoIP, interconnection networks, IC yield modeling, timing analysis of digital circuits, clock distribution and fault-tolerant technologies for VLSI/WSI. He has published over 150 referred technical papers in these areas. He is a senior member of IEEE.