Minimum cost multicast trees in ad hoc networks

Aleksi Penttinen
Networking Laboratory, Helsinki University of Technology
P.O.BOX 3000, FIN-02015 HUT, Finland
Aleksi.Penttinen@tkk.fi

Abstract— **We propose a simple nearly-optimal heuristic algorithm for multicast tree construction in static ad hoc networks. The objective is to deliver a data packet from a source node to a set of destination nodes with a sequence of transmissions so that the sum of given transmission-related costs is minimized. Also a brute-force enumeration method is described to obtain optimal solutions to the problem in unit cost case. Performance of the algorithms is compared against existing methods by simulations.**

I. Introduction

Consider a wireless multihop network of simple wireless nodes which communicate with their neighbors with omni-directional transmissions. In each transmission all the neighbors of the transmitting node receive the same data packet simultaneously. As packets are relayed towards their destinations over multiple hops, each relay transmission is assumed to have a cost. Our problem is to find a minimum cost multicast tree, i.e. a set of transmissions, which can be used to forward a packet from a given source node to every node in a given destination set so that the sum of transmission costs is minimized. We refer to this problem as the wireless multicast tree problem, WMTP for short.

Multicast trees enable cost optimization of one-to-many communications in static or slowly varying ad hoc networks, sensor networks and multihop extensions of base station-based wireless networks. The costs may represent factors related to e.g. incurred load, energy consumption, queuing or processing delay, security, reliability or detectability. Using a suitable routing tree, the corresponding data transfer has often substantially lower costs compared to the two alternatives, sequential unicast transmissions or flooding. The advantages of multicast routing become increasingly evident if either the transferred data are large or the resources are scarce.

In this paper we treat WMTP as a packet forwarding problem, but equivalently the same algorithms can be applied in building multicast trees over any directed graph where the costs are related to nodes rather than links. Trees can be constructed on operational time scales to route certain transmission or off-line, e.g. to provide solutions to multicast routing subproblems in cross-layer optimization schemes of wireless networks. Example applications are control traffic delivery in sensor networks, distribution of multimedia stream

from a gateway to wireless clients and radar information dissemination in a military network. The multicast tree approach is limited to fairly static networks and applications where the cost information can be kept up-to-date. In highly dynamic networks resource optimization is hardly possible and one-to-many communications require less subtle means such as flooding variants.

A well-known and important instance of WMTP is the broadcast problem in ad hoc networks. In the broadcast problem a node attempts to send a message to all nodes in the network so that the number of retransmissions is minimized. In this case, the destination set consists of all nodes but the source and all the transmission costs are identical. This problem can be seen as (constrained) minimum connected dominating set problem which remains NP-hard even in unit disk graphs [1], a fact which is a strong motivation for development of simple heuristics for the broadcast problem and for WMTP in general.

Indeed, the broadcast problem has been studied widely in ad hoc networks, see [16] for an extensive overview, but general multicast trees have received much less attention. In [11] an efficient distributed algorithm for unit weight WMTP was proposed. The main contribution of the present work is to introduce even more efficient, yet centralized, tree construction method which adapt itself also to the general weighted WMTP where the transmission costs (node weights) may be arbitrary. The proposed novel greedy algorithm adds nodes to the multicast tree one-by-one so that the sum of shortest paths to receivers is maximally decreased. This simple, but rather coarse idea is shown to be versatile and perform extremely well compared to the other intuitive methods adapted from wireline networks, where the corresponding problem is solved using Steiner tree heuristics [13]. We give also an efficient enumeration method that can be used to find optimal solutions in the unit weight case.

The organization of the rest of the paper is as follows. Section II states the network model and problem definition. Section III gives an overview on multicast tree construction algorithms in ad hoc networks. Section IV presents the proposed algorithms. Section V gives a detailed description of the existing multicast algorithms against which the performance of the contributions are evaluated in Section VI. Section VII concludes the paper.

II. Network model

We study a multihop network where nodes communicate with their neighbors using wireless links. The following as-

sumptions are made:

- Neighborhood of a node is the set of nodes which can receive a packet transmitted by the node. Any packet transmitted by a node are received by all its neighbors.
- Each node may have an arbitrary packet transmission cost, no other cost are involved. We use the term (node) weight interchangeably with the term cost.
- Neighborhoods and transmission costs are assumed to remain fixed in the routing time scale.

Note that the neighborhood relation do not need to be reciprocal and the nodes do not have to be identical. Neighborhood relation is used to model the effects of transmission parameters such as power, coding and modulation of the nodes.

WMTP can be defined formally as follows. We attempt to find a minimum weight tree (in a directed graph) which is rooted at a given source vertex and contains a set of destination vertices. The weights are associated to vertices rather than edges and only non-leaf tree vertices are accounted for the total weight of the tree. In other words we seek a minimum weight connected set of nodes, which originates from the source node and dominates all the receiver nodes.

We are inclined to treat the unit weight case and the general weighted case of WMTP separately, because existing reference results are limited to the former scenario.

III. OVERVIEW ON MULTICAST ROUTING ALGORITHMS IN AD HOC NETWORKS

Most of the novel routing challenges brought forth by wireless and infrastructureless technology can be addressed by simple conversions to shortest path problems. These challenges, often energy or reliability concerns, are reflected to link/node costs after which standard shortest path algorithms can be used to find optimal unicast routing.

Optimal multicast routing, however, can generally be obtained only through explicit enumeration. This is the case even in the wireline networks, which corresponds to the Steiner tree problem. Consequently, simple sub-optimal heuristics are usually required for decisions taking place in the time scales of routing. An overview and elegant implementation of a class of practical (and most common) approximation algorithms to the wireline multicast routing is given in [13]. Unfortunately, the proposed algorithms as such lack efficiency in wireless multihop networks and, thus, novel methods are needed.

The topic of the present paper, the wireless counterpart of the Steiner tree problem, WMTP, has not been treated extensively in the literature. One of the very few references to the present problem, [6], introduces two heuristic algorithms for the unit weight case. Despite the proven worst case approximation guarantees, the algorithms leave much room for improvement in practical applications. A simple distributed algorithm based on unicast routing information for the unit weight case was given in [11], but the generalization of the method to the weighted case is not straightforward. Practical approaches to the problem were also suggested in [4], [3], where the idea is to construct a virtual mesh that connects the multicast group members by unicast tunnels and compute a

source based delivery tree on the mesh. This approach lacks efficiency.

Although the multicast tree problem is not that widely studied in ad hoc networks, its special case of broadcasting has received a significant amount of interest. Broadcasting is applied widely in the networks for information discovery and update. Frequently occurring broadcasts can constitute a significant share of the network load and, accordingly, development of efficient broadcasting methods is of utmost importance. In [16] the authors present an extensive overview and comparison study on the practical broadcasting methods proposed for ad hoc networks. More theoretical approaches to the broadcast problem can be found, e.g., in [14], [10], [1], [5]. Naturally, the algorithms proposed in this paper can also be used to determine efficient broadcast trees.

A closely related problem from graph theory, the node weighted Steiner tree, is addressed in [9] and [7]. The main difference of this problem to WMTP is that in WMTP we seek a connected set of nodes that dominates the receivers rather than contains them.

Another related line of thought addresses multicast tree construction within a different network model. Assume that the nodes can adjust their transmission power to avoid unnecessary energy usage. In this setting each node has a (typically discrete) set of operating modes each of which corresponds to a certain cost and a certain subset of neighbors that are reached. In this scenario the multicast tree assigns both the nodes and their power level e.g. to minimize the energy consumption [15], [12]. In the present work we do not consider this problem. Instead, we assume that the set of neighbors is fixed for each node.

IV. NOVEL ALGORITHMS FOR MULTICASTING IN AD HOC NETWORKS

A. Exact solution for the unit weight case

We outline here an enumeration method that can be used to find optimal tree sizes for unit weight WMTP, i.e to minimize the number of transmissions in the multicast tree.

Even the unit weight multicast tree problem is generally rather tedious to solve. An important observation is that although the set of possible trees that originate from a multicast source and consist of i transmissions is typically large, the set of maximal covers spanned by the sets of i connected transmissions (one of which is the source) may be relatively small. By a cover we refer to the set of nodes which hear the transmission from any of the transmitting node.

For example, a random 50-node network on a unit square with the node transmission range 0.286 gives us the following statistics: There are 80688 different 6-node directed trees that originate from a certain node, but the number of different covers spanned by these transmissions is only 802. Moreover, of these only 24 are maximal in the sense that they are not subsets of any other cover. If the set of destinations is not a subset of any of these 24 covers, we may proceed with the search by enumerating all the 7-node maximal covers, which are easily obtained from the maximal 6-node covers.

We outline here an enumeration algorithm for finding the minimum multicast tree size. Let T be the set of destinations, C a cover (i.e. set of nodes), and $\mathcal{S}_i$ the set of maximal covers spanned by i connected nodes (including the source). Furthermore, let $N(n)$ be the set of neighbors of the node n and let s be the multicast source. Starting with $\mathcal{S}_0 = \{\{s\}\}$ repeat the recursion step summarized as Algorithm 1.

Algorithm 1 Optimal multicast tree recursion step

for each $C \in \mathcal{S}_i$ **do**
    for each $n \in C$ **do**
        $cover \leftarrow C \cup N(n)$
        if $\nexists C \in \mathcal{S}_{i+1}, \ cover \subseteq C$ **then**
            if $T \subseteq C$ **then**
                Output $(i+1)$. STOP.
            end if
            $\mathcal{S}_{i+1} \leftarrow \mathcal{S}_{i+1} \backslash \{C \in \mathcal{S}_{i+1} | C \subseteq cover\}$
            $\mathcal{S}_{i+1} \leftarrow \mathcal{S}_{i+1} \cup \{cover\}$
        end if
    end for
end for

The algorithm can be conveniently implemented if the covers C are represented using, e.g. binary arrays. To find an optimal multicast tree one must additionally keep track of the nodes associated to the covers. From the nodes associated to an optimal cover one can form a multicast tree by any traversal method starting from the source. Although the algorithm could be used directly for finding optimal multicast trees for routing in small networks, it is sensitive to the multicast tree size and the computation time increases quickly after certain limits have been achieved.

B. Heuristic solution to the weighted case

Assume that the all-pairs shortest path information is available in terms of given node weights. The node weight information is fully contained in the path lengths and we may propose the following greedy algorithm for general WMTP.

During the algorithm each node belongs to one of the following three sets: covered nodes, transmitting nodes or idle nodes. The set of covered nodes contains all the non-transmitting nodes who have the data packet, i.e. which are neighbors of a transmitting node closer to the source. Idle nodes are not part of the multicast operation.

The novel idea is very straightforward. Initialize all nodes "idle" and set the source node as "covered" as it has the data packet. We construct the multicast tree then iteratively. In each iteration round we select one node from the covered nodes, mark it as transmitting and mark all of its idle neighbors as covered. The selection is carried out so that the sum of the minimum distances from the transmitting nodes to each of the non-covered multicast receivers is maximally decreased. The iteration stops as soon as all multicast receivers are covered.

Figure 1 illustrates the algorithm. The practical performance of the algorithm is illustrated in the next section.
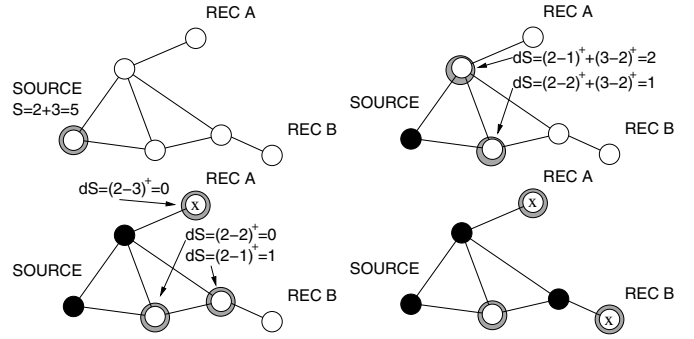

Fig. 1. Example: Minimize the number of transmission needed to connect SOURCE to REC A and REC B. All node weights are unity, thus the shortest paths are given in terms of hop count. Let S be the sum of minimum distances between transmitting nodes and non-covered receivers. To start, mark source as covered and initialize S (see top left figure). In the iteration step, compute dS, the decrement in S if the node was added to the transmitting nodes, for each covered node. Mark the node with maximum dS as transmitting and mark all its idle neighbors as covered. Iteration step is repeated until all receivers are covered. After the selection of the second transmitting node (see bottom left figure) REC A is covered and S contains only the distance to REC B.

Let $\mathbf{D}$ be the all-pairs shortest path matrix. Let $\mathbf{D}_i$ denote the i:th row of the matrix. The pseudocode implementation of the algorithm is described in Algorithm 2.

Algorithm 2 Centralized multicast tree construction

$K \leftarrow \{s\}, \ M \leftarrow \emptyset$
$\mathbf{c} \leftarrow \mathbf{D}_s$
while $T \neq \emptyset$ **do**
    $n \leftarrow \arg\max_{i \in K} (\mathbf{c}(T) - \mathbf{D}_i(T))^+ \cdot \mathbf{1}$
    $M \leftarrow M \cup \{n\}$
    $K \leftarrow (K \cup N(n)) \backslash M$
    $T \leftarrow T \backslash N(n)$
    $\mathbf{c}(T) \leftarrow \min(\mathbf{c}(T), \mathbf{D}_n(T))$
end while
Output M

In the algorithm $(\cdot)^+$ and $\min$ are taken component-wise from the vectors. The set K represents already covered, but not transmitting, nodes. In each round we select the node which maximally reduces the sum of shortest distances. The worst case complexity of the algorithm in this straightforward implementation is $O(N^2 T)$, since selection of the next node has complexity $O(NT)$ and the outer loop will be repeated at most $N - 2$ times. Naturally the shortest path information needs to be gathered separately.

C. Practical remarks

The requirement for all-pairs shortest path information renders the proposed algorithm centralized. Shortest path information may be obtained by standard algorithms [2] or from unicast routing tables. The required information is a minor concern if the multicast trees are constructed for off-line use, e.g. as a part of cross-layer optimization tasks. However, the requirement limits the applicability of the algorithm in
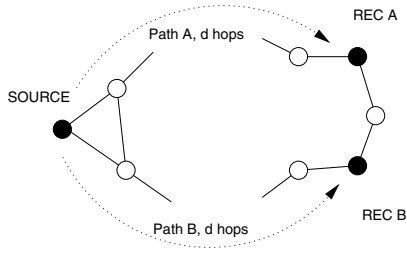
Fig. 2. Worst case example.

on-line routing decisions to fairly static networks with fairly static costs. Wherever global cost information can be kept up-to-date, the algorithm allows generation of highly efficient multicast trees. If the cost information cannot be maintained at all, cost effective multicasting is hardly possible by any approach and one has to resort to flooding variants.

The average performance of the algorithm is excellent, but some rare particular instances may cause the algorithm to perform badly. Consider, for example, a situation depicted in Figure 2. There are two receivers, path metric is the hop count, receivers are two hops from each other and the shortest hop paths from the source to the receivers (paths A and B) are node-disjoint and of the same length $d >> 2$.

In principle, it is possible that the proposed algorithm would result in a tree which is the concatenation of paths A and B instead of the optimal – path A or B and the two hop path separating the receivers. This happens if the transmitting nodes are selected in turns, so that every second node belongs to path A and the ones in between to path B. This is a feasible (yet unlikely) selection order for the algorithm; the reward of selecting a node from either of the paths is 1 and remains so unless other path is selected twice in a row. The problem can be easily avoided by breaking the ties in the cost function appropriately, e.g. by artificially decreasing the all distances to a receiver by a small $\epsilon > 0$ every time a transmitting node decreases the shortest distance to this receiver.

V. Existing Algorithms

Almost all existing algorithms for WMTP are limited to special cases of the problem. Only the methods adapted from fixed networks can be used for the general weighted WMTP. In this section we describe the algorithms that are compared against the proposed algorithm in Section VI.

There are three commonly used heuristic algorithms for the Steiner tree problem [13]. Pruned Dijkstra (PD) method finds the shortest paths to each node, i.e. a shortest path tree, and then prunes the tree by iteratively removing all leaf nodes which are not multicast receivers. Minimum-Cost Path (MCP) method entails incrementally adding multicast receivers to the multicast tree. In each round a receiver closest to the existing tree (which is initially the source node) is grafted to the multicast tree using the path corresponding the shortest distance. The third heuristic algorithm, Distance Network Heuristic, comprises of constructing a minimum spanning tree

on the distance network spanned by the receiver nodes and then replacing the spanning tree edges by the corresponding minimum cost paths in the original graph. Finally the multicast tree is constructed by pruning all non-receiver leaf nodes.

These algorithms are easy to adopt in the wireless environment to obtain results for comparison in the general weighted multicast problem. We consider the pruned Dijkstra method because of its simplicity and minimum cost path heuristic because of its efficiency. Pruned Dijkstra method remains unmodified in ad hoc networks and is simply the concatenation of shortest paths to the receivers. Minimum cost path is easy to change to consider paths from covered nodes instead of tree nodes. This way we avoid counting the transmission cost of any node more than once.

If we limit our study to the unit weight case of WMTP, very good multicast trees can be obtained by the distributed algorithm in [11] (DMT), which utilizes similar ideas as the proposed algorithm, but is localized. This localization carries a cost in efficiency as the nodes have to fix also which next hop nodes forward to which receivers. The algorithm proposed by Guha et al. [6] constructs first a small dominating set and then connects it using an edge weighted Steiner tree approximation algorithm. It can be proven that this algorithm has an approximation factor $O(H(\Delta))$, where Δ is the maximum degree of a node in the network, and $H()$ is the harmonic function. Also optimal cases can be considered with the enumeration method presented earlier in Section IV-A.

Further by limiting the study to consider only the unit weight broadcast we arrive at the minimum connected dominating set problem for which many heuristics exist [16]. However, the simple greedy algorithm presented below (Algorithm 3) is often much more efficient in practice, especially in unit disk graphs [8]. The downside of this simple method is that it cannot be directly generalized to the multicast case. On the other hand the proposed algorithm can be seen to fulfill this role; it differs from Algorithm 3 only by the selection of the next transmitting node.

Algorithm 3 Greedy algorithm for MCDS

$M \leftarrow \{s\}$
while $T \neq \emptyset$ **do**
$\quad n \leftarrow \arg\max_{i \in N(M)} |N(n) \cap T|$
$\quad M \leftarrow M \cup \{n\}$
$\quad T \leftarrow T \backslash N(n)$
end while
Output M.

VI. Simulation Studies

In this section we compare the performance of the algorithm to other available proposals discussed in the previous section. Analytical analysis of the algorithm is of minor value as the performance bounds are dominated by pathological cases such as discussed in Section IV-C. In practical considerations the average performance is much more important than performance bounds unless they can be made very tight. Despite

the pathological cases, the algorithm succeeds to produce highly efficient multicast trees on average, outperforming others alternatives by a clear margin.

To illustrate the practical performance of multicast tree algorithms, we carry out an extensive numerical study in three special scenarios; unit weight broadcast, unit weight multicast and weighted multicast. These scenarios are treated separately because the set of existing algorithms available in scenarios differ.

In the experiments we consider simulation cases which are parameterized by the vector $(n, R_{\max}, m, \mathbf{w})$. n is the number of nodes, which are placed randomly (from an uniform distribution) in an unit square. We model the network as unit disk graph, where $R_{\max}$ is the transmission range, i.e. two nodes are neighbors of each other if their mutual distance is less than $R_{\max}$. We have fixed $R_{\max} = 0.286$ (in the unit square) in all studies to allow direct comparisons to [16]. The size of the set of multicast receivers is denoted by m. The source and receiver nodes are selected randomly among the n nodes. Finally, $\mathbf{w}$ is the vector of node weights or costs.

For each different case or parameter combination we have generated 200 random instances of the multicast problem, where both the node locations and the selection of source and receiver nodes are randomized. Each instance is then solved using all the methods in comparison. The results shown in the figures are averages over the 200 instances.

A. Unit weight broadcast

To study the broadcast scenario, we consider random networks of varying sizes, $n = 20, ..., 70$, and minimize the number of transmissions (i.e. $w_i = 1$, $\forall i$) in a source based broadcast tree ($m = n-1$). The algorithms we consider are PD and (modified) MCP [13], DMT [11], Algorithm 3, Algorithm 2, algorithm by Guha et al. [6] (Note that [6] provides also specialized algorithms for this connected dominating set scenario, but we consider an algorithm that is suitable for general multicast) and optimal obtained by enumeration as stated in Section IV-A. Figure 3 shows that the proposed algorithm performs equally well as the specialized Algorithm 3 in this scenario.

B. Unit weight multicast

We consider networks with $n = 50$ nodes and vary the number of multicast receivers, $m = 7, ..., 49$. Weights are again constant ($w_i = 1$, $\forall i$). With $m = 49$ we have the broadcast scenario analyzed above. The algorithms in consideration are PD, MCP, DMT Guha and optimal. Figure 4 shows the obvious benefits of the proposed algorithm. It clearly outperforms all the other alternatives in the multicast case. Despite the proven performance guarantee of the algorithm by Guha et al. its practical performance is not very good.

C. General weighted multicast

For general weighted multicast problems only applicable algorithms in comparison are PD and MCP. Similarly to the unit weight case, we consider networks with $n = 50$ and $m =$
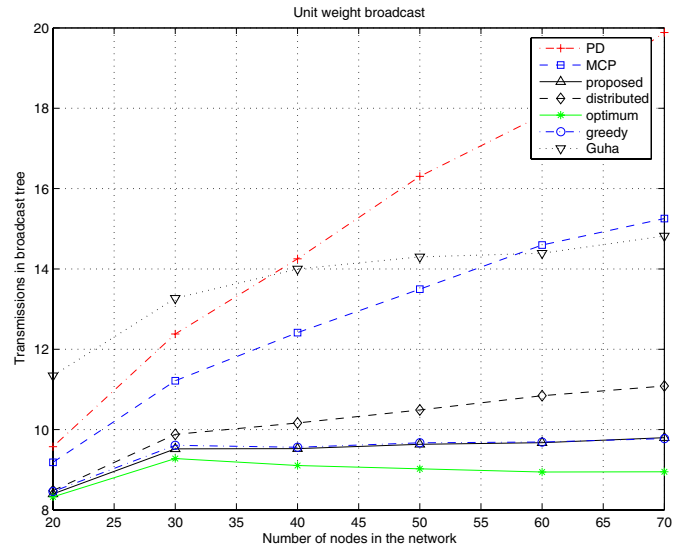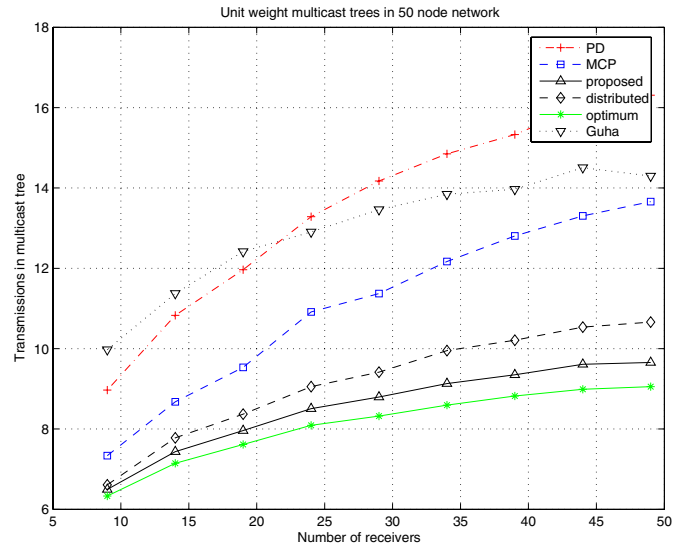

Fig. 3. Broadcasting

Fig. 4. Multicasting

$7, \ldots, 49$, but now each node is given a random integer weight, $w_i \in \{1, ..., 100\}$. Again the proposed algorithm outperforms clearly the alternatives, as shown in Figure 5.

VII. CONCLUSION

Whereas broadcast trees and connected dominating sets have been studied thoroughly in the context of wireless multihop networks, its generalized version, the multicast tree problem has received little attention. Furthermore, even the broadcast case cannot be handled efficiently if the nodes are associated with a transmission cost, or a weight, other than unity.

Despite the apparent lack of results in the area, WMTP is a central problem in wireless networks. Minimum cost trees
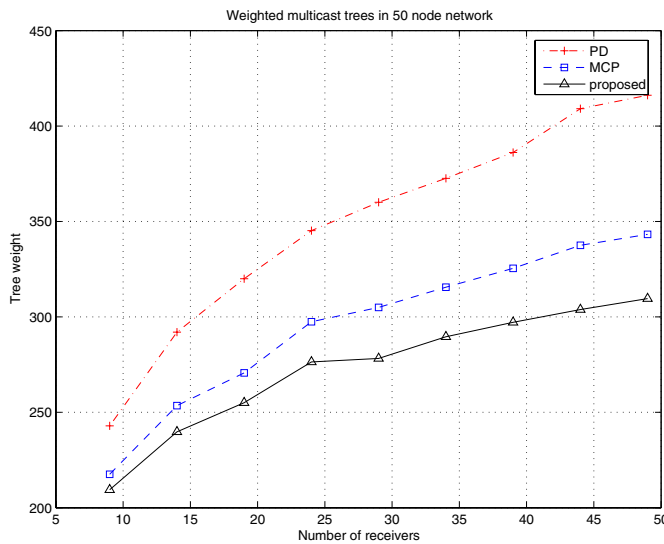
Fig. 5. Weighted multicasting

are required both for optimizing the performance of a single multicast transmission as well as for solving subproblems of larger scale optimization tasks considering e.g. load balancing or network lifetime.

In this paper we have introduced and experimentally analyzed a simple multicast tree algorithm that can be used to find nearly minimum cost multicast trees. The benefits of the algorithm are indeed its simplicity, efficiency and adaptivity to generalized weighted problems.

The downside of the proposed algorithm is that it relies on the availability of valid all-pairs shortest path information, which makes it non-suitable in applications where maintenance of the information is impossible or expensive. The maintenance costs depends on the dynamics and the size of the network and on the nature of the cost information.

Further work is required to develop distributed algorithms for general weighted WMTP. Distributed operation can be easily implemented by separating WMTP in each tree node into subproblems which are then passed forward to next hop tree nodes for solving. Based only on local information, each node needs to decide jointly the next hop forwarding neighbors in the multicast tree and their respective subsets of multicast receivers. The exact mechanism how to implement this efficiently in the weighted case remains to be determined.

REFERENCES

[1] K. M. Alzoubi, P.-J. Wan, and O. Frieder, "Distributed heuristics for connected dominating sets in wireless ad hoc networks," *Journal of Communications and Networks*, 4(1), Mar. 2002.

[2] D. Bertsekas and R. Gallager, *Data Networks*, Prentice Hall, 2nd edition, 1992.

[3] G. Chao and P. Mohapatra, "Efficient overlay multicast for mobile ad hoc networks," in *Proc. of IEEE Wireless Communications and Networking 2003*, 2003, pp. 1118–1123.

[4] K. Chen and K. Nahrstedt, "Efficient location-guided tree construction algorithms for small group multicast in MANET," in *Proc. of INFO-COM*, 2002, pp. 1180–1189.

[5] R. Gandhi, S. Parthasarathy, and A. Mishra, "Minimizing broadcast latency and redundancy in ad hoc networks," in *Proc. of MobiHoc'03*, 2003, pp. 222–232. Corrected version: http://crab.rutgers.edu/~rajivg/pubs.html.

[6] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, 20(4), pp. 374–387, 1998.

[7] S. Guha and S. Khuller, "Improved methods for approximating node weighted steiner trees and connected dominating sets," *Information and computation*, 150, 1999.

[8] D. Jurca and J.-P. Hubaux, "Joint synchronization, routing and energy saving in CSMA/CA multi-hop hybrid networks," in *Proceedings of IEEE MASS*, 2004, pp. 245–254.

[9] P. Klein and R. Ravi, "A nearly best-possible approximation algorithm for node-weighted steiner trees," *J. Algorithms*, 19(1), pp. 104–115, 1995.

[10] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc networks," in *Proc. of MSWIM*, 2000, pp. 61–68.

[11] A. Penttinen, "Efficient multicast tree algorithm for ad hoc networks," in *Proceedings of IEEE MASS*, 2004, pp. 519–521.

[12] A. Penttinen and J. Virtamo, "Improving multicast tree construction in static ad hoc networks," in *Proceedings of IEEE LCN*, 2003, pp 762–765.

[13] S. Ramanathan, "Multicast tree generation in networks with asymmetric links," *IEEE/ACM Transactions on Networking*, 4(4), pp. 558-568, Aug. 1996.

[14] I. Stojmenovic, M. Seddigh, and J. Zunic, "Dominating sets and neighbor elimination-based broadcasting algorithms in wireless networks," *IEEE Transactions on parallel and distributed systems*, 13(1), pp. 14-25, Jan. 2002.

[15] J. Wieselthier, G. Nguyen, and A. Ephremides, "Energy-efficient broadcast and multicast trees in wireless networks," *Mobile Networks and Applications*, 7(6), pp. 481–492, Dec. 2002.

[16] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proc. of MobiHoc'02*, 2002, pp. 194–205.