Int J Wireless Inf Networks (2015) 22:205–219
DOI 10.1007/s10776-015-0272-1

Minimizing Expected Transmissions Multicast in Wireless Multihop Networks

Kostas Christodoulopoulos[1] · **Christos Papageorgiou**[1] · **Emmanouel Varvarigos**[1]

Received: 9 July 2014 / Accepted: 16 May 2015 / Published online: 30 May 2015

Abstract We address the multicast problem in a wireless multihop network in the presence of errors. Finding the multicast schedule that minimizes the expected number of transmissions, while accounting for both the wireless broadcast advantage (WBA) and the wireless unreliable transmission (WUT) characteristics, has not been solved optimally before. We formally define the problem and propose an initial algorithm of non-polynomial complexity to obtain the optimal solution. The algorithm starts by transforming the wireless network topology into an auxiliary expanded graph that captures the WBA property, and assigning appropriate weights to the links so as to capture the WUT property. The proposed transformation is interesting on its own merit, since it permits us to consider only point-to-point transmissions (as in wireline networks) in the expanded graph, instead of the point-to-multipoint transmissions present in the original wireless network, and can also be useful in other optimization problems some of which we briefly describe. By solving the minimum Steiner tree problem on the expanded graph we obtain the optimal multicast solution for the initial graph. Since the optimal algorithm is of non-polynomial complexity, we also propose a number of heuristic algorithms. In particular, we first present a truncated graph transformation and then describe two heuristic algorithms for obtaining the multicast schedule. Simulation results show that the proposed heuristics have performance close to that of the optimal algorithm, at least for the instances for which we were able to track optimal solutions, while outperforming other heuristic multicast algorithms.

Keywords Wireless multihop networks · Multicasting · Wireless broadcast advantage · Wireless unreliable transmission · Graph transformation

1 Introduction

Communication among the nodes of a wireless multihop network takes place in a hop-by-hop fashion, so that each node, in addition to sending and receiving the packets originating from and destined to it, respectively, also acts as a relay for forwarding other packets transferred over the network, without the use of any fixed infrastructure or centralized coordination. Thanks to their unique characteristics, wireless multihop networks have long received considerable attention in the research community and are currently beginning to establish a place in the communication market.

The wireless medium poses challenges to network protocol designers that are different than those found in the wireline world, but at the same time, its unique properties, if appropriately exploited, also open up opportunities for more efficient performance. A primary example is the innate broadcast character of the wireless medium, often referred to as the wireless broadcast advantage (WBA) [1]. In contrast to wireline point-to-point links, wireless transmissions are by default point-to-multipoint: when a node

✉ Kostas Christodoulopoulos
kchristodou@ceid.upatras.gr

Christos Papageorgiou
xpapageo@ceid.upatras.gr

Emmanouel Varvarigos
manos@ceid.upatras.gr

[1] Department of Computer Engineering and Informatics, University of Patras, Patras, Greece

transmits, all nodes within a transmission range, determined by the transmission power used, can hear the transmission. Nevertheless, in the early stages of research in wireless multihop networks and until recently, researchers were simply transferring ideas from the wireline field, ignoring the benefits that could be obtained from the WBA.

Another basic characteristic of the wireless medium is that it is relatively unreliable, lossy, and time-varying, due to fading, mobility or interference. We refer to this as the wireless unreliable transmission (WUT) characteristic, which makes wireless transmissions less deterministic. In unicast point-to-point routing, the WUT can be accounted for through the expected number of transmissions (ETX) metric [2]. In the case of point-to-multipoint transmission, the probability of successful reception may vary significantly across the multiple receivers. Actually, it has been observed that the probabilities of correct reception at each end are independent of each other [3], and thus, recently, variants of the ETX metric have been proposed for multicasting [4].

Multicast is a primitive communication task that appears in many important applications of wireless multihop networks, including data gathering (sensor networks), network state/topology information dissemination (mesh or ad hoc networks), event notification (vehicular networks), video distribution and general one-to-many content delivery in urban backbone networks, and other applications. As a result, devising efficient algorithms to perform it has attracted significant research interest [7–17]. However, the problem of finding the minimum cost multicast tree in traditional wireline networks with point-to-point links (the so called minimum Steiner tree) is NP-complete even when all links have the same cost [5, 6]. Ruiz et al. [7] realized that in wireless multihop networks, calculating the minimum Steiner tree does not yield the solution with the minimum number of transmissions when the WBA and the WUT properties are taken into account. This led them to propose new multicasting heuristics for such networks. Zhao et al. [8] proposed a metric, called expected multicast transmissions (EMT), which captures both properties through the ETX required by a node to successfully transmit a packet to a subset of its neighbors under certain probabilistic assumptions. The authors also proposed a heuristic algorithm for constructing a good EMT schedule, without however obtaining an optimal solution.

In the present work we describe an optimal algorithm for obtaining the multicast schedule with the minimum ETX, referred to as the minimum expected multicast transmissions (abbreviated MinEMT) problem. The proposed algorithm considers an error-prone medium (WUT) and takes into account the WBA property as well. Even though MinEMT seems to be a basic problem, its optimal solution has not, to the best of our knowledge, been produced before. In the process of describing the MinEMT optimal algorithm, we define a graph transformation that produces an auxiliary expanded graph from the original wireless network topology. The expanded graph accounts for the WBA characteristic and is interesting on its own right, since on that graph only point-to-point transmissions (as in wireline networks) have to be considered. The WUT property is also captured by appropriately defining the weights for the links of the expanded graph. The proposed transformation is general and can be used to solve other optimization problems in wireless networks, some of which we briefly discuss. This graph transformation is non-polynomial for dense networks with node degrees that are super-logarithmic in the number of nodes, in which case we also define a truncated graph transformation of polynomial complexity. The optimal MinEMT algorithm has non-polynomial complexity on both the expanded and on the truncated expanded graphs, thus we also present polynomial time heuristic algorithms that achieve most of the performance gains of the optimal algorithm with considerably less computational effort.

The performance of the proposed optimal and heuristic algorithms was evaluated using simulations and compared to that of other heuristic algorithms. Our results show that while the optimal algorithm is able to find minimal expected transmissions multicast solutions for a wide variety of scenarios, it is unable to track solutions for large input instances in reasonable time. The heuristic algorithms achieve performance that is comparable to that of the optimal algorithm in low running times, while outperforming other multicast algorithms proposed in the literature.

The remainder of the paper is organized as follows. In Sect. 2 we present the related work. In Sect. 3 we propose a graph transformation that captures the WBA and WUT properties. In Sect. 4 we formally describe the minimum expected transmissions multicast problem and present an optimal algorithm to solve it. To tackle complexity issues, heuristic algorithms are proposed in Sect. 5. The simulation experiments follow in Sect. 6. Finally, Sect. 7 concludes the paper.

2 Background Work

The multicast problem in a traditional network with point-to-point links can be formulated as a minimum Steiner tree problem, which is a well-known NP complete problem [5]. Several heuristics for the minimum Steiner tree problem running in polynomial time have been proposed in the literature [6]. The multicast problem in a wireless network is quite different, however, due to the WBA and the WUT characteristics.

Several wireless-specific protocols with emphasis on the distributed nature of the operation have been proposed, a survey of which can be found in [11, 12, 17]. Most protocols define a metric capturing the optimization criterion of interest and calculate either Shortest Path Trees or pruned Minimum Spanning Trees based on that metric. The authors in [4] summarize the various metrics proposed for the multicast problem in wireless networks. Many works have also examined multicasting in wireless networks for energy-efficiency purposes. Two such schemes to perform energy-efficient routing and transmission power assignment for multicasting, based on the minimum spanning tree concept, are presented in [13, 14]. The work in [15] examines the problem using a multi-constrained optimization approach and presents an energy-efficient algorithm for the case of error-free links. In [16] the authors transform the original network graph by inserting additional nodes expressing the broadcast nature of each transmission and then calculate the minimum Steiner tree. In all the above works, however, the wireless medium is assumed to be perfect, in the sense that if a node transmits a packet to a node that is located close enough, the communication is successful.

The introduction of the ETX [2] as an appropriate metric for unicast point-to-point routing in error-prone wireless networks (captured by the WUT property) gave rise to works that try to apply the same idea in multicasting. In [7], multicasting is examined under the WUT model and a heuristic algorithm is presented. The authors in [8] extend these ideas to obtain a metric that captures WBA and WUT, without however proposing an optimal algorithm for multicasting. The present paper builds on the aforementioned line of research, presents the optimal algorithm for the multicasting problem in a wireless multihop network, taking into account both the WBA and WUT characteristics of the wireless medium. It also provides a methodology for addressing other optimization problems under these medium characteristics. The authors in XXX proposed an opportunistic protocol for multicast transmissions and also examine the reduction of protocol overheads.

3 Model, Notation, and Graph Transformations

The wireless multihop network is defined as a graph $G = (V, E)$, where V is the set of nodes and E is the set of edges connecting them. In contrast to the often used "sphere" model (where each mobile host has a transmission range represented by a sphere centered at itself), in our model a transmission from mobile host v_i to mobile host v_j is a Bernoulli trial with probability of success p_{ij}. In particular, we will say that a *trusted point-to-point transmission* over edge (v_i,v_j) has been completed when v_j has correctly received and acknowledged the packet sent to it by v_i.

An edge $(v_i, v_j) \in E$ exists, if the probability p_{ij} of node v_j correctly receiving a transmission from node v_i and acknowledging it back is above a certain threshold p_t; in that case we will say that v_j is *within the transmission range* of v_i. Note that the existence of a link $(v_i, v_j) \in E$ does not indicate a trusted delivery of a packet over it in the first transmission (the success is actually defined by p_{ij}), and that the definition of p_{ij} incorporates the reception of the ACK packet as well. Note that in both cases when the forward or ACK packets are lost we assume that we retransmit after a timeout. So we can consider the forward and backward transmission as a single event and model it as a single Bernoulli trial. In general and we could consider both events separately, that would result in a different definition for the probability p_{ij}. Note also that the threshold p_t can be as low as zero, but in this case an infinite number of retransmissions are needed to achieve a successful transmission. Depending on the network at hand (number of nodes, multicast set, etc.) and the performance or running time requirements, the value of the threshold can be chosen so as to allow links that would on average require a certain number, e.g. 10, of transmissions.

The success probability p_{ij} captures the diverse transmission, physical and other parameters affecting the correct delivery of a packet over (v_i,v_j), such as the distance and the obstacles between v_i and v_j and their possible mobility, the transmitter's power and the receiver's sensitivity, the use or not of FEC, the MAC protocol used and the traffic load present (that determine the interfering traffic and, consequently, the possibility of collisions), etc.

Since the wireless medium is a broadcast channel (WBA property), a transmission from v_i can be heard by a set V_i of receivers that are within v_i's transmission range. In other words, V_i is the set of nodes that are within the transmission range of v_i under our model:

$$v_j \in V_i \text{ if } \left(v_i, v_j\right) \in E \text{ (or, equivalently, if } p_{ij} > p_t).$$

When node v_i wants to intentionally and reliably (as opposed to opportunistically) transmit a packet to a set $R \subseteq V_i$ of receivers that are within its transmission range, we will refer to it as a *trusted single hop point-to-multipoint transmission*, or *single hop multipoint transmission* for short, and will denote it by (v_i,R). A trusted single hop multipoint transmission is considered to be completed when all the nodes in the intended receiving set R have correctly received and acknowledged the packet. This may entail a number of retransmissions that are carried out by the underlying MAC protocol with an appropriate acknowledgement mechanism, which we assume to exist (e.g. IEEE 802.11 protocol [24]).

Each edge (v_i,v_j) in G is characterized by a weight w_{ij}, and will be viewed as the cost of the trusted point-to-point transmission (v_i,v_j). The weight w_{ij} may depend on p_{ij} as well as other parameters, and different definitions of the weight will give rise to different optimization problems for the wireless multihop network.

We also assume that we are given (or we have a way to find, given the definition of the w_{ij}'s) a function f that computes the weight w_{iR} of the single hop multipoint transmission (v_i,R) by the weights w_{ij} for all edges (v_i,v_j), $v_j \in R$, that is,

$$w_{iR} = f(w_{ij} | v_j \in R) \tag{1}$$

The definition of w_{iR} (equivalently, of the function f) depends on the optimization problem at hand; the only constraint is that we should have $w_{i,\{v_j\}} = w_{ij}$ for consistency [$R = \{v_j\}$ representing the singleton set of v_j, and w_{ij} representing the cost for the point-to-point transmission (v_i,v_j)]. For the general case where $R \neq \{v_j\}$, w_{iR} should be defined as the cost of the single hop multipoint transmission (v_i,R) for the optimization function at hand (see also Sect. 3.4).

3.1 Graph Expansion to Capture the WBA Property

Since single hop multipoint transmissions are harder to visualize and work with than point-to-point transmissions, we will find it useful to introduce a transformation of the original network graph G into an auxiliary graph G', such that single hop multipoint transmissions in G correspond to point-to-point transmissions in G'.

- through a directed virtual edge (v_i,v_R) with weight w_{iR} calculated by Eq. (1), and
- through directed virtual edges (v_R,v_j) for all $v_j \in R$ of zero weight.

In other words, for each set of receivers R within v_i's transmission range, we create a virtual node v_R in G' that acts as an intermediate node to connect v_i to all nodes in set R. Virtual node v_R represents the case in which v_i transmits until all nodes in R receive and acknowledge successfully, that is, it represents a trusted single hop multipoint transmission (v_i,R). In an error-prone channel, a successful trusted single hop multipoint transmission may entail several retransmissions (due to errors or collisions in the forward or reverse direction), until all nodes in R correctly receive the information and v_i correctly receives all the acknowledgements. Note also that some nodes (even in set R) might receive and acknowledge more than once correctly, and this cannot be avoided because it is inherent to the broadcast advantage of the medium.

The virtual edge (v_i,v_R) in G' is assigned the weight of the single hop multipoint transmission w_{iR} defined by Eq. (1). Then zero weight edges are used to connect virtual node v_R to all receiver nodes in R. The trusted single hop multipoint transmission (v_i,R) in the original graph G is equivalent to the trusted point-to-point transmission (v_i,v_R) in the expanded graph G'. The edges between v_i and each node in R in G are maintained in G' to represent the unicast point-to-point transmissions from v_i to each of these nodes. The pseudo code presented in Algorithm 1 summarizes the above process for constructing the auxiliary graph G'.

```
G'(V',E') =G(V,E)
FOR all v_i in V  // For all network nodes
define V_i as the set of receivers with the range of v_i's transmission
    FOR all R ⊆ V_i with |R|≥2  //For all subsets of V_i with more than two elements
        Create virtual node v_R
        Calculate w_iR using eq. (1)
        Create virtual link (v_i, v_R) with cost w_iR
        FOR all v_j in R
            Create virtual link (v_R, v_j) with zero cost
        End FOR
        add to V' and E' the new virtual node v_R and its virtual links
    END FOR
END FOR
```

Algorithm 1: The network graph expansion.

In particular, the auxiliary expanded graph $G' = (V',E')$ is obtained from the original network graph G as follows. In the beginning, we set $G' = G$. For each node $v_i \in V$ and for each subset $R \subseteq V_i$ with more than two nodes ($|R| \geq 2$), we create a *virtual node* v_R, which is inserted in the expanded graph G' and is connected in the following way:

3.2 Graph Transformation and WUT Property

The previously described graph transformation captures the WBA property by introducing all possible single hop multipoint transmissions. A key attribute of this graph transformation is that on the expanded graph G' we only have to deal with point-to-point transmissions (as in wireline networks) as opposed to the single hop multipoint transmissions present in the wireless network G under the WBA property. This transformation can be useful in a variety of optimization problems in wireless networks, through different definitions of the function f in Eq (1). In what follows we will use a definition that captures the WUT property and aims at minimizing the ETX required to perform a multicast; other definitions of f, corresponding to different optimization objectives, will be discussed in Sect. 3.4.

To model the WUT property, we have assumed that a trusted transmission over edge (v_i, v_j) is a Bernoulli trial with probability of success p_{ij}. If some of the recipient nodes fail to receive the data or the acknowledgement (ACK) is lost, the sender will resend the packet until the ACK is received successfully. Thus, the success probability p_{ij} corresponds to both the forward and the reverse (ACK) transmission being successful. The number X_{ij} of required transmissions before a successful trusted transmission has been completed over (v_i,v_j) follows a geometric distribution, and its expected value defines the weight of edge (v_i,v_j):

$$w_{ij} = \mathrm{ETX}\left(v_i, v_j\right) = E\left[X_{ij}\right] = 1/p_{ij}.$$

The ETX metric has been widely adopted in wireless networks [2], as it is related to the actual achievable throughput over an error-prone link. ETX can be calculated by collecting statistics on the control/data packet transmissions, which can be combined in a weighted fashion to obtain the probability of a successful trusted transmission. A periodic transmission of special purpose control packets can also be used to estimate it.

As already mentioned, a packet transmission from node v_i is heard by a set V_i of possible receivers. We assume that the probabilities of successful reception of a packet and its corresponding acknowledgement are independent for the different receivers [2]. Consider now the case where we want to perform a trusted (intentional) single hop multipoint transmission from node v_i to a set of receivers $R \subseteq V_i$. For each node $v_j \in V_i$ we denote by $f_{ij} = 1 - p_{ij}$ the complementary failure probability. The ETX for a successful trusted single hop multipoint transmission (that is, for the successful reception of a packet and its acknowledgement by all nodes in R) is denoted by $w_{iR} = \mathrm{EMT}(v_i,R)$, where EMT stands for EMT. An extension of such metric called expected multicast transmission time (EMTT) that considers also sender's adaptive bit-rate transmissions is presented in [10]. According to [8], we can calculate $\mathrm{EMT}(v_i,R)$ as follows:

$$w_{iR} = \mathrm{EMT}(v_i, R) = \sum_{c=1}^{|R|} (-1)^{c-1} \cdot \sum_{S:\, S \subseteq R \text{ and } |S|=c} \frac{1}{1 - \prod_{j \in S} f_{ij}}, \tag{2}$$

where we sum over all numbers c from 1 to the cardinality of set R, and over all subsets S of R, that have cardinality c. Note that in case of $f_{ij} = 0$ for some j, that is for failure probability $p_{ij} = 1$, we need an infinite number of retransmissions w_{iR} to reach all nodes of set R.

From the above, we have

$$\mathrm{EMT}(v_i, R) \geq \mathrm{ETX}\left(v_i, v_j\right), \text{ for all, and also } \mathrm{EMT}(v_i, R) \leq \sum_{j: v_j \in R} \mathrm{ETX}\left(v_i, v_j\right)$$

Note that if the set R consists of a single receiver, e.g. $R = \{v_j\}$, then $\mathrm{EMT}(v_i,R) = \mathrm{ETX}(v_i,v_j)$. Thus, the definition of the EMT metric for single hop multipoint transmissions includes simple point-to-point transmissions as a subcase, and both point-to-point and multipoint transmissions can be treated in a unified way.

Equation (2) enables us to calculate the weight of a single hop multipoint transmission from the weights of the point-to-point links that comprise it. Thus, Eq. (2) is a special case of Eq. (1) and can be used in the graph transformation process described earlier (Algorithm 1) to obtain the weights of the auxiliary expanded graph G'.

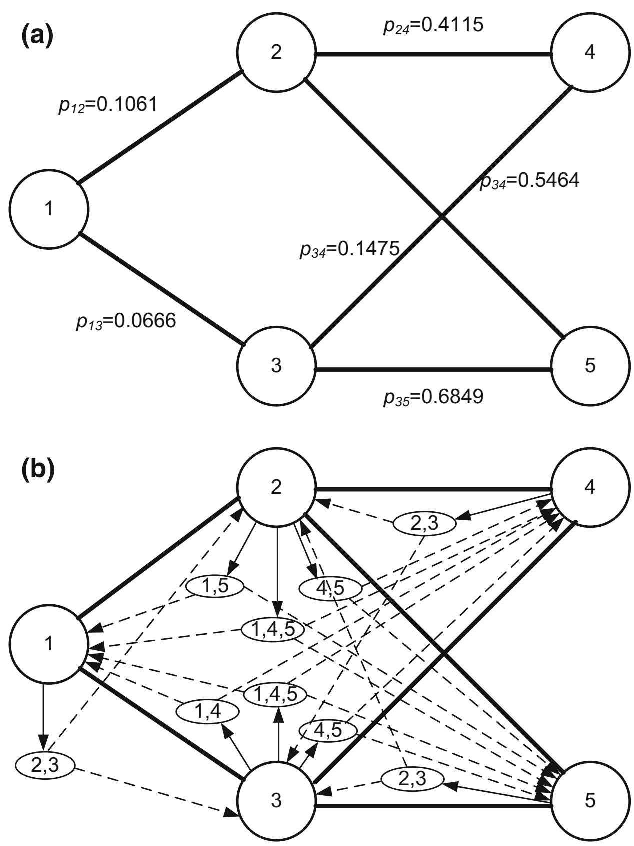

Fig. 1 An example of the network graph expansion step. The original graph presented in (**a**) is expanded to the graph shown in (**b**). The virtual nodes in the expanded graph are named according to the nodes they lead to with zero weight edges (*dashed lines*)

Figure 1 presents an example of the expansion of a graph that utilizes the ETX metric. The original graph is depicted in Fig. 1a. In the expanded graph, shown in Fig. 1b, the edge weights are omitted for the purpose of clarity. The bold edges connecting the virtual nodes with original nodes signify zero weights. Each virtual node in the expanded graph represents a trusted single hop multi-point transmission (v_i,R) and is labeled according to the set of receiver nodes R. Note that the proposed graph transformation process creates all virtual nodes. Then it is the role of the algorithm, such as the ones presented in Sect. 4 to select the ones to use.

3.3 Complexity of the Graph Transformation

An issue that needs to be discussed is the complexity of the transformation used to obtain the expanded graph G' from the original graph G. Recall that node v_j is said to be within the transmission range of v_i if the probability p_{ij} of v_j correctly capturing and acknowledging a transmission from node v_i is above a certain threshold p_t. Let $d_i = |V_i|$ be the node degree of v_i in G. The number of virtual nodes that are added in G' corresponding to node v_i is equal to $2^{d_i} - d_i - 1$. This is because a virtual node is added for each subset of nodes of V_i except for the empty and single element subsets. The number of virtual edges can also be calculated in a similar manner. We see that the size of the expanded graph depends, in general, exponentially on the node degrees of the original graph and thus comprises a non-polynomial transformation. Note, however, that the important in practice case of regular network topologies like line-arrays, 2-dim grids and other topologies with constant or logarithmic node degree do not exhibit such an issue. Also, a virtual node (v_i,R) corresponding to receiving set R may not have to be created if the cost of the links leading to it have weight w_{iR} that is above a certain threshold. For example, it would be natural to use the threshold p_t on p_{ij}, used in the definition of the network graph G, and remove nodes from the expanded graph G' corresponding to subsets R for which $w_{iR} > 1/p_t$ for all v_i. Note that by changing the threshold p_t that defines the nodes considered to be within reach, we can control the node degrees and thus the complexity of the algorithm. This will be explored further in Sect. 5.1, where we will give a truncated graph transformation of polynomial complexity for general network topologies.

3.4 Applications of the Graph Transformation to Other Optimization Problems

Combining the above transformation with appropriate definitions of function f in Eq. (1), a number of different optimization problems in wireless networks can be formulated and solved. For example, consider the minimum energy multicasting problem under the popular "sphere" model, where all nodes within a specific distance ρ_i from transmitter v_i correctly decode a packet, while all nodes at distance larger than ρ_i do not correctly decode it. We assign weight w_{ij} to the edge (v_i,v_j) equal to the power required for node v_i to transmit to node v_j (that is, the power required to make the transmission distance ρ_i equal to the distance between v_i and v_j). Since, in the sphere model, all nodes that require less transmission power to be used by v_i are able to overhear the transmission, the energy required to transmit successfully to all the nodes in set R is given by

$$w_{iR} = f\left(w_{ij} | v_j \in R\right) = \max_{v_j \in R} \left(w_{ij}\right)$$

Thus, the proposed graph transformation with the preceding definition of function f in Eq. (1) can be used to solve the minimum energy multicasting problem in a wireless multihop network under the "sphere" model.

Taking a different direction, consider co-operative communication [22], where each transmission implicates, in addition to the transmitter, a set of helper nodes that cooperate with it to either increase the range up to which the packet is correctly received or decrease the total power expended in transmitting the packet. Cooperative communication can be viewed as the dual ("gathering") of the multicasting problem considered here, since in this case the multicast set R corresponds to the set of transmitters that cooperate for a multi-point-to-point transmission. Using our notation, the power required for cooperative transmission [22] is

$$w_{Ri} = f\left(w_{ji} | v_j \in R\right) = 1 \Bigg/ \sum_{v_j \in R} \left(1/w_{ji}\right)$$

With this definition of function f in Eq. (1) the proposed graph transformation together with the algorithms to be described in the next section can be used to solve the minimum energy routing problem under the cooperative transmission paradigm.

4 Minimum Expected Multicast Transmissions Problem

In this section we formulate the MinEMT problem and present an optimal algorithm to solve it. As discussed previously, the minimum energy multicast under the sphere model problem and the minimum energy routing under the cooperative transmissions model problem could be solved using an algorithm similar to the MinEMT algorithm to be presented shortly (the details for these two problems are omitted).

4.1 MinEMT Problem Formulation

We are given the network graph $G = (V,E)$ and the success probabilities p_{ij} for each $(v_i, v_j) \in E$. The weights of the edges are defined as $w_{ij} = \mathrm{ETX}(v_i,v_j) = 1/p_{ij}$. We are also given a source node v_1 and a multicast group $M \subseteq V$ whose nodes have to receive a copy of the packet located at node v_1. The objective is to find the multicast schedule that minimizes the ETX that have to take place in order to carry a copy of the packet to all nodes in the multicast group M.

A multicast schedule is defined as a sequence of single hop multipoint transmissions $T_k = (v_k,R_k)$, each consisting of the transmitter v_k and the set of neighbors R_k that have to (reliably) receive the transmission. Thus, at each step T_k the transmitting node v_k keeps retransmitting until all nodes in R_k successfully receive and acknowledge the packet. The expected number of transmissions $\mathrm{EMT}(v_k,R_k)$ required to accomplish this is given by Eq. (2). Note that T_k may be a simple point-to-point transmission, as this as a special case of single hop multipoint transmission. A feasible multicast schedule is obtained by having the transmitter v_k at step k be among the nodes that have received the packet up to step $k - 1$. More formally, we require

$$v_k \in \cup_{z=1:k-1} R_z.$$

for a feasible schedule. The starting node is the source node v_1 of the packet. The schedule finishes at step K when all nodes in the multicast group M have received the transmission, that is, when

$$M \subseteq \cup_{z=1:K} R_z.$$

The variable K corresponds to the number of different transmitters used in the schedule. The objective of the MinEMT problem is to find the schedule that minimizes the ETX

$$\sum_{k=1:K} \mathrm{EMT}(v_k, R_k)$$

required to get a copy of the packet to all nodes in the multicast group M.

In our problem definition, the number K of transmitting nodes used in the schedule is not an optimization criterion; instead, we are interested in the ETX performed by these nodes, which is captured by the EMT metric (importantly, this is also a measure of the load that the multicast will cause on the network). Following this schedule, irrespectively of the success or failure of the transmissions, would give us the lowest a priori average number of transmissions. Note that during the multicast and based on the actual successes and failures of the transmissions, other schedules could be calculated that could improve performance. This would however happen at the cost of rerunning the algorithm and coordinating the transmissions at each step. An additional disadvantage of such algorithms, which rely on opportunistic rather than trusted transmissions, would be that the packets in a multicast session would not follow a deterministic path and could be received out of order at the receivers, something that is undesirable for video and other (e.g. topology updates) data. In the future we plan to examine such dynamic rescheduling policies and examine in more detail their advantages and disadvantages.

Note that calculating the minimum Steiner tree in the network graph $G = (V,E)$ does not yield an optimal solution, as would be the case in a wireline network with point-to-point links. This is because the WBA and the consequent gains that can be obtained by multipoint transmissions are not exploited when solving the Steiner tree problem in the original network graph.

4.2 Multicast Optimal Algorithm on the Expanded Graph

We now describe an optimal algorithm to solve the MinEMT problem. The algorithm's operation is divided into two phases. In the first phase the network graph is expanded so as to capture the WBA and the WUT properties. Then in the second phase the minimum Steiner tree on the expanded graph is found to obtain the MinEMT schedule.

4.2.1 *Expanding the Graph to Capture WBA and WUT*

In the first phase of the algorithm we use the graph transformation algorithm described in Sect. 3 to obtain the expanded graph G', using as link weights the ETX metric and Eq. (2) to calculate the EMT of a trusted single hop multipoint transmission.

4.2.2 *Minimum Expected Transmission Multicast*

In the second phase of the algorithm, the minimum Steiner tree is calculated in the auxiliary graph G'. The start node is source v_1, the destination nodes are the nodes in the multicast group M, and the Steiner nodes are the rest of the nodes in G' (the nodes of the original network along with the virtual nodes inserted in the first phase of the algorithm). Since the definition of the expanded graph captures the WBA and WUT properties, we only have to consider point-to-point (unicast) transmissions in the expanded graph G', and, thus, the minimum Steiner tree of G' yields the optimal MinEMT solution.

To optimally solve the minimum Steiner tree problem we can use a number of algorithms, ranging from integer linear programming formulations (ILPs) [18] to dynamic programming methods, such as the widely used dynamic programming algorithm of Dreyfus–Wagner [19]. In this study we used the ILP formulation of [20].

After obtaining the minimum Steiner tree in the expanded graph G' we translate the solution to the original graph G to obtain the optimal MinEMT schedule. A transmission over a virtual edge in G' is translated to the corresponding single hop multipoint transmission in G, while a simple single hop transmission (a transmission over an original and not a virtual edge) in G' remains the same in G. A single hop multipoint transmission requires the source to repeatedly transmit the packet until all nodes described in that transmission receive and acknowledge it correctly.

4.2.3 Complexity Issues

Finding the minimum Expected Transmissions Multicast without taking into account the WBA is an NP-hard problem [5]. Moreover, considering the WBA we have to consider all possible single hop multipoint transmission combinations, which complicates even more the problem (a task that is carried out in the generally non-polynomial graph transformation presented in Sect. 3). Since finding the optimal solution is difficult, we propose in the following section heuristic algorithms that can provide practical solutions to the problem.

5 Heuristic Algorithms

In this section we will propose a number of heuristic algorithms to find the multicast schedule with the minimum ETX (MinEMT problem) in a wireless multihop network. We will start by presenting a truncated graph transformation of the initial graph so as to reduce the complexity of obtaining the expanded graph and then we will also give two heuristic algorithms to find the multicast schedule. Both heuristics for the multicast schedule start from source node v_1 and build the multicast transmission schedule in a greedy fashion, on a node-by-node basis. At each step, the transmission with the highest gain is added to the multicast schedule. The gain used is a metric of how closer to the multicast group we reach with the transmission and will be formally defined in the following paragraphs. The difference between the two algorithms to be described is that the first heuristic operates on the expanded graph, while the second on the original graph.

5.1 Truncated Graph Transformation

Since the complete graph transformation described in Sect. 3 depends exponentially on the node degrees, it is not practical for networks whose connectivity degree grows faster than logarithmically with the number of nodes. To reduce the complexity, we developed and evaluated a heuristic variation to this optimal transformation. This *truncated graph transformation* uses a connectivity threshold D_t and considers for each node only up to D_t outgoing links, and in particular the D_t links with the highest success probability, or equivalently the lowest EMT values. So for node v_i, the set V_i of receivers considered in the graph expansion (Algorithm 1) has cardinality bounded by D_t. The complexity of the obtained transformation can be controlled through parameter D_t. The heuristic algorithm is polynomial if D_t is chosen to be at most logarithmic in the number of nodes. Note that this heuristic truncated graph transformation does not generate a disconnected graph, if the initial graph is connected. All links that existed in the original graph G are included in the expanded graph G', while the threshold D_t prunes links only in the graph expansion process.

5.2 Heuristic Multicast Schedule on the Expanded Graph (HME)

This algorithm works on the expanded graph G' calculated using the full non-polynomial transformation of Sect. 3 or the truncated graph transformation of Sect. 5.1.

On the expanded graph G' we calculate the all-pairs shortest paths using, for example, the Floyd–Warshall algorithm [21] and use this information to construct for every node v_i a distance vector D_i with size $|M|$, with the shortest distances from v_i towards every node in the multicast group M. The algorithm works as follows. At every step, a set C is maintained containing the nodes already covered by the algorithm. When $M \subseteq C$, the algorithm terminates. Initially, the set C contains only the source node v_1. We also keep a distance vector D_C that includes the shortest distances from the covered nodes in C to the nodes in the multicast group M. Initially $D_C = D_1$. At every step, a node contained in the covered set C is selected for transmitting over one of its outgoing edges (either original or virtual edges). For all nodes v_i in C, and for all outgoing edges $(v_i, v_j) \in E'$ we first calculate the gain vector $Q_{ij} = D_j - D_C$, defined as the difference of the distance vector of the new node v_j minus the distance vector of the covered set C up to that step. Then we calculate a scalar total gain q_{ij} by considering only the nodes belonging to the multicast set that are not yet covered in $C (m \notin C)$ and are closer to v_j, that is, the nodes for which $Q_{ij}(m) > 0$. More formally,

$$q_{ij} = \left(\sum_{m \notin C,\, Q_{ij}(m) > 0} Q_{ij}(m) \right) - w_{ij}, \qquad (3)$$

where w_{ij} is the weight of edge (v_i,v_j) and $Q_{ij}(m)$ corresponds to the mth element (multicast node) of the gain vector Q_{ij}.

Having obtained the scalar total gains for all outgoing edges of nodes in C, we select as the transmission for the current step the outgoing edge that yields the highest gain. Since we are working on the expanded graph G', this can be point-to-point or a single hop multipoint transmission. Based on this, we update accordingly the covered set C and the distance vector D_C. If all the nodes in the multicast set have been covered ($M \subseteq C$) the algorithm terminates. Else, we proceed to next step. The operation of the heuristic algorithm is summarized in the pseudo code presented in Algorithm 2. For the remaining of this paper we will refer to this algorithm as Heuristic Multicast Schedule on the Expanded graph (HME).

```
Find for all vi in V' the shortest distance between vi and all nodes in M and store in vector Di
Initialize the Covered set C=v1, and the vector DC=Dv1
Initialize the step index k=1
WHILE M ⊈ C
    MaxGain = 0
    FOR vi in C
        FOR vj for which (vi,vj) in E'
            Qij=DC - Dj
            Calculate gij with Eq. (2)
                IF qij > MaxGain
                    (vi*,vj*) = (vi,vj)
                    MaxGain = qij
                ENDIF
        ENDFOR
    ENDFOR
    IF vj* is an original node
        R* = vj*
    ELSE // vj* is a virtual node
        Find original node receivers R*: vm ∈ R* iff ( vj*,vm) in E'
    ENDIF
    C = C U R*
    Update DC
    Define step k of the schedule as {vk, Rk}={vi*, R*}
    k=k+1
ENDWHILE
```

Algorithm 2: The heuristic algorithm on the expanded graph.

The HME algorithm described above is a greedy heuristic algorithm that at each step expands the covered set of nodes with the transmission that takes it *closer* to the uncovered nodes in the multicast set. It uses the expanded graph that captures both the WBA and the WUT, so that the algorithm only has to consider point-to-point transmissions in the expanded graph. Note that as the algorithm is executed, if one node is used for a transmission at a certain point, this node will not be used again later for another transmission. That is, at each point we have only to consider the "front" of the covered set, a feature that is not formally described in the above description but can be used to reduce the running time of the algorithm. Still, the algorithm has to check all the outgoing edges of a potentially large number of nodes. Since the algorithm is executed on the expanded graph and its complexity is polynomial to the number of edges in that, which depend exponentially on the node degrees of the original network graph, the algorithm can take exponential time in the size of the original input to find the solution. However, the complexity of the algorithm can be reduced to polynomial by employing the truncated graph transformation of Sect. 5.1.

Find for all v_i in V' the shortest distance between v_i and all nodes in M and store in vector D_i

5.3 Heuristic Multicast Schedule on the Original Graph (HMO)

The algorithm presented in the preceding section operates on the expanded graph G'. As discussed above, if the expanded graph G' is created using the full transformation it can become impractical for some types of networks. Thus, in this section we present an alternative algorithm, called Heuristic Multicast on the Original graph (HMO), which operates on the original graph and runs in polynomial time.

Since HMO operates on the original graph G, the broadcast nature of the wireless medium (WBA) has to be incorporated explicitly in its operation. The algorithm works as follows. We again maintain a set C of covered nodes and a distance vector D_C from the covered nodes towards the multicast group. For each node v_i in C, we consider its outgoing edges (only original edges this time since we are working on G) and keep those that have positive gain, as defined in Eq (2). These outgoing edges form the receiver nodes R of the single hop multipoint transmission for node v_i for which we calculate the corresponding expected number of multicast transmissions EMT(v_i,R). We select the node and the single hop multipoint transmission with the smallest EMT and update the set C with the selected receivers and the distance vector D_C. The algorithm continues until all the nodes in the multicast set M have been covered.

The HMO algorithm searches at each step all covered nodes. For each such node, it checks the edges/next hops that would lead closer to the multicast group, forming a single hop multipoint transmission. It then selects the best candidate single hop transmission and continues to the next step. In the worst case, O(N) steps will be performed and in each step O(N^2) single multihop connections will be examined. Compared to the HME heuristic that runs on the expanded graph and searches all possible single hop multipoint transmissions (that depend exponentially on the node degree of the original network graph), the HMO heuristic checks a limited and polynomial number of single hop multipoint transmissions. We expect this to deteriorate performance, but the algorithm will be polynomial in all cases.

In Fig. 2 we illustrate an example of finding the minimum expected transmissions for a multicast session. In particular we graph the multicast trees obtained by the Shortest Path Tree (SPT) algorithm, the heuristic proposed by Zhao et al. in [8], which we will refer to as Zhao Heuristic algorithm on the original graph (ZHO), and our optimal MinEMT algorithm that runs on the expanded graph. The trees (schedules) obtained are compared in terms of the ETX (EMT) required for multicasting a packet from source node 1 to the nodes 4 and 5 comprising the multicast group. The optimal MinEMT algorithm, by exploiting fully the WBA and WUT properties manages to save one transmission, compared to the schedule produced by the heuristic ZHO of Zhao. Interestingly, both of our heuristic algorithms, HME and HMO, in this example, succeed in obtaining the optimal (minimum ETX) multicast tree.

5.4 Extensions

The proposed algorithms can be extended in several directions. It would be interesting to consider concurrent transmission of packets located at different nodes and propose multicast schedules for minimizing not only the total ETX required to complete all the multicasts (the MinEMT algorithm applied to each source packet separately would achieve that, assuming collisions are resolved at the MAC layer), but possibly also the expected time required to complete all transmissions (the so called makespan). To do so, one would have to take into account the interference radius of each transmission and calculate the corresponding collision free transmission sets. These collision free sets could be used for concurrent transmissions, reducing the time required to complete all multicasts. A second direction is to incorporate co-operative [22] and opportunistic [23] features in the proposed algorithms. Note, that since the proposed graph transformation is general, the proposed technique can be used to solve a number of other problems in wireless networks. In the end of Sect. 3 (Subsection 3.2.1) we outlined how the proposed graph transformation can be used to capture the cooperative transmission paradigm and solve the minimum energy multicasting problem.

6 Simulation Results

We conducted simulation experiments to evaluate the performance of the proposed optimal and heuristic multicast algorithms and compare it to that of other multicast algorithms proposed in the literature.

In our experiments, we randomly generated networks with a given (1) number of nodes N (2) average node degree D, and (3) average value W for the link weights (a weight represents the average number of transmissions required over a link). To create a random network, we start with the given number of nodes N and create an all-nodes-to-all-nodes edge matrix with random numbers as it entries using a uniform distribution. We keep only the edges that result in a network with the requested connectivity degree D, discarding the edges with the smallest randomly created numbers. Having obtained the topology, we assign to each edge a weight (the ETX for a successful reception) drawn according to an exponential distribution with mean equal to W. In each experiment, we also select the source node and the multicast set of nodes with a given cardinality $|M|$, which is also a parameter in the experiments. Both the source and the nodes in the multicast set are drawn from a uniform distribution.

The algorithms considered in the simulation study are the optimal MinEMT multicast algorithm that runs on the expanded graph as presented in Sect. 4.2, the heuristic HME algorithm that runs on the expanded graph and the heuristic HMO algorithm that runs on the original graph, which were proposed in Sects. 5.2 and 5.3, respectively. For the optimal MinEMT and the heuristic HME algorithms that run on the expanded graph, we have two

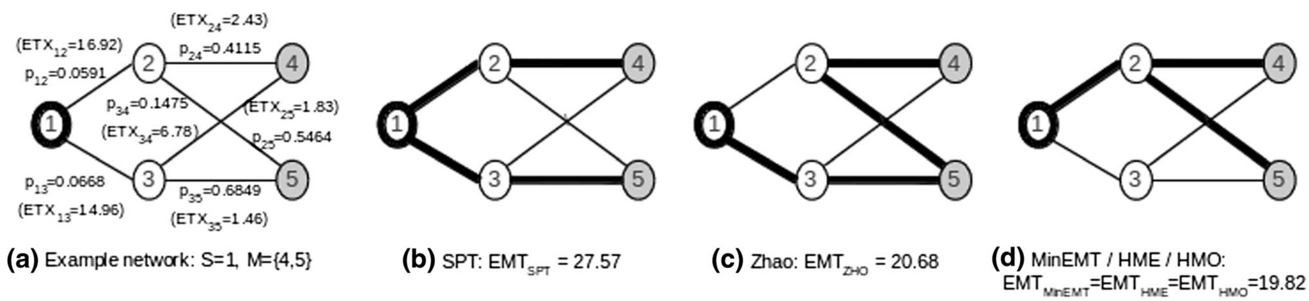

Fig. 2 **a** The example network with the probabilities of correct transmission p_{ij} and the corresponding weights $w_{ij} = \mathrm{ETX}(v_i,v_j)$. The source node is node 1 and the multicast set consists of nodes 4 and 5. The multicast tree (*with bold edges*) calculated by the **b** Shortest Path Tree (SPT) algorithm, **c** the heuristic algorithm of Zhao et al. [8], and **d** the proposed optimal/heuristic algorithms, along with the total EMT cost of each solution

options, regarding the transformation that is used to obtain the expanded graph. By default we will assume that the expanded graph is obtained using the complete transformation of Sect. 3, and refer to the corresponding algorithms simply as MinEMT and HME. In specific experiments where we utilize the truncated graph transformation of Sect. 5.1, we will refer to the algorithms as MinEMT-D_t and HME-D_t, where D_t will give the corresponding value of the connectivity threshold that is used. In our experiments, we stopped the execution of the optimal MinEMT algorithm after letting it run for 1 h. In cases where the MinEMT algorithm was not able to finish execution and track the optimal solution in this time period, we use the best solution found up to that point. For comparison purposes we also implemented a heuristic algorithm that is based on Shortest Path Tree (SPT), which routes a packet from its source to the nodes in the multicast set over their shortest paths (copying a packet at intermediate nodes when appropriate). To have a fair comparison, when a node is utilized in more than one shortest path in SPT algorithm we consider this as a single hop multipoint transmission and calculate the corresponding EMT value. We have also implemented the heuristic algorithm of Zhao et al. [8], which runs on the original graph and employs the EMT metric to obtain a multicast tree in a greedy fashion. We refer to Zhao′s Heuristic on the Original graph as ZHO algorithm.

The metrics used in comparing the algorithms are the average number of transmissions required to complete a packet multicast and the average execution time. These performance metrics are depicted under various scenarios, each focusing on the effect of a different parameter, such as the number N of nodes in the network, the average node degree D, the average number W of expected transmissions over a point-to-point link, and the size $|M|$ of the multicast group. Unless stated otherwise, the default parameters used in our experiments are the following: $N = 50$ nodes, $D = 3$, $W = 1.5$ transmissions, $|M| = 10$ nodes. Experiments were run on a laptop with Intel i5-2520 CPU at 2.5 GHz and 4 GB of memory. The algorithms were developed in Matlab and CPLEX 11.2 [25] was used to obtain the solution for the ILP Steiner tree formulation implemented for the MinEMT optimal algorithm. In each experiment we generate the network, the source and the multicast set and we run the chosen algorithm to calculate the average number of transmissions required, without actually performing transmissions. For each reported experiment we created ten random problem instances, run the chosen algorithm to calculate the multicast schedule, and averaged the results over these ten experiments.

Figure 3 presents the average number of transmissions and the average running time of the examined algorithms as the number of nodes N ranges from 25 up to 100 nodes with a step of 25 nodes. As expected, the average number of transmissions and the average running times of the algorithms increases with the number of nodes. The Shortest Path Tree (SPT) algorithm exhibits the worst performance, resulting in the largest average number of transmissions, which was expected given the unsophisticated nature of the algorithm that does not account for the WBA and the WUT properties. Since SPT is quite simple, its execution time is considerably smaller than that of the other algorithms examined. The optimal MinEMT algorithm was able to track optimal solutions for all the experiments with 25 nodes and more than half of the experiments with 50 nodes, exhibiting the best performance in both cases. Even for higher values of N, where we are not sure if it has found the optimal solution in many instances, MinEMT still exhibits the best performance. However, as can be seen in Fig. 3b, the average running time of MinEMT is quite high and it reaches the limit of 1 h per experiment for $N = 100$ nodes. Zhao’s heuristic algorithm (ZHO) achieves generally good average number of transmissions performance with relatively low running times. The heuristic HME algorithm that runs on the expanded graph performs significantly better than ZHO. The performance of HME comes quite

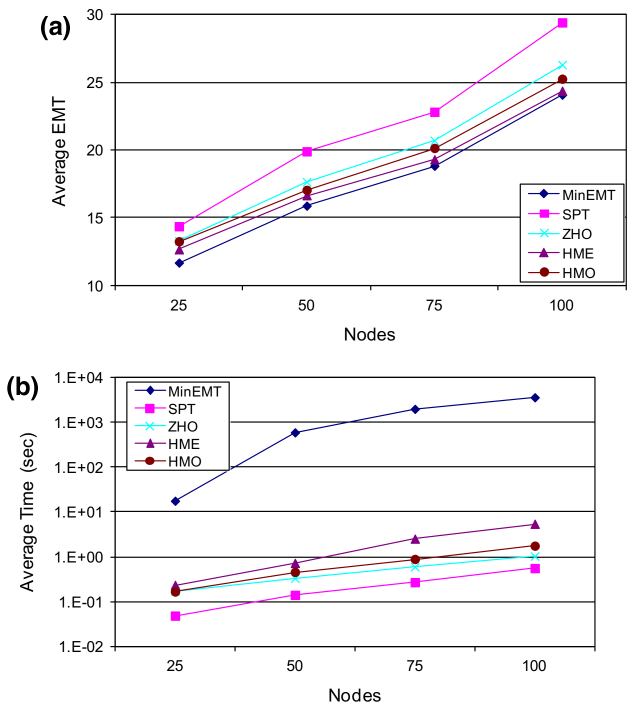

Fig. 3 **a** Average expected number of transmissions and **b** average running time as a function of the number of nodes *N* for the optimal MinEMT algorithm, the HME and HMO heuristics, and the SPT and ZHO algorithms

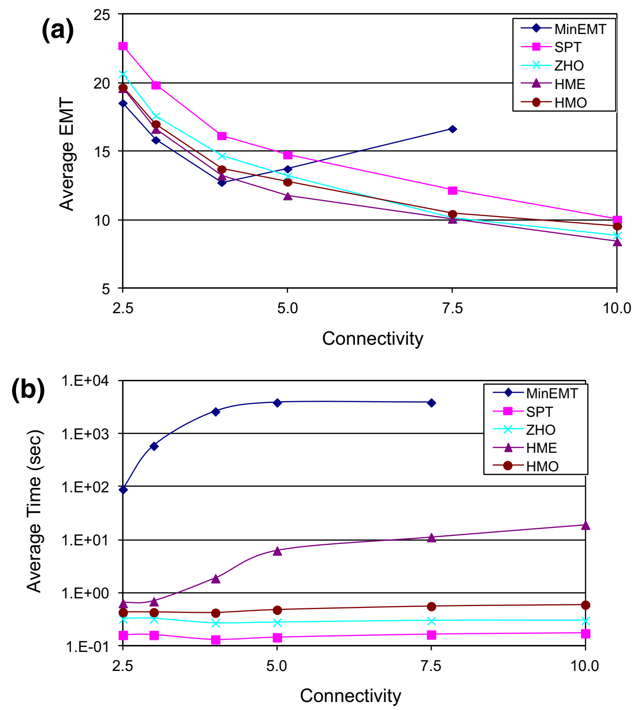

Fig. 4 **a** Average expected number of transmissions and **b** average running time as a function of the average connectivity degree *D*

close to that of the optimal MinEMT algorithm, especially as *N* increases and MinEMT is no longer able to track the optimal solutions. The running time of HME is the highest among the heuristic algorithms, but it scales well with the number of nodes *N*. The HMO algorithm, which runs on the original graph, exhibits performance that is slightly worse than that of HME and slightly better than that of ZHO, while its average running time also lies in between that of these two algorithms.

Figure 4 presents the results obtained for different values of the average node degree *D,* ranging from 2.5 up to 10. The average number of transmissions decreases as the node degree increases, for two main reasons: (1) the paths that lead to nodes in the multicast set become shorter, and (2) single hop multipoint transmissions become more efficient. Again, the SPT algorithm exhibits the worst average number of transmissions but the best average running time. The MinEMT algorithm has the best performance for low values of the average node degree *D*, but as *D* increases it is no longer able to track optimal solutions within the 1 h limit, and its performance deteriorates. This is because the optimal algorithm runs on the expanded graph that utilizes virtual nodes to capture all the possible single hop multipoint transmissions. As *D* increases the expanded graph becomes very large and the performance of the MinEMT algorithm deteriorates vastly. For $D \geq 5$ the performance of the MinEMT algorithm turns out to be worse than that of the other algorithms. Again Zhao's heuristic algorithm (ZHO) has good average number of transmissions and average running time performance. The HME algorithm that runs on the expanded graph outperforms both ZHO and HMO as well as MinEMT for $D \geq 5$, but its running time increases as *D* increases. This is because it runs on the expanded graph, whose size increases non-polynomially with *D*. This bad scalability of HME with respect to the node degree *D* was expected, and this is reason we introduced the HMO algorithm that runs on the original graph, as well as the versions of MinEMT and HME that run on the truncated graph, and will be considered in subsequent experiments. The performance of HMO is quite good, though slightly worse than that of HME, and it is achieved with polynomial running time. Note that the average running times of the SPT, ZHO and HMO algorithms increase only slowly with the node degree *D*.

To further elaborate on the effect of the node degree, we focus in the results reported in Fig. 5 on the MinEMT and the HME algorithms that run on the expanded graph. In particular, we graph the performance of MinEMT and HME when using the complete and the truncated graph transformation, and for different truncation values D_t equal to 5 and 7, (referred to as MinEMT-5, MinEMT-7, and HME-5, HME-7, respectively). For comparison purposes we also graph the performance for the case where the full non-polynomial graph transformation is used, and also the performance of the HMO heuristic that operates on the original graph. We see that the truncated graph transformation improves vastly the solutions that can be obtained by MinEMT in reasonable (1 h) time, since MinEMT-5, MinEMT-7 are able to track better solutions for networks with high node degree, compared to the case where MinEMT is used on the complete expanded graph. With respect to the heuristic HMO algorithm, we note only small differences in the performance, indicating that it is not significantly affected when the truncated graph transformation is used. On the other hand, we can see that the average running times of MinEMT-5 and MinEMT-7 are reduced when a smaller node degree threshold D_t is used. This indicates that by using the truncated graph transformation we can control the running time of the MinEMT algorithm while maintaining most of its performance benefits. Finally, by comparing HMO that runs on the original graph with HME-5 and HME-7 that run on the truncated expanded graph, we can see that the HME heuristics that run on the expanded graph have slightly better performance than the HMO algorithm, at the cost of slightly higher average running times. The differences between these heuristic approaches are not substantial, so both approaches can be used to provide practical solutions to the multicast problem.

Figure 6 presents the results obtained for different values of the average link weights W, and in particular for W ranging from 1 up to 5. Recall that the weight $w_{ij} = 1/p_{ij}$ of a link (v_i,v_j) corresponds to the ETX required for a trusted v_i to v_j point-to-point transmission, and thus increasing values of W correspond to less reliable links. The conclusions regarding the performance of the examined algorithms are as previously described, with the SPT algorithm exhibiting the worst performance (average total number of transmissions required to complete a multicast) but the best running times, and the MinEMT algorithm exhibiting the best performance but slow running times. In particular, the MinEMT algorithm was able to track optimal solutions for the vast majority of instances in these experiments, as can be also verified by its average running time, which was significantly lower than the 1 h time limit we have set. The HME exhibits the best performance among the heuristics. The second proposed heuristic (HMO), has better performance than Zhao's algorithm (ZHO), for low values of W, but the latter becomes better for high values of W. The average running times of all the algorithms are not significantly affected when W varies.

Finally, Fig. 7 presents the results obtained for varying values of the cardinality $|M|$ of the multicast group, and in particular for values of $|M|$ ranging from 5 up to 30 nodes with a step of five nodes. The conclusions drawn from this set of experiments are qualitatively similar to those presented in the previous graphs. The optimal MinEMT

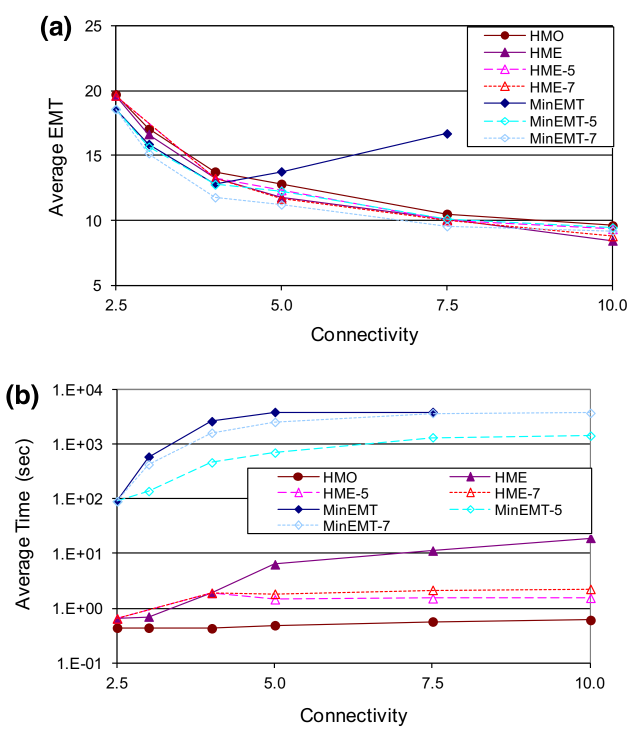

Fig. 5 Evaluation of the performance of heuristic graph transformation: **a** Average expected number of transmissions and **b** average running time as a function of the average connectivity degree D for different values of the connectivity degree threshold D_t

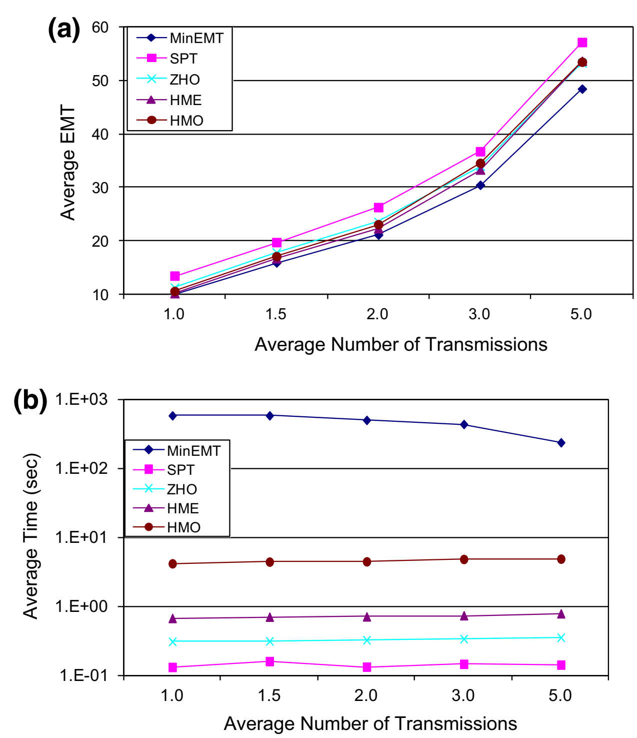

Fig. 6 **a** Average expected number of transmissions and **b** average running time as a function of the average number of transmissions for unicast communication W

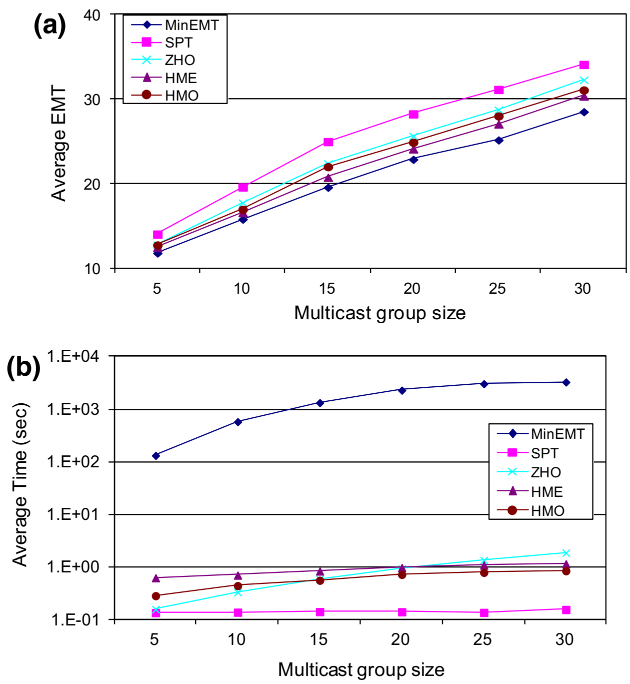

Fig. 7 **a** Average expected number of transmissions and **b** average running time as a function of the multicast group size |*M*|

algorithm exhibits the best performance and the highest running time, while HME seems to be the best heuristic algorithm. The running time of ZHO is significantly affected by the multicast group size, probably because for every member of the multicast group, ZHO examines all candidate paths to the already calculated multicast tree. As the cardinality of the multicast group increases, this process becomes computational demanding and its running time increases. The average running times of the HME and HMO heuristics and of the SPT algorithm are only slightly affected by the multicast group cardinality |*M*|.

7 Conclusions

We considered the multicasting problem in wireless multihop networks taking into account both the WBA and the WUT characteristic of the medium. We proposed a graph transformation that captures the WBA property, resulting in an expanded auxiliary graph where we have to consider only simple point-to-point transmissions. The proposed transformation is general and can be used to solve other optimization problems in wireless networks under the WBA property, a number of which were outlined in this paper. We used an appropriate weight function to capture the WUT characteristic on the expanded graph and we formulated the MinEMT problem as the minimum Steiner tree problem over the expanded graph. To address the non-polynomial complexity of the optimal algorithm, we also proposed a number of heuristic algorithms. Heuristics include a truncated graph transformation and also algorithms to build the multicast transmission schedule based on the expanded graph, the truncated expanded graph, or the original network graph. Simulation results showed that the proposed heuristics exhibit performance that is close to that of the optimal algorithm, at least for the instances for which we were able to track optimal solutions, while also outperforming other heuristic algorithms proposed in the literature.

References

1. J.E. Wieselthier, G.D. Nguyen, A. Ephremides, "On the construction of energy-efficient broadcast and multicast trees in wireless networks", IEEE INFOCOM, pp. 585–594, 2000.
2. D. S. J. De Couto, D. Aguayo, J. Bicket, R. Morris, "A high-throughput path metric for multi-hop wireless routing", Int'l Conf. on Mobile Computing and Networking (MobiCom), 2003.
3. C. Reis, R. Mahajan, M. Rodrig, D. Wetherall, J. Zahorjan, "Measurement-Based Models of Delivery and Interference in Static Wireless Networks", ACM SIGCOMM'06, pp. 51–62, 2006.
4. S. Roy, D. Koutsonikolas, S. Das, Y.C. Hu, "High-throughput multicast routing metrics in wireless mesh networks", IEEE International Conference Distributed Computing Systems (ICDCS), 2006.
5. R.M. Karp, "Reducibility Among Combinatorial Problems", 50 Years of Integer Programming 1958-2008, Springer Berlin, pp. 219-241, 2010.
6. B. M. Waxman, "Routing of multipoint connections", IEEE Journal on Selected Areas in Communication, 6(9):1617–1622, 1988.
7. P. M. Ruiz, A. F. Gomez-Skarmeta, "Approximating optimal multicast trees in wireless multihop networks", IEEE Symposium on Computers and Communications (ISCC), pp. 686–691, 2005.
8. X. Zhao, C. T. Chou, J. Guo, S. Jha, A. Misra, "Probabilistically reliable on-demand multicast in wireless mesh networks", Symp. on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2008.
9. A. Bilh, C. T. Chou, "An opportunistic multicast routing protocol for wireless mesh networks", IEEE Conference on Local Computer Networks (LCN), 2012.
10. X. Zhao, J. Guo, C. T. Chou, A. Misra, S. Jha, "A high-throughput routing metric for reliable multicast in multi-rate wireless mesh networks", INFOCOM, 2011.
11. B. Wang; J.H. Hou, "Multicast routing and its QoS extension: problems, algorithms, and protocols", IEEE Network, 14 (1), pp.22-36, 2000.
12. C. de Morais Cordeiro, H. Gossain, D.P. Agrawal, "Multicast over wireless mobile ad hoc networks: present and future directions", IEEE Network, vol. 17, no. 1, pp. 52–59, Jan/Feb 2003.
13. M. Cagalj, J.P. Hubaux, C. Enz, "Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues", Conf. on Mobile Computing and Networking (MobiCom '02), pp. 172–182, 2002.
14. I. Kang, R. Poovendran, "A novel power-efficient broadcast routing algorithm exploiting broadcast efficiency", IEEE Vehicular Technology Conference, Vol.5, pp.2926–2930, 2003.
15. C. Papageorgiou, P. Kokkinos, E. Varvarigos "Energy-efficient multicasting in wireless networks with fixed node transmission

power". International Conference on Wireless Communications and Mobile Computing (IWCMC), 2009.
16. D. Li, X. Jia, H. Liu, "Energy efficient broadcast routing in static ad hoc wireless networks", Journal of IEEE Transactions on Mobile Computing, Vol.3, No.2, pp.144–151, 2004.
17. U. T. Nguyen, "On multicast routing in wireless mesh networks", Journal of Computer Communications, 31(7), 2008.
18. M. Goemans, Y. S. Myung, "A Catalog of Steiner Tree Formulations", Networks, Vol. 23, No. 1, 1993.
19. S. Dreyfus, R. Wagner, "The Steiner problem in graphs", Networks, 1, 195–207, 1971.
20. J. Beasley, "An algorithm for the Steiner problem in graphs", Networks 14, 147-159, 1984.
21. R. W. Floyd, "Algorithm 97: Shortest path", Communications of the ACM, v.5 n.6, p.345, June 1962.
22. A. Khandani, J. Abounadi, E. Modiano, L. Zheng, "Cooperative Routing in Wireless Networks", Allerton Conference on Communications, Control and Computing, 2003.
23. R. Laufer, H. Dubois-Ferrière, L. Kleinrock, "Multirate Anypath Routing in Wireless Mesh Networks", INFOCOM'2009. pp. 37–45.
24. IEEE Computer Society LAN MAN Standards Committee. IEEE 802.11: Wireless LAN Medium Access Control and Physical Layer Specifications, August 1999.
25. http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/

Konstantinos (Kostas) Christodoulopoulos is a research fellow at the Department of Computer Engineering and Informatics, University of Patras, Greece. He holds a Diploma in Electrical and Computer Engineering from the National Technical University of Athens, Greece. He obtained the M.Sc. in Advance Computing from Imperial College of London, UK, in 2004, and the Ph.D. degree from the Department of Computer Engineering and Informatics, University of Patras, Greece, in 2009. He worked as an adjunct assistant professor at the same department from 2009 to 2011, and then as a research fellow at the School of Computer Science and Statistics, Trinity College Dublin and as a contractor for I.B.M. Research Ireland, from 2011 to 2013. His main research interests are in the areas of algorithms and protocols for optical and high speed networks, and networks for distributed and parallel computing systems.

Christos Papageorgiou received a Ph.D. in 2009, a M.Sc. degree in 2005 and a Diploma in 2002 from Computer Engineering and Informatics Department of University of Patras, Greece. Currently, he is employed as a senior professional services consultant at Citrix. He also served as Associated Teaching Staff of the Greek Open University and as Adjunct Assistant Professor in Computer Engineering & Informatics Department of University of Patras, Greece. His main research interests are in the field of Wireless Ad Hoc and Sensor Networks, routing, Quality of Service, Security, P2P Communication, Next Generation Networks, Multimedia communication and Open source routing.

Emmanouel (Manos) Varvarigos received a Diploma in Electrical and Computer Engineering from the National Technical University of Athens in 1988, and the M.S. and Ph.D. degrees in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology in 1990 and 1992, respectively. He has held faculty positions at the University of California, Santa Barbara (1992–1998, as an Assistant and later an Associate Professor) and Delft University of Technology, the Netherlands (1998–2000, as an Associate Professor). In 2000 he became a Professor at the department of Computer Engineering and Informatics at the University of Patras, Greece, where he heads the Communication Networks Lab. He is also the Director of the Network Technologies Sector (NTS) at the Research Academic Computer Technology Institute (RA-CTI), which through its involvement in pioneering research and development projects, has a major role in the development of network technologies and telematic services in Greece. Professor Varvarigos has served in the organizing and program committees of several international conferences, primarily in the networking area, and in national committees. He has also worked as a researcher at Bell Communications Research, and has consulted with several companies in the US and in Europe. His research activities are in the areas of protocols for high-speed networks, ad hoc networks, network services, parallel and distributed computation and grid computing.