Fast algorithm for multicast and data gathering in wireless networks

Michael Segal

Communication Systems Engineering Department, Ben-Gurion University of the Negev, Beer-Sheva 84105, Israel

Received 15 April 2007; received in revised form 5 November 2007; accepted 18 December 2007
Available online 21 February 2008
Communicated by S.E. Hambrusch

Abstract

Given a wireless network \( G = (V, E) \), we consider a maximum critical energy problem [J. Park, S. Sahni, Maximum lifetime broadcasting in wireless networks, IEEE Transactions on Computers 54 (9) (2005) 1081–1090] that has an objective of increasing the chances of doing a sequence of broadcasts. We present an optimal generalized solution algorithm running in improved optimal \( O(|V| + |E|) \) time, where \( V \) stands for a set of nodes and \( E \) stands for a set of links in the network. Our approach is applicable in an omnidirectional antenna model and can be used to solve the problem of multicasting traffic so as to maximize the lifetime of the network [A. Orda, B.-A. Yassour, Maximum-lifetime routing algorithms for networks with omnidirectional and directional antennas, in: Proc. ACM MobiHoc, 2005] and a data gathering problem [K. Kalpakis, K. Dasgupta, P. Namjoshi, Maximum lifetime data gathering and aggregation in wireless sensor networks, Computer Networks 42 (2003) 697–716; Y. Xue, Y. Cui, K. Nahrstedt, Maximizing lifetime for data aggregation in wireless sensor networks, ACM Mobile Networks and Applications 10 (6) (2005) 853–864] with an improved running time.

© 2007 Elsevier B.V. All rights reserved.

Keywords: Analysis of algorithms; Wireless network; Multicasting

1. Introduction

A wireless network consists of a set of transceivers communicating with each other by radio. It is customary to assume that the minimal transmission power required to transmit to a distance \( d \) is \( d^\alpha \), where the distance-power gradient \( \alpha \) is usually taken to be in the interval [2, 4] (see [10]). The transmission possibilities resulting from a power assignment induce a communication graph.

We address a maximum critical energy problem [11] and multicast sequence/data gathering problem [6,9,15] for an omnidirectional antenna model. The wireless network is represented by a weighted directed graph \( G = (V, E) \) with \(|V|\) nodes and \(|E|\) edges. The weight \( w(u, v) \) of the directed edge \((u, v)\) is the amount of energy required to transmit a unit message from node \( u \) to node \( v \). Generally, we assume the presence of asymmetric links, i.e., \( w(u, v) \neq w(v, u) \). In an omnidirectional setting, node \( u \) is able to transmit the same unit message to nodes \( u_1, \ldots, u_k \) using \( \max\{w(u, u_j) \mid 1 \leq j \leq k\} \) energy; while for directional antennas, \( u \) spends \( \sum_{j=1}^{k} w(u, u_j) \) energy.

In order to perform a multicast/broadcast from a given source \( s \in G \), we use a multicast/broadcast tree \( T \). We follow the generalized notation from [11]. Let \( \text{ce}(u) \) be the initial energy at node \( u \) of the network before
sending a message. Following a multicast/broadcast using a tree $T$, the residual energy, $re(u, T)$, at node $u$ is equal to $re(u, T) = ce(u) - \max\{w(u, v) \mid u$ is a child of $v$ in $T\} \geq 0$. The critical energy, $CE(G)$, after performing a multicast/broadcast is defined by

$$CE(G) = \min\{re(u, T) \mid u \in T\},$$

$$\min\{ce(v) \mid v \in G, v \notin T\}.\$$

Let $M \subseteq V$ be a set of multicast nodes in the network. A single-topology multicast routing scheme, or simply multicast routing scheme, is a pair $(T, t)$, with $T$ being a multicast tree rooted at source node $s$ that spans $M$ nodes, and $t$ is a time duration satisfying $\forall u \in V, t \cdot \max_{(u, v) \in T} w(u, v) \leq ce(u)$, i.e., the total energy required to activate multicast tree $T$ on which the node participates as a transmitter is bounded by the node’s initial energy. The value $t$ is called the lifetime of the multicast scheme.

Next, we define three related problems considered in this paper.

- **The maximum critical energy (MCE) problem** [11] asks to find, for a given wireless network $G = (V, E)$ and source node $s$, a multicast/broadcast tree $T$ maximizing $CE(G)$ (in the multicast tree version of the problem we are also given a set $M$ of multicast nodes required to receive a message from $s$).

- **In the multicast sequence (MS) problem** [9] we are given a wireless network $G = (V, E)$, source node $s$ and a set of multicast nodes $M$. The goal is to find a tree $T$ for the multicast scheme in order to maximize the lifetime, i.e., the duration time, of activating this tree $T$ (the number of multicast transmissions that we can make) is maximized and the total energy consumed by each node is bounded by its initial energy. Notice, every time we activate $T$ the residual energy of internal nodes is decreased. In some sense, both problems are quite similar with a difference that in a later problem we aim to find a tree of maximum lifetime in terms of number of transmissions rather only one.

- **In the data gathering (DG) problem** (see [6,15]), we are given a wireless network $G = (V, E)$, a subset $I \subseteq V$ of data nodes and a target node $d$. We aim to find a scheme of transmitting unit message information from data nodes in $I$ to the target node $d$ such as to maximize the network lifetime (i.e., we try to maximize the number of transmissions from $I$ nodes to target $d$). In other words, we wish to find a tree $DT$ spanning $I$ data nodes and target node $d$, maximizing the number of times we can activate this tree $DT$ starting at nodes of $I$ and convergecasting towards a target node $d$ while keeping the positive energy of every node in $DT$. We consider the case of gathering with data aggregation (see [4,6,8]) when any intermediate node of $DT$ can aggregate multiple incoming messages into a single message. For example, it can be done when the target node is supposed to compute some Boolean function (e.g., AND) on values kept at $I$ data nodes. Clearly, the solution to the data gathering problem can not be obtained immediately from solution of a multicast sequence problem in an omnidirectional setting.

For all above-mentioned problems we present optimal $O(|V| + |E|)$ runtime algorithms. Our approach to solve the problems includes a step for efficient producing potential solution values with the subsequent fast feasibility examination.

In what follows we briefly describe previous work regarding the problems considered in this paper as well as for other related problems dealing with energy efficiency and network lifetime prolongation. Next we present our approach with an algorithm for the MCE problem and finally show how it can be used to solve the rest of the problems in this paper.

**Previous work**

Park and Sahni [11] were the first to introduce the maximum critical energy problem by providing $O(|V| + |E| \log |E|)$ time solution. They considered only the case of a broadcast tree. We have to mention that the algorithm given in [11] can be easily generalized to deal with a multicast tree not affecting the running time of their algorithm. The multicast (in fact, broadcast) sequence problem was solved in $O(|E| \log |V|)$ time in [7] with a subsequent improvement by Orda and Yassour [9]. The algorithm given in [9] solves the multicast sequence problem in $O(|V| + |E| \log |V|)$ time. Another interesting combinatorial result concerning the multicast sequence problem has been obtained by Duin and Volgenant [3]. Regarding the data gathering (or data aggregation) problem, Kalpakis et al. [6] give a number of solutions which, however, lack mathematical evaluation. Xue et al. [15] also considered the problem of energy-efficient data collection in wireless networks. They [15] dealt with additional constraints such as generating rates and energy consumption with reception of messages and present an approximation algorithm which is based on the concurrent multicommodity flow approach.
Other relevant work in the area of energy efficient power assignment includes energy efficient broadcasting and multicasting in wireless networks. Given a number of wireless nodes with a source node $s$, the problem is to find a minimum power assignment for each node such that the induced communication graph contains a spanning tree rooted at $s$. This problem has been proved to be NP-Hard. In [2,5,13,14], authors presented heuristic solutions and gave some theoretical analysis. Srinivas and Modiano in [12] provided a polynomial algorithm that optimally finds $S$-closeness to immediate O-satisfying critical value of $c$. In [2,5,13,14], authors presented a polynomial time solution for solving the two edge-disjoint paths problem. They also provide a polynomial time solution for solving the 2 edge-disjoint paths problem.

2. Maximum critical energy problem

We follow the approach proposed in [11]. First we produce a list $C$ of potential candidate values for the solution and then build an oracle that checks these values for feasible solution efficiently. Given such oracle, i.e., an algorithm which for a given value $c$ checks whether $CE(G) \geq c$, the obvious approach is to sort all potential candidate values $C$ and to use the oracle in a binary search fashion over the list of sorted values. However, it would lead immediately to $O(|E| \log |E|)$ time since the cardinality of $C$ is $O(|V| + |E|)$ as shown below. In order to avoid this, we can do the binary search in an implicit way.

First, we observe that the set $C$ of potential candidate values for a possible solution

$$C = \{ce(u) - w(u, v) \mid (u, v) \in E, ce(u) \geq w(u, v)\} \cup \{ce(u) \mid u \in V\}.$$ 

The total cardinality of $C$ is obviously $O(|V| + |E|)$ and also can be computed at the same time.

Denote by mce($G$) a value from $C$ that guarantees an optimal solution. Opposite to the algorithm presented in [11], we can use a linear time selection algorithm proposed by Blum et al. [1]. The algorithm [1] finds, for a given sequence of $n$ values and an integer value $k$, $1 \leq k \leq n$, the $k$th smallest element in the sequence in $O(n)$ time. Using this algorithm, we do not need to sort all the $O(|V| + |E|)$ potential candidate values in order to perform a binary search as explained below.

Our next stage is to describe an oracle algorithm, i.e., a method which for a given a potential solution value $c$, checks in $O(|E|)$ time whether there exists a broadcast/multicast tree $T$ rooted at source vertex $s$, with $CE(G) \geq c$. We should note here that the oracle given in [11] has running time $O(|V| + |E|)$ rather than $O(|E|)$ which is critical in our analysis. We modify the depth-first search (DFS) algorithm in order to check the feasibility of value $c$. In fact, we need to find whether there exists a tree of $G$ spanning all (multicast set) nodes under the constraint that the subset of edges in $E$, $\{(u, v) \mid w(u, v) + c > ce(u)\}$ is forbidden from using during the DFS execution. This can be done in $O(|E|)$ time for all nodes by counting a number of connected components of $G$ (which should be only one) after ignoring forbidden edges. It also can be done in $O(|E|)$ time for multicast set nodes in the following way: we count the total number of multicast set nodes in the first connected component containing $s$.

Our final step is to describe how to perform an implicit binary search using a selection value algorithm and an oracle algorithm described above. We introduce the following notation. Let $G^{\geq c}$ be a subgraph of $G$ without edges $(u, v)$ satisfying $ce(u) - we(u, v) < c$, and let $G^{> c}$ be a subgraph of $G$ without edges $(u, v)$ satisfying $ce(u) - we(u, v) \leq c$. Clearly, the value $c$ is a required solution to our problem if an oracle gives a positive answer for $G^{\geq c}$, but negative answer to $G^{> c}$. Notice that if an oracle returns a positive answer for both $G^{\geq c}$ and $G^{> c}$, then obviously the solution value mce($G$) of critical energy is larger than $c$, and if an oracle returns a negative answer for $G^{> c}$, then the solution value of critical energy is smaller than $c$.

We proceed as follows. We find a median value $x_1$ in $C$. We can do it in $O(|V| + |E|)$ time using a selection algorithm of Blum et al. [1]. The two following cases can occur.

1. If mce($G$) < $x_1$, not all multicast members have been together at the same connected component containing $s$. Thus, we contract all connected components of graph $G^{\geq x_1}$ into single vertices by joining all the nodes which are connected by edges $(u, v)$ such that $w(u, v) + x_1 \geq ce(u)$ into a single node and discarding all these edges. We keep track of multicast nodes which were contracted at the same new vertex. Notice, that the obtained graph has only $O(|E|)$ edges, and, in order to calculate it we have to spend $O(|E|)$ time.

2. If mce($G$) > $x_1$, we consider a graph $G^{> x_1}$ that can be constructed in $O(|E|)$ time and observe that mce($G$) = mce($G^{> x_1}$). Moreover, graph $G^{> x_1}$ has $O(|E|)$ edges.

Of course, we can verify at $O(|E|)$ time (see above) whether mce($G$) > $x$ or mce($G$) < $x$. Thus, we proceed to the next step after spending $O(|V| + |E|)$ time for
the selection procedure and $O(|E|)$ time for the oracle step. We obtain a new graph having $O(|E|)$ edges. At the following step we spend $O(|V|+|E|)$ time for selection procedure, $O(|E|)$ time for oracle, and produce a new graph with $O(|E|)$ edges. By continuing this procedure at most $O(\log n)$ times, we reach the desired solution value in the total $O(|V|+|E|)$ time.

3. Multicast sequence and data gathering problems

In the multicast sequence problem we aim to find a multicast tree of maximum lifetime in terms of number of transmissions. Orda and Yassour [9] observed that the problem of finding such a multicast tree can be transformed to the following problem. Given a directed weighted graph $G = (V, E)$, with a source node $s$ and multicast set of nodes $M \subseteq V$. The new weight $w'(u, v)$ of each edge $(u, v)$ is equal to the maximal time during which this edge can be used before the energy of transmitting node drains, i.e., $w'(u, v) = ce(u)/w(u, v)$. Then, our problem is to find a multicast tree of $G$ spanning all $M$ nodes whose minimal weight edge is maximized.

We observe that our proposed strategy exactly fits a solution of the above-mentioned problem. Indeed, the set of all possible solution values is of cardinality $|E|$—every edge’s weight belongs to this set. The oracle algorithm is based on DFS strategy and is almost identical to one described in the previous section. For a given value $c$, the oracle checks whether there exists a tree of $G$ spanning the nodes of $M$ under the constraint that edges $(u, v)$ with $w'(u, v) > c$ are forbidden for use. Definitely, this can be verified in $O(|E|)$ time. The definition of $G^{\geq x}$ and $G^{> x}$ is slightly changed. Now, $G^{\geq x}$ is a subgraph of $G$ without edges $(u, v)$ such that $w'(u, v) < x$, and $G^{> x}$ is a subgraph of $G$ without edges $(u, v)$ such that $w'(u, v) \leq x$.

As before, we can determine in $O(|E|)$ time whether the optimal solution is equal, smaller or larger than a given value $c$. We perform an implicit binary search on the potential solution values, using oracle at each step in order to verify the feasibility of the given value and to navigate our algorithm exactly as we did in the previous section. More precisely, if an optimal solution is less than some value $x$, we contract all connected components of the graph $G^{\geq x}$ into single vertices by joining all the nodes connected by edges of weight $x$ (at least) into a single node and discarding these edges. Otherwise, we build a graph $G^{> x}$ and proceed by looking at its edges. Thus, we obtain an $O(|V|+|E|)$ runtime solution, improving previously known results by Orda and Yassour [9] by a logarithmic factor in terms of number of edges.

In order to solve the data gathering problem, we define the weights of edges exactly in the same fashion as for the previously described multicast problem. The change is in the graph considered by the oracle: now we change the direction of edges with their corresponding weights. The target plays the role of source and set $I$ of data nodes plays a role of multicast set nodes. All the rest remains the same and leads to $O(|V|+|E|)$ time algorithm. This approach is possible since we consider the model of data gathering with aggregation (see [4,6,8]), meaning that any intermediate node can aggregate multiple incoming messages in a single outgoing message.

References
