Integer Maximum Flow in Wireless Sensor Networks with Energy Constraint*

Hans L. Bodlaender[1,**], Richard B. Tan[1,2], Thomas C. van Dijk[1], and Jan van Leeuwen[1]

[1] Department of Information and Computing Sciences
Utrecht University, Utrecht, The Netherlands
{hansb,rbtan,thomasd,jan}@cs.uu.nl
[2] Department of Computer Science
University of Sciences & Arts of Oklahoma, Chickasha, OK, USA

Abstract. We study the integer maximum flow problem on wireless sensor networks with energy constraint. In this problem, sensor nodes gather data and then relay them to a base station, before they run out of battery power. Packets are considered as integral units and not splittable. The problem is to find the maximum data flow in the sensor network subject to the energy constraint of the sensors. We show that this *integral* version of the problem is *strongly* NP-complete and in fact APX-hard. It follows that the problem is unlikely to have a polynomial time approximation scheme. Even when restricted to graphs with concrete geometrically defined connectivity and transmission costs, the problem is still strongly NP-complete. We provide some interesting polynomial time algorithms that give good approximations for the general case nonetheless. For networks of bounded treewidth greater than two, we show that the problem is *weakly* NP-complete and provide pseudo-polynomial time algorithms. For a special case of graphs with treewidth two, we give a polynomial time algorithm.

1 Introduction

A wireless sensor (or smart dust) is a small physical device that contains a microchip with a miniature battery, and a transmit/receive capability of limited range. Sensors have been deployed in different environments to gather data, perform surveillances and monitor situations in diverse areas such as military, medical, traffic and natural environments. (See e.g. Zhao and Guibas [16].)

As the battery power of a sensor is limited and non-replaceable, it is crucial to maximize the lifetime of the wireless sensor network to ensure the continuing function of the whole network. We study the situation of a data-gathering sensor network, where sensors are deployed in the field to gather data and then relay the data packets via other sensors back to a base station. It is desirable to get

* This work was supported by BSIK grant 03018 (BRICKS: Basic Research in Informatics for Creating the Knowledge Society).
** Corresponding author.

J. Gudmundsson (Ed.): SWAT 2008, LNCS 5124, pp. 102–113, 2008.
© Springer-Verlag Berlin Heidelberg 2008

as many data packets as possible from the source sensors to the base station, before some of the sensor batteries are depleted. This then becomes an instance of the maximum flow problem, subject to the energy constraint of the lasting battery power of each sensor.

Most of the research papers for the maximum flow problem with energy constraint on wireless sensor networks (e.g. [5,6,8,12,13,14,15]) cast the problem into a Linear Programming (LP) form and assume *fractional* flows, i.e., splitting of packets into fractional portions is allowed. The corresponding LP-formulations then have polynomial time algorithms. These papers then present several heuristics that speed up the algorithms and compare various simulation results. A few papers [5,8,12,13] modified the Polynomial Time Approximation Scheme (PTAS) of Garg and Könemann [10] to obtain fast approximation algorithms.

As data packets are usually quite small, there are situations where splitting of packets into fractional ones is not desirable nor practical. We consider a model where data packets are considered as units that cannot be split, i.e. the packet flows are of *integral* values only. We call this the *Integer* Maximum Flow problem for Wireless Sensor Network with Energy Constraint: the Integer Max-Flow WSNC problem. The corresponding LP formulation becomes Integer Programming (IP) and may no longer have a polynomial time solution.

We show that the problem is in fact *strongly* NP-complete, and thus unlikely to have a Fully Polynomial Time Approximation Scheme (FPTAS) or a pseudo-polynomial time algorithm, unless P=NP. This result also holds for a class of graphs with geometrically defined connectivity and transmission costs, even when the nodes lie on a line. Furthermore, we show that even for a special fixed range model, the problem is APX-hard, thus unlikely to have even a PTAS (unless P=NP). We also provide some approximation algorithms for the problem that do give good approximations nonetheless.

Many hard problems have polynomial time solutions when restricted to networks with bounded treewidth (see e.g. [3]). However, we show that for networks with bounded treewidth greater than two, the Integer Max-Flow WSNC problem is *weakly* NP-complete. We provide pseudo-polynomial time algorithms to compute integer maximum flows in this case. For a special case of graphs that have treewidth two, namely those graphs that have treewidth two when we add an edge from the single source to the sink, we provide a polynomial time algorithm.

The paper is organized as follows. In Section 2, we describe the model and the problem in detail. Section 3 covers the complexity issues. We show here the various NP-completeness results and describe some approximation algorithms. Section 4 contains the results for networks with bounded treewidth.

2 Preliminaries

In this section, we discuss the model we use and the precise formulation of the Integer Max-Flow WSNC problem. We also discuss some variants of the problem.

2.1 The Model

Our model of a sensor is based on the *first order radio model* of Heinzelman et al. [11]. A sensor node has limited battery power that is not replenishable. It consumes an amount of energy $\epsilon_{elec} = 50nJ/bit$ to run the receiving and transmitting circuitry and $\epsilon_{amp} = 100pJ/bit/m^2$ for the transmitter amplifier. In order to receive a k-bit data packet, a sensor has to expend $\epsilon_{elec} \cdot k$ energy, while to transmit the same packet from sensor i to sensor j will cost $\epsilon_{elec} \cdot k + \epsilon_{amp} \cdot k \cdot d_{ij}^2$ energy, where d_{ij} is the distance between sensors i and j.

We model a wireless sensor network as a *directed* graph $G = (N, A)$, where $N = \{1, 2, \ldots, n\} \cup \{t\}$ are the n sensor nodes along with a special non-sensor sink node t, and A is the set of directed arcs ij connecting node i to node j, $i, j \in N$. A sensor node i has energy capacity E_i and each arc ij has cost e_{ij}, the energy cost of receiving (possibly from some node) a packet and then transmitting it from node i to node j. We assume that no data is held back in intermediate nodes $\neq t$, i.e., data that flows in will flow out again, subject to the battery constraint of these nodes. All E_i's and e_{ij}'s are non-negative *integer* values.

The general model assumes that each sensor can adjust its power range for each transmission. We also consider in the next section the fixed range model, where each sensor has only a few *fixed* power settings. All our graphs are assumed to be *connected*. For each arc $ij \in A$, there is a directed path from a source node to the sink node that uses this arc.

2.2 The Problem

Given a wireless sensor network G, there is a set S of *source* sensor nodes, used for gathering data. The sink node t is a *base station* and is equipped with electricity and thus has unlimited energy to receive all packets. The remaining nodes are just *relaying* nodes, used to transfer data packets from the source nodes to the sink node. One would like to transmit as many packets as possible from the source nodes to the sink node. This is feasible as long as the battery power in the network suffices to do so. The transmission process can be viewed as a *flow* of packets from the sources to the sink. The problem is then to find the *maximum flow* of data packets in the network subject to the battery power constraint.

We assume that the data packets are quite small, thus it is neither reasonable nor practical to split them further into fractional portions. A *flow* f_{ij} is a function that assigns to each arc ij a non-negative integer value. This corresponds to the number of packets being sent via the arc ij. A flow is a *feasible* flow if $\sum_j f_{ij} \cdot e_{ij} \leq E_i$ for all nodes $i \in N$, where the sum is taken over all j with $ij \in A$; i.e., the flow through a node cannot exceed the battery capacity of the node.

We can now formulate the maximum flow problem for wireless sensor networks as the problem of determining the maximum number of packets that can be received by the sink node. We call this problem the Integer Maximum Flow problem for Wireless Sensor Networks with energy Constraints or *Integer Max-Flow WSNC problem* for short. The problem has the following Integer Linear Programming formulation.

The Integer Max-Flow WSNC problem:
Objective: maximize $F = \sum_{j \in N} f_{jt}$, t is the sink node,
subject to the following constraints:

$$f_{ij} \;\; integer, \qquad \forall ij \in A \tag{1}$$

$$f_{ij} \geq 0, \qquad \forall ij \in A \tag{2}$$

$$\sum_{j \in N} f_{ij} = \sum_{j \in N} f_{ji}, \qquad \forall i \in N - S - \{t\} \tag{3}$$

$$\sum_{j \in N} f_{ij} \cdot e_{ij} \leq E_i, \qquad \forall i \in N \tag{4}$$

Condition (3) is the conservation of flow constraint. It simply states that with the exception of the source and sink nodes, every node must send along the packets that it has received. Condition (4) is the energy constraint for the feasible flow: the energy needed to (receive and) transmit packets must be within the capacity of the battery power of each node. This condition also distinguishes the (integer) max-flow WSNC problem from the standard max-flow problem: there, the constraint condition is just $f_{ij} \leq c_{ij}$, where c_{ij} is the flow capacity of arc ij.

We note that without loss of generality, we can augment the network with a *super source* node s with unlimited energy to send and connect it to all the source nodes with some fixed cost. We can then view the network as having a single source and a single sink with a single commodity, subject to the battery energy constraint. However, note this may affect the treewidth of the network; the results for networks of bounded treewidth in Section 4 assume a single source.

We do not include a 'fairness' constraint: we maximize the number of packets reaching the sink, but do not balance the load of the source nodes. Note that the hardness proofs in Section 3 do exhibit fair flows, where all source nodes send an equal number of packets.

2.3 Other Variants

Other variants of the problem formulation exist for wireless sensor networks with energy constraint. See for example Floréen et al. [8] and Chang and Tassiulas [6].

3 Complexity

We will first look at the complexity of the problem on general graphs with arbitrary costs at the arcs. We show the problem is NP-complete. Next we show this proof carries over to a restriction of the problem to a class of graphs with geometrically defined connectivity and energy consumption. Then we look at another restriction of the problem, on general graphs again, but with only a fixed amount of distinct energy costs. The problem is polynomial time solvable for one energy level, but with more distinct power levels the problem is shown to be APX-hard. Lastly, we demonstrate a polynomial time approximation algorithm for the general case. Note however that this algorithm does not guarantee a constant approximation ratio.

3.1 General Graphs

Since each data packet is a self-contained unit and cannot be split, the corresponding LP formulation is an Integer Programming (IP) formulation and may no longer have a polynomial time solution. In fact, we prove that the problem is *strongly* NP-complete.

Theorem 1. *The decision variant of the Integer Max-Flow WSNC problem is strongly NP-complete.*

Proof. We reduce the 3-Partition problem to the decision version of the Integer Max-Flow WSNC problem.

> 3-Partition
> Instance: Given a multiset $\mathcal{S}$ of $n = 3m$ positive integers, where each $x_i \in \mathcal{S}$ is of size $B/4 < x_i < B/2$, for a positive integer B.
> Question: Can $\mathcal{S}$ be partitioned into m subsets (each necessarily containing exactly three elements) such that the sum of each subset is equal to B?

The 3-Partition Problem is strongly NP-complete [9].

For any instance I of the 3-Partition problem we create an instance I' of a wireless sensor network as follows. Each number $x_i \in \mathcal{S}$ corresponds to a sensor relay node r_i. Additionally, we have m source nodes $s_1, \ldots, s_m$ each having exactly B energy. The source nodes play the role of the subsets. Now connect each of the source nodes s_j with all the relay nodes r_i with arc cost $e_{ji} = x_i$. The intention is that it will cost each source node exactly x_i energy to send one packet to relay node r_i. We further connect all the relay nodes to a sink node t. Each relay node r_i has energy $E_i = B$ and arc cost $e_{it} = B$, just sufficient energy to send only one packet to the sink node.

Then our instance of the 3-Partition problem has a partition into m subsets $\mathcal{S}_1, \ldots, \mathcal{S}_m$, each with sum equals B if and only if for each subset $\mathcal{S}_i = \{x_{i1}, x_{i2}, x_{i3}\}$ the source node s_i sends three packets, one each to relay nodes r_{i1}, r_{i2}, r_{i3} consuming the energies x_{i1}, x_{i2}, x_{i3}, thus draining all of its battery power of $B = \sum_{j=1}^{3} x_{ij}$. This will give a maximum flow of $n = 3m$ packets for the whole network. Thus, 3-Partition reduces to the question whether the WSNC network can transmit at least $3m$ packets to the sink.

We have now given a pseudopolynomial reduction (see [9]), and thus we have shown that the Integer Max-Flow WSNC problem is strongly NP-complete. □

Corollary 2. *The Integer Max-Flow WSNC problem has no fully polynomial time approximation scheme (FPTAS) and no pseudo-polynomial time algorithm, unless P=NP.*

Proof. This follows from the fact that a strongly NP-complete problem has no FPTAS and no pseudo-polynomial time algorithm, unless P=NP. (See [9]). □

We show later on that even in a restricted case, the problem is APX-hard, i.e. the problem does not even have a PTAS unless P=NP.

3.2 The Geometric Model

We will now look at the geometric version of the problem in which the nodes are concretely embedded in space. In this version, each node has a position, and transmitting to a node at distance d costs d^2 energy. This quadratic cost is a typical model of radio transmitters.

Definition 3. *A* geometric configuration *is a complete graph where each node $i \in N$ has a location $p(i)$ and initial battery capacity E_i. The cost of the edge between vertices i and j is $e_{ij} = |p(i) - p(j)|^2$.*

In this section we show that the WSNC-problem remains NP-complete when restricted to geometric configurations where all nodes lie on a line. We will prove this for the case where we allow variable battery capacities E_i (Theorem 4). This can be extended to the case where all nodes have equal battery capacity (Theorem 9), but we omit the more elaborate proof [4] for space reasons.

Theorem 4. *The Integer Max-Flow WSNC-problem is strongly NP-complete on geometric configurations on the real line.*

Proof. Again, we give a pseudopolynomial reduction from 3-Partition. First we describe how to construct a geometric configuration $\mathbf{C}$ on the real line, where the WSNC-problem is equivalent to a given instance of 3-Partition. We then construct an equivalent configuration $\mathbf{C_P}$ that can be described in polynomial size. These steps together show that the WSNC-problem is strongly NP-complete on geometric configurations.

Like before, we have m source nodes $s_1, \ldots, s_m$. Each starts with $B = \sum x_i/m$ energy. These source nodes again play the role of the subsets and we place them all at the origin of our geometric configuration, i.e., $p(s_i) = 0$ for all i.

Corresponding to each $x_i \in \mathcal{S}$ we have a 'relay' node r_i that serves the same purpose as before: to receive one packet from a source node, costing x_i energy for this source node, and relay the packet to the sink. By setting $p(r_i) = \sqrt{x_i}$ we achieve that a source node must use x_i energy to send a packet to r_i.

Finally, we place a sink node t with $p(t) = \sqrt{B}$. We want each relay node to have just enough energy to send exactly one packet to the sink, so we set $E_{r_i} = (p(t) - p(r_i))^2 = B + x - 2\sqrt{B}\sqrt{x}$.

This concludes the construction of our geometric configuration $\mathbf{C}$. The construction is illustrated in Figure 1.

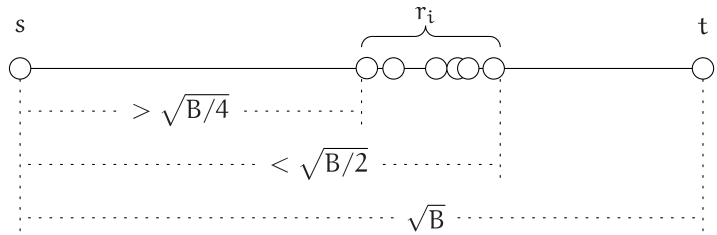

Fig. 1. Configuration **C**

Lemma 5. *Suppose we have a flow of value n in* **C***. Then every relay node receives exactly one packet from a source and sends it to the sink.*

Proof. If n packets reach the sink, then n packets must have left the sources. By the restriction on the values x_i of the 3-Partition instance, each edge leaving the sources costs strictly more than $B/4$ energy. Since the source nodes start with B energy, no source node can send more than 3 packets. There are only $m = n/3$ source nodes, so every source node must send exactly 3 packets. In particular, no packets are sent directly from a source node to the sink, as this would use up all energy of the source node. Therefore, the sink node only receives packets from relay nodes. No relay node can afford to send more than one packet to the sink, so in fact every relay node sends exactly one packet to the sink. □

Using Lemma 5, the following can now be shown in the same way as Theorem 1 for the case on arbitrary graphs.

Proposition 6. *The configuration* **C** *has a solution of the WSNC-problem with n packets if and only if the corresponding 3-Partition instance is* **Yes***.*

Note that this does not yet give an NP-hardness proof for the WSNC-problem on geometric configurations on the line, as configuration **C** has nodes at real-valued coordinates: the specification of the location of the points in **C** contains square roots. We shall now construct a geometric configuration $\mathbf{C_P}$ which is equivalent to **C**, but whose positions are all polynomially representable rational numbers.

We do this by choosing the locations as integer multiples of some ε (value to be determined later), rounding down. The initial power of the batteries also needs to be quantized. We give the source nodes exactly B energy, which is already integer. We give the relay nodes precisely enough energy to send one packet to the sink; this amount can be calculated from the actual distance in $\mathbf{C_P}$.

Lemma 7. *The value for ε can be chosen such that $\mathbf{C_P}$ is equivalent to* **C** *and can be represented in polynomial size.*

We omit the proof that e.g. $\varepsilon = (5\lceil\sqrt{B}\rceil)^{-1}$ satisfies the lemma [4]. The proof of Theorem 4 now follows from Lemmata 5, 7 and Proposition 6: the WSNC-problem is strongly NP-complete on geometric configurations, even when restricted to a line. □

The nodes in $\mathbf{C_P}$ have non-integer positions, but this not essential: scaling can achieve the following.

Corollary 8. *The Integer Max-Flow WSNC-problem is strongly NP-complete on geometric configurations on a line, where each node has an integer coordinate.*

The given results make essential use of the fact that battery capacities can vary. A natural next question is whether the results still hold in case all sensors have equal battery capacity. By a different construction and a non-trivial combinatorial argument one can show that this is indeed the case [4].

Theorem 9. *The Integer Max-Flow WSNC-problem is strongly NP-complete on geometric configurations on a line, even when all battery capacities are equal.*

3.3 Fixed Range Model

Now we return to the case for arbitrary graphs. Suppose that every sensor node has only a *fixed* number of power settings, i.e., only a fixed number of different energy cost values at its outgoing edges. For example, there may be only *one* setting, so that every node within the range is considered a neighbor; or perhaps there are only *two* settings: *short* and *long* range power settings.

It turns out that for the case when there is only one power setting, there is an easy solution. Since now the energy cost e_{ij} is the same for all neighbors j, the maximum flow capacity $f_{ij} = \lfloor E_i/e_{ij} \rfloor$ is also fixed for all outgoing arcs of node i. We can then transform the sensor network into a regular flow network by using the splitting technique in flow networks as follows. (See e.g. the book by Ahuja, et al. [1].) Split each node i into two nodes i and i' and connect them with an arc with capacity $c_{ij} = \lfloor E_i/e_{ij} \rfloor$. The capacity of the original arcs ij will also all be set to c_{ij}, for all $ij \in A$. We then have a new graph that is a flow network with twice as many nodes and n additional arcs. Then it is easy to see that this variant of the Max-Flow WSNC problem is just the standard Max-Flow Min-Cut problem and has a polynomial time algorithm of $O(n^3)$, even in the integer case. This fact has also been noted by Chang and Tassiulas [5]. For the sake of completeness, we record this fact below.

Theorem 10. *If there is only one power setting at each sensor node, then there is a polynomial time algorithm to solve the Integer Max-Flow WSNC problem.*

The situation changes when the number of fixed power settings is increased to two.

Theorem 11. *If there are two power settings at each sensor node, then there is no PTAS for the Integer Max-Flow WSNC problem, unless P=NP.*

Proof. We reduce a restricted version of the Generalized Assignment Problem (GAP) by Chekuri and Khanna [7] to the Integer Max-Flow WSNC problem with two power settings.

2-SIZE 3-CAPACITY GENERALIZED ASSIGNMENT PROBLEM (2GAP-3)

INSTANCE: A set $\mathcal{B}$ of m bins and a set $\mathcal{S}$ of n items. Each bin j has capacity $c(j) = 3$ and for each item $i \in \mathcal{S}$ and bin $j \in \mathcal{B}$, we are given a size $s(i,j) = 1$ or $s(i,j) = 1+\delta$ (for some $\delta > 0$) and a profit $p(i,j) = 1$.

OBJECTIVE: Find a subset $\mathcal{U} \subseteq \mathcal{S}$ of maximum profit such that $\mathcal{U}$ has a feasible packing in $\mathcal{B}$.

Chekuri and Khanna [7] show that the 2GAP-3 problem is APX-hard, hence it does not have a PTAS (unless P=NP).

Given an instance I of 2GAP-3 we create an instance I' of the Integer Max-Flow WSNC as follows. For each bin $j \in \mathcal{B}$ we have a source node j' with energy capacity $E_{j'} = 3$. Corresponding to each item $i \in \mathcal{S}$ we have a relay node i'. Each source node j' is connected to each of the relay nodes i' by an arc $j'i'$ with energy cost $e_{j'i'} = 1$ if $s(i,j) = 1$ and $e_{j'i'} = 1+\delta$ if $s(i,j) = 1+\delta$. We also have one sink node t. Each of the sensor relay node i' is further connected to the sink node t and provided with just sufficient battery power to have the arc energy cost to send only one packet to the sink node, i.e., we set $E_{i'} = 1$ and $e_{i't} = 1$.

As each node i' that represents an item can forward only one packet, each arc of the form $j'i'$, $j \in \mathcal{B}$, $i \in \mathcal{S}$ can also carry at most one packet. Thus, there is a one-to-one correspondence between integer flows in I fulfilling energy constraints, and feasible packings of sets of items $\mathcal{U} \subseteq \mathcal{S}$: an item i that is placed in bin j corresponds to a unit of flow that is transmitted from j' to i' and then from i' to t. The value of the flow equals the total profit of the packed items.

Thus, we can observe that our reduction is an AP-reduction. As AP-reductions preserve APX-hardness (see e.g., [2]), we can conclude the theorem. □

3.4 Approximation Algorithms

As the Integer Max-Flow WSNC problem has no PTAS (unless P=NP), our hope is to find some approximation algorithms. We first give a very simple approximation algorithm. Then we give a slightly more involved algorithm with a better approximation performance.

Theorem 12. *There is a ρ-approximation algorithm for the Integer Max-Flow WSNC problem, where $\rho = \max_{i \in N} \frac{\max_{ij \in A} e_{ij}}{\min_{ij \in A} e_{ij}}$.*

Proof. Convert the sensor network into a flow network as follows. We give the edges unbounded capacity and we give each node i the capacity $c_i = \lfloor \frac{E_i}{\max e_{ij}} \rfloor$. Then a standard Max-Flow Min-Cut algorithm with node capacities will yield a polynomial time algorithm that is at worst a ρ-factor from the optimum. □

Unfortunately the above approximation is not of constant ratio. Note that for the fixed range model where each sensor has a constant number of fixed power settings, the above algorithm does give a constant ratio approximation.

A better approximation algorithm is the following. The key idea is to first solve the fractional LP-formulation in polynomial time, and then try to find a large integer flow that is close to the optimum value.

Theorem 13. *There is a polynomial time approximation algorithm for the Integer Max-Flow WSNC problem, which computes an integer maximum flow with value $F_{approx} \geq F_{optimum} - m$, where m is the number of arcs of the graph.*

Proof. We give the proof for the case when there is only one source. It is a simple exercise to generalize the proof to the case with multiple sources.

First, we solve the relaxation of the problem optimally, i.e., we allow flows to be of real value. As this is an LP, the ellipsoid method gives us in polynomial time an optimal solution F^*, that can be realized within the energy constraints.

We now find a large integer flow inside F^* in the following way. We use an integer flow function F, which invariantly will map each arc ij to a non-negative integer f_{ij} with $f_{ij} \leq f^*_{ij}$, and which has conservation of flows; i.e., F will invariantly be a flow that fulfills the energy constraints. Initially, set F to be 0 on all arcs.

Now, repeat the following step while possible. Find a path P from s to t, such that for each arc ij on the path, $f^*_{ij} - f_{ij} \geq 1$. Let $f_P = min_{ij \in P} \lfloor f^*_{ij} - f_{ij} \rfloor$ be the minimum over all arcs ij on the path P. Note that $f_P \geq 1$. Now add f_P to each f_{ij} for all arcs ij on the path P. Observe that the updated function F fulfills the energy constraint conditions.

The process ends when each path from s to t contains an arc with $f^*_{ij} - f_{ij} < 1$. We note that $F^* - F$ is a flow, and standard flow techniques show that its value is at most m, the number of arcs. (For let S be the set of nodes, reachable from s by a path with all arcs fulfilling $f^*_{ij} - f_{ij} \geq 1$. Since $t \notin S$, $(S, V - S)$ is a cut and its capacity with respect to the flow $F^* - F$ is at most the number of arcs across the cut.) Since the value of F^* is at least $F_{optimum}$, and hence the value of F is at least $F_{optimum} - m$.

In each step of the procedure given above, we obtain at least one new arc ij with $f^*_{ij} - f_{ij} < 1$; this arc will no longer be chosen in a path in a later step. Thus, we perform at most m steps. Each step can be done easily in linear time. Hence, the algorithm is polynomial, using the time of solving one linear program plus $O(m(n + m))$ time for computing the approximate flow. □

This algorithm is still not of constant ratio but we conjecture that it is a step toward a 2-approximation algorithm.

4 Graphs with Bounded Treewidth

Many hard graph problems have polynomial (sometimes even linear) time algorithms when restricted to graphs of bounded treewidth. This however is not the case for the Integer Max-Flow WSNC problem: the problem remains hard. The treewidth parameter nicely delineates the classes of graphs for which the Integer Max-Flow WSNC is of apparently increasing complexity in the following manner.

Theorem 14. *The Integer Max-Flow WSNC problem can be solved in linear time for graphs of treewidth 1.*

Proof. These are just *forests* and have a very simple linear time algorithm: remove all nodes that do not have a path to the sink t; compute in the resulting tree in post-order for each node the number of packets it receives from its children and then the number of packets it can send to its parent. □

We show that if there is a single source, and the graph with an edge added between this single source and the sink node has treewidth two, then there is a polynomial time algorithm for the Integer Max-Flow WSNC problem. The general case for treewidth two remains open.

Theorem 15. *The Integer Max-Flow WSNC problem, with parallel arcs and arc capacities and with a single source s and sink t, can be solved in $O(m \log \Delta)$ time on directed graphs $G = (N, A)$, such that the graph $(N, A \cup \{st\})$ has treewidth at most two, where Δ is the maximum outdegree of a node in G, and $m = |A|$.*

For larger treewidth we show that the problem is *weakly* NP-complete. We give a pseudo-polynomial time algorithm for this class.

Theorem 16. *The Integer Max-Flow WSNC problem is (weakly) NP-hard for graphs of treewidth three.*

The omitted proof of Theorems 15 and 16 can be found in [4].

As shown in the previous section, the case of unbounded treewidth is *strongly* NP-complete and even APX-hard.

5 Conclusion

The Maximum Flow WSNC problem is an interesting and relevant problem, with practical implications in the context of wireless ad hoc networks. In this paper, we studied the integer variant of the problem. We obtained a good polynomial time approximation algorithm for the problem, which is a step toward an algorithm of constant performance ratio. We also studied how the complexity of the problem depends on the treewidth of the network. We found that except for the case where each sensor has one fixed power setting or when the underlying graph is of treewidth two with an edge joining the source and sink nodes, the problem is weakly NP-complete for bounded treewidth greater than two. It is strongly NP-complete for networks of unbounded treewidth and in fact even APX-hard. It is also strongly NP-complete on geometric configurations on a line.

Acknowledgements. We thank Alex Grigoriev, Han Hoogeveen, Arie Koster, Erik van Ommeren and Gerhard Woeginger for fruitful discussions and helpful comments.

References

1. Ahuja, R.K., Magnanti, T.L., Orlin, J.B.: Network Flows: Theory, Algorithms, and Applications, Upper Saddle River. Prentice-Hall, Englewood Cliffs (1993)
2. Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccamela, A., Protasi, M.: Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties. Springer, Berlin (1999)

3. Bodlaender, H.L.: A tourist guide through treewidth. Acta Cybernetica 11, 1–23 (1993)
4. Bodlaender, H.L., Tan, R.B., van Dijk, T.C., van Leeuwen, J.: Integer maximum flow in wireless sensor networks with energy constraint. Technical Report UU-CS-2008-005, Department of Information and Computing Sciences, Utrecht University (2008)
5. Chang, J., Tassiulas, L.: Fast approximate algorithms for maximum lifetime routing in wireless ad-hoc networks. In: Pujolle, G., Perros, H.G., Fdida, S., Körner, U., Stavrakakis, I. (eds.) NETWORKING 2000. LNCS, vol. 1815, pp. 702–713. Springer, Heidelberg (2000)
6. Chang, J., Tassiulas, L.: Maximum lifetime routing in wireless sensor networks. IEEE/ACM Transaction on Networking 12(4), 609–619 (2004)
7. Chekuri, C., Khanna, S.: A polynomial time approximation scheme for the multiple knapsack problem. SIAM J. Comput. 35(3), 713–728 (2005)
8. Floréen, P., Kaski, P., Kohonen, J., Orponen, P.: Exact and approximate balanced data gathering in energy-constrained sensor networks. Theor. Comp. Sc. 344(1), 30–46 (2005)
9. Garey, M.R., Johnson, D.S.: Computers and Intractability, A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, New York (1979)
10. Garg, N., Könemann, J.: Faster and simpler algorithms for multi-commodity flow and other fractional packing problems. In: Proceedings of the 39th Annual Symposium on Foundations of Computer Science, pp. 300–309 (1998)
11. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: Proceedings 33rd Hawaii International Conference on System Sciences HICSS-33 (2000)
12. Hong, B., Prasanna, V.K.: Maximum lifetime data sensing and extraction in energy constrained networked sensor systems. J. Parallel and Distributed Computing 66(4), 566–577 (2006)
13. Kalpakis, K., Dasgupta, K., Namjoshi, P.: Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks. Computer Networks 42(6), 697–716 (2003)
14. Ordonez, F., Krishnamachari, B.: Optimal information extraction in energy-limited wireless sensor networks. IEEE Journal on Selected Areas in Communications 22(6), 1121–1129 (2004)
15. Xue, Y., Cui, Y., Nahrstedt, K.: Maximizing lifetime for data aggregation in wireless sensor networks. Mobile Networks and Applications 10, 853–864 (2005)
16. Zhao, F., Guibas, L.: Wireless Sensor Networks: An Information Processing Approach. Morgan Kaufmann, San Francisco (2004)