Wireless Sensor Network, **2010, 2, 924-935**
doi:10.4236/wsn.2010.212111 Published Online December 2010 (http://www.scirp.org/journal/wsn)

Scientific Research

Integer Programming Formulations for Maximum Lifetime Broadcasting Problems in Wireless Sensor Networks

Roberto Montemanni
Istituto Dalle Molle di Studi sull'Intelligenza Artificiale,
Lugano, *Switzerland*
E-mail: *roberto@idsia.ch*

Received November 2, 2010; *revised November* 10, 2010; *accepted November* 17, 2010

Abstract

Approaches based on integer linear programming have been recently proposed for topology optimization in wireless sensor networks. They are, however, based on over-theoretical, unrealistic models. Our aim is to show that it is possible to accommodate realistic models for energy consumption and communication protocols into integer linear programming. We analyze the maximum lifetime broadcasting topology problem and we present realistic models that are also shown to provide efficient and practical solving tools. We present a strategy to substantially speed up the convergence of the solving process of our algorithm. This strategy introduces a practical drawback, however, in the characteristics of the optimal solutions retrieved. A method to overcome this drawback is discussed. Computational experiments are reported.

Keywords: Sensor Networks, Mixed Integer Linear Programming, Energy Models, Topology Optimization

1. Introduction

Since the very beginning of research in the area of wireless sensor networks, one of the major issues has been saving power. This optimization is typically faced during the design, and prior the deployment of the nodes of a network. Such a high attention for this factor is easy to identify: the nodes of the network (devices) are typically equipped with low capacity, tiny batteries, and they have to stay alive in the longest possible time horizon. This in an environment usually characterized by reduced accessibility. A tight management of the power budget is imposed by all these factors. Another peculiarity of sensor networks is that the largest share of power consumption is normally due to communication rather than to computation, sensing and state-changing activities [1,2]. We will base our study on this assumption.

Wireless sensor networks are typically used in commanding actuators, monitoring events or measuring values at locations where people cannot reach easily, or where a long term sensing task is required. Examples of applications are habitat monitoring [3], civil structural monitoring [4] and environmental monitoring [5]. Nodes can usually be characterized as low cost devices, and are to be deployed in a potentially inaccessible area. Recharging the sensors after the deployment might therefore not be an option, both for logistic and economical reasons. In this context, energy-efficiency becomes perhaps the most important design criteria for sensor networks, since it directly impacts on the time the network itself is kept in operation. Many sensor network applications are intrinsically about dissemination of information from a well-identified source node. It is therefore critical to identify energy efficient network topologies, optimized according to the type of communications that has to be supported. In this paper we will concentrate on broadcasting topologies, where a piece of information has to be sent from a source node to all of the other nodes of the network. We will assume to work on a static network, i.e. nodes do not move after the deployment.

Researchers have proposed many different mechanisms for achieving energy-efficiency in sensor networks. For example, transmission power control in radios, periodic cycling of nodes' activity schedules [6], in-network processing of sensor data [7] and aggregation [8] for data-gathering applications. In most of the communication models adopted in these studies, the total energy cost for a transmitted bit is computed as the cost of a transmission over a wireless channel over a certain distance. In some cases, the cost due to reception by the destination radio hardware [9] is also taken into account. Most of the communication models however ignore this, do to the multicast nature of the wireless channel, many nodes in the vicinity of a sender node overhear its packet transmissions, even if they are not the intended recipients of these transmissions [10]. This redundant reception results in unnecessary expenditure of battery energy of all the recipients. It is finally worth to observe that temporarily turning off neighboring radios during a certain point-to-point wireless transmission can mitigate overhearing costs [1,2].

The research in the field of integer linear programming models for energy-efficiency in wireless sensor network has been focused until now on very theoretical representation of the interactions among nodes. In most of the works [11,12,13] the power consumed by a communication from a device to another one is simply evaluated as the distance between the devices elevated to the power of 2 (or 4). This model is extremely simplistic, and many studies that show this have been proposed. For example, it has been shown that energy is consumed while receiving [1,10], and that also the overhearing phenomenon (mentioned above) has an important impact on consumption [14]. In order to limit the effects of overhearing, some authors suggest the use of nodes able to turn themselves into a temporary sleeping mode when a communication is not for them [6,10]. It is worth to observe that the idea of going into a temporarily sleep mode can be extended to situations in which nodes know that no communication is going to happen in a subsequent time window of a given length. Finally, an interesting extension of the models presented so far might consider heterogeneous networks, where devices with different characteristics (in terms of battery capacity and/or maximum transmission power) are part of the same network.

The topology design of wireless sensor networks is usually split into two phases: selection and placement of the nodes [15] and power optimization of the transmission powers of the nodes. In this paper we will concentrate on the second phase.

The aim of the present paper is to plug the realistic issues and ideas mentioned before into integer linear programming models. The objective of the problem we model is maximizing the lifetime of wireless sensor networks [16]. The realistic issues we will embed into our models cover the following aspects:

Hardware/Network models. Characteristics of the devices are considered. In particular, each device can be characterized by a maximum transmission power and a residual battery capacity at deployment time. Heterogeneous networks, formed by devices with different characteristics are considered.

Energy consumption models. The energy consumed in the following situations is taken into account: transmitting, receiving, overhearing, sensing, computing and state-changing.

Communication models. We take into account protocols that, in addition to straight communication, are able to switch into a long-time sleep mode when no message is going to be transmitted for a given time, and to switch into a temporarily (short-time) sleep mode when they understand they are overhearing a message.

As far as we are aware, the one we present is the first integer linear programming framework able to optimally solve problems of realistic size with the characteristics listed above. Notice that a preliminary (and partial) version of the work presented in this paper appeared in [17].

Sections 2, 3 and 4 are devoted to the formal description of the models adopted to describe hardware, energy consumption and communication, respectively. Integer linear programming models of increasing realism, are described in Section 5, where a solution approach is also introduced and experimental results are presented and discussed. Section 6 presents a speed-up strategy based on an observation about the characteristics of the solving procedure previously discussed. Experimental results confirm that the strategy is able to remarkably improve the convergence speed of the algorithm in many cases. Section 7 discusses a practical drawback introduced by the speed-up strategy. A very fast post-processing procedure that overcomes this drawback is also discussed. Conclusions are drawn in Section 8.

2. Hardware/Network Models

We assume that a static network formed by heterogeneous nodes is provided, with a given node s designed as the source for the broadcasting process.

Messages have to be periodically sent from the source node to all the other nodes in the network. Any node can be used as a relay node to reach other nodes in the network, realizing a so-called *multi-hop* communication. We assume all nodes to be equipped with omnidirectional antennae, so that if node i transmits to node j, all nodes closer to i than j will also receive the transmission. This phenomenon has been defined in [13] as the *wireless multicast advantage*, meaning that transmitting at high powers can be convenient, since many nodes are potentially reached by a single transmission. An example of the wireless multicast advantage is provided in **Figure 1**, where, for the sake of simplicity, we assume that the power required for a transmission is proportional to the diameters of the circle representing transmission ranges. We will keep this assumption for all the examples discussed in this paper. In **Figure 1**, node r transmits at such a power to reach node j. Because of the wireless multi- cast advantage, also nodes l and k will receive the message, since they are closer to *r* than *j*.

We assume the power necessary to transmit from each node of the network to each other node to be known in advance. It usually depends on the characteristics of the terrain in which nodes are deployed. We will refer to the power necessary to transmit a bit of information from node *i* to node *j* as p_{ij}. We also assume to know, for each node *i*, its maximum transmission power MP_i and its residual battery capacity at deployment time, referred to as cap_i. To represent the problem in mathematical terms, we can therefore define a complete directed graph $G = (V, A)$, where *V* is the set of nodes of the network, and *A* is a set of arcs corresponding to all the possible one-hop communications between the nodes of the network: $(i, j) \in A \ \forall i, j \in V$. Power p_{ij} can be then seen as a label associated with arc (i, j).

It is important to stress that the models and methods we will discuss in Section 5 are independent on the model used to estimate power requirements.

Notice that placement and characteristics of the nodes are regarded as the output of a previous optimization phase.

3. Energy Consumption Models

In most of the power-optimization studies proposed so far [2,11,13,16], very simplistic, *basic* energy models were considered: typically anything different from transmitting power p_{ij} (see Section 2) was classified as negligible, and therefore ignored. On the other hand, many works exist to precisely quantify the energy consumption involved in the different operations carried out by the nodes [10].

In wireless sensor networks nodes consume energy while sensing, changing state, computing and communicating. Sensing, state-changing and computing activities are strictly application dependent, and we therefore assume to have an estimate for the energy consumption involved in them. Different is the situation for the power consumed in communication activities. We assume that energy is consumed when a packet is sent, received, and overheard, i.e. when a node receives a message that was not destined to it. The model we adopt is that proposed in [14]. In the network of **Figure 1**, node *k* (which has already received the transmission from *r*) overhears the communication from node *j* to node *m*, being closer to *j* than m. Considering the overhearing phenomenon makes the energy model particularly realistic.

Let us formally define what we will refer to as the *advanced* energy model. Energy consumption in one bit data transmission, ε_{tx}, is proportional to the transmitting power *p* of the transmitting node:

$$\varepsilon_{tx} = \varepsilon_{txelec} + \in p \tag{1}$$

where ε_{txelec} is the energy consumption by transmitter electronics, $\in$ is an amplifier characteristic constant (that depends on the media) .

Energy consumption in reception is independent of the transmitting power, and refers to the energy consumed by receiver electronics. We will refer to the energy dissipated for receiving a bit of information as ε_{rc}.

The logic for the calculation of the energy dissipated in overhearing activities ε_{oh}) is the same as for the calculation of the energy required while receiving. In this paper, following the approach already adopted in [14], we will assume that the energy consumed while overhearing is exactly the same as the energy consumed when receiving: $\varepsilon_{oh} = \varepsilon_{rc}$. This assumption leads to a simplified notation (without ε_{oh}), which is adopted in the reminder of the paper.

The energy dissipated in computing, state-changing and sensing has finally to be considered. As previously observed, this quantity is extremely application-dependent, and has to be estimated case by case. We therefore assume to know in advance the cumulative energy dissipated by each node i in these tasks during each broadcasting cycle. We will refer to it as $\varepsilon_{sc(i)}$. Notice that this quantity is not necessarily the same for all the nodes, and that the energy dissipated between two consecutive broadcasting cycles is also counted here: if the devices are able to switch themselves into a temporary sleeping mode between two consecutive broadcasting cycles, the energy consumed will be low, otherwise the energy dissipated while in idle state will be higher.

4. Communication Models

Two different communication models are considered in this paper, based on two different communication protocols implemented in the reality.

The *standard communication protocol* considered is that working on devices without the capability of switching into a temporary sleep mode when they under

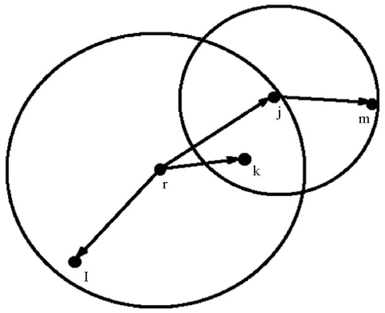

Figure 1. The wireless multicast advantage [13]. Example of overhearing phenomenon.

stand that a message that is being transmitted is not of their interest. The smart communication protocol is that associated with advanced devices able to switch into a temporary sleep state. Notice that this protocol implies the presence of header messages, used to identify the subsequent data messages. Header messages will intuitively contain the information of the data message that will follow, together with the length of the message itself. Consider again the example in **Figure 1**. If the smart protocol is in use, node *k* can temporarily turn itself off when it understands that the transmission generated from node *j* is not of its interest (it is about a message already received at *k* while transmitted from *r*).

It is easy to foresee that a smart communication protocol can be very useful for mitigating power consumption due to overhearing at the nodes. This justifies the increase in the complexity of both the hardware of the devices and the communication protocol.

In the reminder of the paper we will refer to the dimension of the data packet that has to be broadcasted from the source node to all the other nodes of the network as D . We will also indicate with the H dimension of the header of the data packet, which will contain a message-id and the dimension of the following data packet. The header will be sent immediately before the data packet and (if a smart communication protocol is used) nodes hearing it will know if they have received the message already, and undertake the appropriate actions.

5. Integer Linear Programming Formulations

5.1. Maximum Transmission Powers

According to Section 2, a maximum transmission power MP_i is given for each node . This parameter is, in fact, transparent to the integer linear programming models we will propose.

Consider the case where the transmission power required to transmit from a node *i* to another node *j* (p_{ij}) is greater than MP_i . It means that node *i* is not able to transmit to *j* in a single hop. We can therefore modify p_{ij} and assign an infinite value to it. This will rule out the single-hop communication *i*–*j* from every feasible solution of our models. Formally, we will modify the power requirements for the one hop communication for some pairs of nodes (i, j), as follows:

$$p_{ij} > MP_i \Rightarrow p_{ij} := +\infty \tag{2}$$

5.2. Skeleton Structure

The integer linear programming models we propose are all based on the common idea of skeleton structure. Such an approach is common in the literature of topology optimization (see, for example [11,18]). Each feasible solution is characterized by at least a skeleton, which ensures that all the nodes of the network receive, in a multi-hop fashion, the message broadcasted from the source node *s*. For example, in **Figure 1** a skeleton of the broadcasting structure is the depicted directed tree (arborescence) rooted in *s*. Notice that all the transmission powers of the nodes are such that the arcs of the arborescence are covered. The vice-versa is not true: not all the arcs covered by the power assignment are part of the arborescence. In our approach, we will adopt a structure that contains (is a superset of) an arborescence.

We define the skeleton in terms of mixed integer programming formulation. The transmission power $r(i)$ assigned to a generic node *i* will assume either a value of zero or of p_{ij}, for some $j \in V$. A set of variables *x* is then introduced to describe the transmission power of each node. For each pair of nodes $i,j \in V$ we have:

$$x_{ij} = \begin{cases} 1 & \text{if } \mathrm{r}(i) = p_{ij} \\ 0 & \text{otherwiise} \end{cases} \tag{3}$$

It is easy to verify that if $x_{ij} = 1$, then all the nodes closer to *i* than *j* will also receive (or overhear) the transmission from i. We have the following definition:

Definition 1 Given two nodes *i* and *j* such that $p_{ij} < +\infty$, we define $N_i(j)$ as follows:

$$N_i(j) := \{k \in V \mid pik \geq pij\} \tag{4}$$

The set $N_i(j)$ contains the nodes of *V* which are not closer to *i* than *j*. Notice that $j \in N_i(j)$. The set *Skl* of all possible skeleton structures defined on graph *G*, is described by the following constraints:

$$\sum_{\substack{i \in C, \\ j \in U_{k \in V \setminus C} Ni(k)}} x_{ij} \geq 1 \quad \forall \mathrm{C} \subset \mathrm{V}, \mathrm{s} \in \mathrm{C} \tag{5}$$

$$\sum_{j \in V} x_{ij} \leq 1 \quad \forall i \in V \tag{6}$$

$$\sum_{i \in V} x_{is} = 0 \tag{7}$$

$$x_{ij} \in \{0,1\} \quad \forall (i,j) \in \mathrm{A} \tag{8}$$

In constraints (5), *C* is used to model each non-empty subset of *V* containing the source node *s*. For each one of these sets, there is a constraint in the model specifying that at least one of the nodes of *C* must transmit at such a power to reach at least one node out of *C*. Constraints (6) state that at most one arc among the outgoing ones from each node must be active. Constraint (7) states that none of the incoming arcs of the source node can be active. Constraints (8) define variables domain.

5.3. Objective Function

In our optimization we seek for a broadcasting structure that maximizes the number of broadcasting cycles supported before the first device runs out of battery. The power consumption at each node of the network and its initial battery capacity are taken into account. This translates into counting the number of broadcasting cycles supported by each node, and choosing the smallest value.

Given an skeleton $x \in Skl$, we will refer to the power consumed in a single broadcasting cycle by node i as $pc(i,x)$. The problem studied in the paper can be formally defined through the following equation:

$$n = \max_{x \in Skl} \min_{i \in V} \left\{ \frac{cap_i}{pc(i,x)} \right\} \tag{9}$$

where n is the number of broadcasting cycles supported by the optimal broadcasting structure. Constraints (9) cannot be directly expressed in terms of mixed integer linear programming, because the variables of the model are at the denominator. It is however possible to manipulate (9) in order to obtain an equivalent description that can be expressed as a mathematical program. If we invert the fraction in (9), we end up with an expression for the inverse of the number of cycles supported by each node. The min and max operators have also to be complemented as a consequence of the inversion. We obtain the following equation:

$$\frac{1}{n} = \min_{x \in Skl} \max_{i \in V} \left\{ \frac{pc(i,x)}{cap_i} \right\} \tag{10}$$

from which it is trivial to obtain the original value of n, once (10) has been solved. Starting from (10), and assuming that we are able to find a linear expression for $pc(i,x)$, a mixed integer linear program is easy to obtain. First we have to get rid of the nested max operator. This can be done by introducing a free variable z, and by transforming (10) into the following system, which is linear if we can find a linear expression for $pc(i,x)$:

$$\frac{1}{n} = \min z \tag{11}$$

$$s.t.\ z \geq \frac{pc(i,x)}{cap_i} \quad \forall i \in V \tag{12}$$

$$x \in Skl \tag{13}$$

$$z \in IR \tag{14}$$

Notice that constraints (13) can be substituted by their linear expression, which is provided by inequalities (5), (6) and (7).

What remains to be disclosed is the expression in linear terms of $pc(i,x)$. In Subsections 5.4, 5.5 and 5.6 we will see how this can be done for different combinations of the energy and communication models described in Sections 3 and 4.

5.4. Model M1: Basic Energy Model, Basic Communication Protocol

The energy and communication models adopted for this formulation are those commonly used in the theoretical studies for minimum power broadcasting structures [11,13] and maximum lifetime broadcasting structures [16]. A very high level of abstraction is implied, and therefore these models are often regarded as too theoretical.

The energy model does not take into account the energy consumed while receiving and overhearing (ε_{rc} = ε_{oh} = 0), flattering down also the role of the communication protocol (a smart protocol is not of any help if overhearing is not taken into account).

It is important to notice that this model, although not of particular practical interest, has been considered in this study because it provides a sort of a bridge between the very theoretical mathematical programming models provided so far [11,13,18], and the more realistic models M2 and M3 we will describe in Sections 5.5 and 5.6.

The model is composed of constraints (11), (5), (6), (7) and (14), plus the following inequalities (15), that make constraints (12) explicit.

$$z \geq \frac{\varepsilon_{sc}(i)}{cap_i} + \sum_{j \in V \setminus \{i\}} \frac{(H+D)(\varepsilon_{tr} + \in p_{ij})}{cap_i} x_{ij} \quad \forall i \in V \tag{15}$$

The first term in the right hand side of constraints (15) models the fraction of the available energy consumed by nodes during each broadcasting cycle for sensing, computing and state-changing. The second term models the fraction of battery consumed for transmitting during each cycle. Notice that for each node i, at most one of the x_{ij}s will take value 1 in each feasible solution, because of constraints (5), (6) and (7). Consequently, in each one of constraints (15), no more than one x variable will take value 1.

5.4. Model M2: Advanced Energy Model, Basic Communication Protocol

Model M2 is for problems described through the advanced communication models presented in Section 3, but where no smart communication protocol is implemented (see Section 4).

Beside constraints (11), (5), (6), (7) and (14), model M2 includes the following inequalities (16), that make constraints (12) explicit.

$$z \geq \frac{\varepsilon_{sc}(i)}{cap_i} + \sum_{j \in V \setminus \{i\}} \frac{(H+D)(\varepsilon_{tr} + \in p_{ij})}{cap_i} x_{ij} + \sum_{\substack{j \in V \setminus \{i\} \\ k \in N_j(i)}} \frac{(H+D)\varepsilon_{rc}}{cap_i} x_{jk} \quad \forall i \in V \tag{16}$$

The first term in the right-hand-side of constraints (16) has the same meaning as in constraints (15). The first sum represents the fraction of the available energy dissipated by node i for data transmission. Notice that if $x_{ij}= 0$ $\forall j \in V$, it means that node i does not transmit in the current skeleton, and consequently this term will give a contribution of 0. The second sum represents the fraction of the available energy dissipated while receiving and overhearing. Notice that this contribution is paid for each communication reaching node i.

5.6. Model M3: Advanced Energy Model, Smart Communication Protocol

The integer linear programming model presented in this section builds up on that proposed in Subsection 5.5. The difference is that now the advantages derived from the use of a smart communication protocol (which is useful when nodes can go in a sleep mode when they are not interested in a transmission) have been taken into account.

This model can be regarded as the most realistic model we present in this paper, and is based on the following constrains: (11), (5), (6), (7), (14), plus (17) and (18):

$$z \geq \frac{\varepsilon_{sc}(i)}{cap_i} + \sum_{j \in V \setminus \{i\}} \frac{(H+D)(\varepsilon_{tr} + \in p_{ij})}{cap_i} x_{ij} + \sum_{\substack{j \in V \setminus \{i\} \\ k \in N_j(i)}} \frac{H\varepsilon_{rc}}{cap_i} x_{jk} + \frac{(H+D)\varepsilon_{rc}}{cap_i} \quad \forall i \in V \setminus \{8\} \tag{17}$$

$$z \geq \frac{\varepsilon_{sc}(s)}{cap_s} + \sum_{j \in V \setminus \{s\}} \frac{(H+D)(\varepsilon_{tr} + \in p_{sj})}{cap_i} x_{ij} + \sum_{\substack{j \in V \setminus \{s \\ k \in N_j(s)}} \frac{H\varepsilon_{rc}}{cap_s} x_{jk} \tag{18}$$

The first term in the right-hand-side of inequalities (17) and (18) is the usual fractional power consumption due to sensing, computing and state-changing during each broadcasting cycle. The first sum of constraints (17) is the same as in constraint (16). The second sum models overhearing at node i, following the same reasoning already seen for the second sum of (16): the sum contains all the communications potentially overheard by i, that will give a contribution or not to the sum according to the value of the x variables associated. The last term finally models the fractional energy consumption due to data reception (not overhearing). This contribution is based on the consideration that each node (apart from the source s) will receive the data packet exactly once. Notice that each node will be able to classify a packet as already received by analyzing the corresponding header. The only difference between constraints (17) and (18), is that in the right-hand-side of the former, the last term of (17) is not present, since the root will not receive any message, being the node from which the broadcasting is actually originated.

5.7. Solution Method

The bottleneck of the three mixed integer linear programming formulations described in Subsections 5.4, 5.5 and 5.6 is represented by the common constraints (5). Specifically, constraints (5) are present in huge number (exponential in the number of devices in the network), and create difficulties to the solvers used to handle the models. To overcome these problems, we adopted a strategy commonly used in similar cases (see, for example, [19]), and that can be summarized as follows. It is an iterative approach, which in the beginning does not consider any of the constraints (5), and adds them only when violated by the current (infeasible) solution.

In the reminder of this section we will refer to a generic mixed integer linear program IP. Formulations M1, M2 and M3 (initially without constraints (5)) can all substitute the generic formulation IP.

At each iteration, the integer linear program IP is solved to optimality and the values of the x variables in the optimal solution are examined. If the arcs corresponding to variables with value 1 (we will refer to them as *active* variables) form a directed structure that reaches all the devices of the network (*skeleton*), then the problem has been solved to optimality, and we can stop. Otherwise, a set of violated constraint of type (5) is identified (by solving maximum flow problem, see [20]) and added to the mixed integer linear program IP, and the process is repeated. In case the optimal solution has not been found, let us consider the arcs corresponding to the active x variables of the last available solution. We define the set C of the constraint (5) as the set of nodes that can be reached from the source s using arcs associated with active x variables only. The new constraint added makes the current solution infeasible for the newly augmented problem IP, forcing the solver to move to a different solution in the next iteration. Iteration after iteration, the incumbent solution x should get closer and closer to feasibility with respect to the original problem, being it of type M1, M2 or M3. The algorithm is summarized in **Figure 2**, where a pseudo-code is presented.

A well-known technique (see, for example, [20]) to

```
Create integer program $IP$; Solve integer program $IP$;
$x$ := value of the $x$ variables in the optimal solution of $IP$;
$C$ := nodes reached from $s$ according to $x$ variables;
While ($|C| < |V|$)
  Add inequality (5) to $IP$; Solve integer program $IP$;
  $x$ := value of the $x$ variables in the optimal solution of $IP$;
  $C$ := nodes reached from $s$ according to $x$ variables;
Return $x$ (optimal solution for the original problem);
```

Figure 2. A pseudo-code for the algorithm used to solve integer linear formulations M1, M2 and M3.

speed up the algorithm described above first considers LR, the linear relaxation of the mixed integer linear programming formulation IP. It is obtained by substituting constraints (8) with the following one:

$$0 \le x_{ij} \le 1 \ \ \forall (i,j) \in \mathrm{A} \tag{19}$$

The algorithm described in **Figure** 2 is run on LR. When there are no more violated cuts, which means LR has been solved to optimality, the original problem IP is considered, and the algorithm presented in Figure 2 is run again, starting from the set of cuts identified for the linear relaxation LR (instead of an empty set). The aim of such a technique is to reduce the number of mixed integer programs that have to be solved to identify violated cuts, in the hope that many of the cuts were already identified on the linear relaxation LR, which is off course much easier to solve, not having binary variables. We will adopt this strategy in our implementation.

5.8. Experimental Results

Computational experiments have been carried out on random networks with up to 80 nodes. The signal propagation model adopted for the experiments is the one commonly used in the literature (see, for example, [13, 21]). For a node *i* of the network, the power required to reach another node *j* is given by:

$$p_{ij} = \left(d_{ij}\right)^2 \tag{20}$$

The data and energy parameter settings adopted for the simulations are the following ones: $D =$ 500 *bites*, $H =$ 10 *bites*, $\in \ = 0.1$ *nJ/bit/m*^{2}, $\varepsilon_{txelec} = 50$ *nJ/bit*, $\varepsilon_{rc} = \varepsilon_{oh} = 50$ *nJ/bit*, $\varepsilon_{rc}(i) = 50$ *nJ/cycle*. For every problem considered, the capacity cap_i associated with the battery of node *i* (expressed in Joule for the sake of simplicity) is chosen at random from the interval [1000; 5000].

Problems with $|V|$ (number of nodes) $\in$ {20, 30, 40, 50, 60, 70, 80} have been considered. Ten instances have been generated at random on a 100×100 grid for each value of $|V|$. Tests have been carried out on a computer equipped with an Intel Pentium M 1.73GHz processor and 512 MB of memory. IBM ILOG CPLEX 12.1 (http://www.cplex.com) has been used to solve mixed integer linear programs. A maximum computation time of 3 hours has been allowed for each combination (model, instance).

Results are summarized in **Table 1**, where for each combination (model, $|V|$) we report average and standard deviation for the number of cuts (5) added (both on the linear relaxation phase and in total), and for the computational time in seconds. Statistics are computed on the instances solved to optimality (reported in the second column of the table) over the ten considered for each line of the table.

The computational results suggest that two of the three methods we developed are able to handle networks with up to 80 nodes in 3 hours. This result is encouraging because it proves that the approach we propose is suitable to be used on networks with sizes of practical interests. A deeper analysis of **Table 1** suggests that model M1 requires much more iterations (cuts (5)) than M2 and M3 to converge to the optimal solution. Analogous considerations can be done for the computation times required to solve the different models. Another important indication emerging from **Table 1** is about the performance degradation as the number of nodes increases. Models M2 and M3 clearly scale up much better than M1.

The most important outcome of **Table 1** is however probably the indication that handling more complex models (M2 and M3) leads to easier problems (from a computational point of view) than the simpler models considers so far in the literature (M1). Basically, M2 and M3 provide better models both from a theoretical (level of realism) and practical (solution times) point of view.

It is finally important to observe that the standard deviation of all the indicators reported in **Table 1** is high. This indicate that the performance of the exact methods we propose tend to be instance-dependent.

6. A Speed-up Strategy

Classic valid inequalities for integer linear models of broadcasting problems on wireless networks can be easily adapted to our problem (we refer the interested reader to [8,20,21]). However, we discovered that a strategy different from (and usually not compatible with) classic valid inequalities is more effective and efficient for the maximum lifetime problems on wireless networks, in a framework like the one we propose.

Notice that in the reminder of the paper we will concentrate, for short, on formulation M2, but all the results can be trivially adapted to both the other models M1 and M3.

The idea exploits a characteristic of the problem: the objective function cost is only determined by the node *i* with the shortest lifetime of the current topology. In this context, it is easy to observe that if all the other nodes of the network transmit at the maximum possible power such that their lifetime is not shorter than that of node *i*,

Table 1. Computational results.

|V|	Solved	LR cuts	(5) added	Total cuts	(5) added	Seconds	
	(over 10)	avg	stdev	avg	stdev	avg	stdev
Model M1							
20	10	69.90	16.78	73.60	17.27	0.30	0.28
30	10	146.40	41043	153.20	36.71	12.79	23.54
40	4	247.50	49.84	278.25	53.56	485.23	658.91
50	3	288.67	27.97	363.33	90.29	1441.73	913.65
60	1	415.00	0.00	415.00	0.00	558.00	0.00
70	0	-	-	-	-	-	-
80	0	-	-	-	-	-	-
Model M2							
20	10	68.90	16.13	73.50	17.15	0.37	0.45
30	10	154.60	52.56	162.10	49.03	115.35	320.05
40	4	242.17	32.88	270.67	39.50	654.64	997.76
50	3	307.50	31.10	365.75	53.54	1287.99	795.31
60	1	408.00	0.00	428.00	0.00	2016.14	0.00
70	0	-	-	-	-	-	-
80	0	-	-	-	-	-	-
Model M3							
20	10	142.90	47.12	142.90	47.12	0.14	0.05
30	10	436.80	143.48	439.80	143.08	2.31	2.06
40	4	949.50	347.31	957.90	338.54	15.78	11.85
50	3	1677.90	457.48	1687.30	463.09	86.02	136.97
60	1	2083.90	994.99	2095.00	988.00	313.39	603.28
70	0	2938.60	930.12	2954.80	933.34	454.97	410.18
80	0	2940.00	652.01	2968.88	645.43	934.44	958.92

the objective function cost does not change, and at the same time the probability of having a violated constraint (5) is dramatically reduced. In this context we have therefore interest in forcing nodes to transmit to their maximum possible power.

In order to achieve our goal, we propose a two level, hierarchical objective function, where the main objective is to maximize the lifetime of the network, and the secondary one is to have nodes to transmit to the highest possible power. The original objective function (11) (that is common to all the models considered in Section 5) is changed to the following one:

$$\min z - \delta \sum_{i \in V} \sum_{j \in V \setminus \{i\}} p_{ij} x_{ij} \qquad (21)$$

Where δ is an arbitrarily small constant. In such a way, for a given value of z, each node i will be assigned the highest possible transmission power (with respect to z). An example is given in **Figure 3(a),** where the nodes of a network are depicted. We assume that all nodes have the same initial battery capacity and that the simple equation (20) is used to measure power requirements. Model M2 is considered for the example. In **Figures 3(b)** and **3(c)** two feasible topologies are depicted. The first one is the optimized solution obtained according to the original objective function (11), while the second one is the optimal topology obtained by considering the hierarchical objective function (21). Notice that the lifetime of both the solutions is the same, and is determined by node 9, which is the first node to run out of battery. The arc representing the transmission power of the first node to fail is in bold (arc (9,1) in our case), while arcs that are obtained by the wireless multicast advantage (*i.e.* they do not correspond to the maximum transmission power of the originating node) are dashed. It is important to observe that the topology of **Figure 3(b)** is contained in that of **Figure 3(c)**. This is a side-effect of the strategy we have applied.

The same experiments summarized in **Table 1** have been replicated with the speed-up strategy in operation, and the results are reported in **Table 2**. The meaning of the columns of **Table 2** is the same as in **Table 1**. In addition the percentage variation, with respect to Table 1, is reported in brackets for each indicator.

Table 2 shows the benefit guaranteed by the speed-up

Table 2. Computational results of the models discussed in Section 5 with the objective function (21). Percentage variations with respect to Table 1 are reported in brackets.

$\|V\|$	Solved	LR cuts (5) added		Total cuts (5) added		Seconds	
	(over 10)	avg	stdev	avg	stdev	avg	stdev
Model M1							
20	10 (+0)	29.80 (-61.5%)	13.78 (-86.9%)	30.00 (-59.2%)	13.58 (-21.4%)	0.12 (-57.4%)	0.04 (-17.8%)
30	10 (+0)	53.70 (-95.9%)	12.22 (-98.7%)	54.10 (-64.7%)	12.38 (-66.3%)	0.53 (-63.6%)	0.17 (-70.5%)
40	10 (+6)	83.90 (-10.1%)	24.37 (+57.3%)	84.20 (-69.7%)	24.61 (-54.1%)	436.49 (-66.1%)	1036.45 (-51.1%)
50	8 (+5)	100.25 (-90%)	23.44 (-64.1%)	100.25 (-72.4%)	23.44 (-74.0%)	144.92 (-65.3%)	327.97 (-16.2%)
60	8 (+7)	166.36 (-49.6%)	34.31 (n.a.)	166.00 (-60.0%)	34.13 (n.a.)	3281.7 (n.a.)	236.80 (n.a.)
70	1 (+1)	188.00 (n.a.)	0.00 (n.a.)	188.00 (n.a.)	0.00 (n.a.)	100.91 (n.a.)	0.00 (n.a.)
80	1 (+1)	167.00 (n.a.)	0.00 (n.a.)	167.00 (n.a.)	0.00 (n.a.)	208.41 (n.a.)	0.00 (n.a.)
Model M2							
20	10 (+0)	29.60 (-71.21%)	13.66 (-94.5%)	29.70 (-59.6%)	13.43 (-21.7%)	0.11 (-57.0%)	0.02 (-15.3%)
30	10 (+0)	57.50 (-99.5%)	14.2 (-99.9%)	58.2 (-64.1%)	14.30 (-70.8%)	0.55 (-62.8%)	0.22 (-73.3%)
40	10 (+4)	81.10 (-0.4%)	16.89 (+28.8%)	81.40 (-69.9%)	17.20 (-56.5%)	652.13 (-66.5%)	1285.33 (-48.6%)
50	9 (+5)	115.22 (-90.9%)	29.52 (-80.2%)	115.89 (-86.3%)	29.34 (-45.2%)	116.81 (-62.5%)	157.53 (-5.1%)
60	5 (+4)	191.80 (-93.6%)	88.13 (n.a.)	191.80 (-55.2%)	88.13 (n.a.)	129.70(-53.0%)	116.89 (n.a.)
70	2 (+2)	91.50 (n.a.)	122.33 (n.a.)	91.5 (n.a.)	122.33 (n.a.)	1085.32 (n.a.)	1473.22 (n.a.)
80	2 (+2)	232.00 (n.a.)	2.83 (n.a.)	232.50 (n.a.)	2.12 (n.a.)	564.02 (n.a.)	459.97 (n.a.)
Model M3							
20	10 (+0)	142.90 (-18.1%)	47.12 (-27.8%)	142.90 (-0.0%)	47.12 (-0.0%)	0.12 (-0.0%)	0.04(-0.0%)
30	10 (+0)	436.80 (-0.2%)	143.48 (+4.1%)	439.80 (-0.0%)	143.08 (-0.0%)	2.31 (-0.0%)	2.14 (-0.0%)
40	10 (+0)	951.00 (-0.4%)	345.95 (-1.0%)	959.40 (+0.2%)	337.10 (-0.4%)	15.73 (+0.2%)	11.73 (-0.4%)
50	10(+0)	1677.90 (-21.9%)	457.48 (-35.4%)	1687.30 (-0.0%)	463.09 (-0.0%)	67.18 (-0.0%)	88.54 (-0.0%)
60	10 (+0)	2245.90 (-2.9%)	805.68 (-4.3%)	2278.10 (+8.7%)	795.68 (-19.5%)	304.33 (+7.8%)	577.26 (-19.0%)
70	10 (+0)	2557.20 (-16.6%)	1232.07 (-8.3%)	2570.80 (-13.0%)	1234.66 (+32.3%)	379.48 (-13.0%)	376.17 (+33.5%)
80	8 (+0)	2980.25 (-1.6%)	670.38 (+14.6%)	3001.13 (+1.1%)	659.96 (+2.3%)	919.26 (+1.4%)	1099.32 (+2.8%)

technique we propose. The advantage taken by models M1 and M2 when the suggested strategy is used is significant with respect to every factor considered, with improvements in the solution times in the order of 60%, and the number of problems solved to proven optimality in the given time is dramatically increased.

Different is the situation for model M3: the results in this case do not show any significant difference with respect to those reported in **Table 1**.

The conclusion is therefore that the speed-up strategy should be adopted when models M1 and M2 are considered, while it its use is not recommended for model M3 (see also Section 7).

7. A practical Drawback and Its Solution

The speed-up strategy we have proposed in Section 6 has a practical drawback, which is clear from the example reported in **Figures 3**: in the optimal solution reported in **Figure 3(c)**, there are many nodes transmitting to a power which is higher than necessary. This is not desirable, since noise and overhearing phenomena on the network increase. Moreover, the network lifetime is more sensitive to erroneous estimations of battery consumptions. In conclusion, all the arcs in **Figure 3(c)** that are not also in **Figure 3(b)** are potentially damaging (and certainly unnecessary). There is therefore a contrast between the desirable solution from a practical, realistic point of view, and that retrieved when the speed-up rule is implemented (notwithstanding they are both optimal from a theoretical point of view). We will see in this section how to overcome this problem.

Notice that a similar problem had already been identified in a very related context in [16]. They proposed an integer linear model for a maximum lifetime problem, and observed that if powers are not constrained to be minimal for each node, it might happen what our strategy forces to happen as a side effect. To overcome what they correctly identifies as a problem, [16] proposed a hierarchic objective function, where the secondary objective is practically the opposite of the one we suggest in Section 6: the authors somehow concentrate on the reality, and not on optimization like we do. Their approach can be adapted to our case by substituting the objective function (11) with the following one:

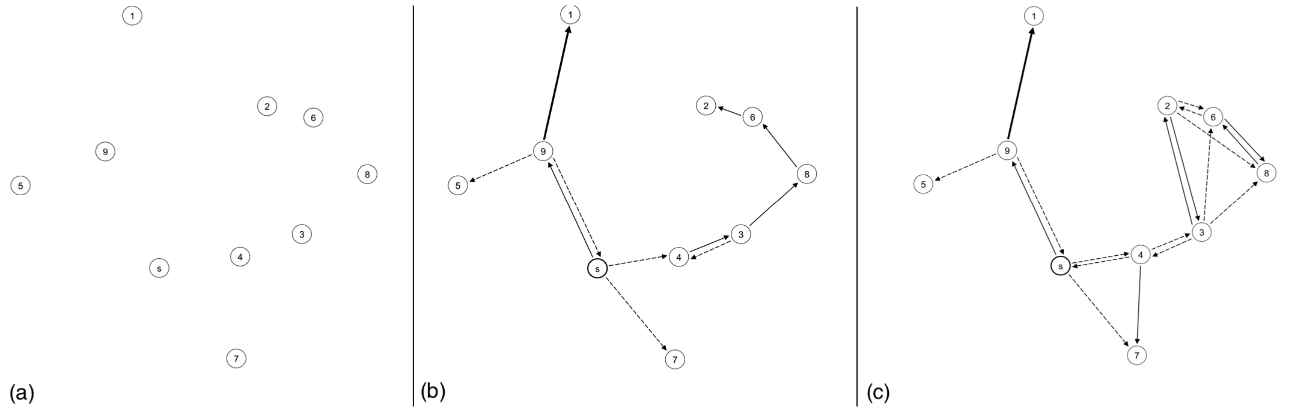

Figure 3. (a) Example of a network ;(b) Topology obtained by solving the original model M2, with the objective function (11), on the network of (a);(c) Topology obtained by solving the modified model M2, with the objective function (21),on the network of (a).

$$\min z + \in \sum_{i \in V} yi \tag{22}$$

where y variables are free variables connected to the rest of the problem through the following constraints:

$$\begin{aligned} yi \geq \frac{\varepsilon_{sc}(i)}{cap_i} + \sum_{j \in V \setminus \{i\}} \frac{(H+D)(\varepsilon_{tr} + \in p_{ij})}{cap_i} x_{ij} \\ + \sum_{\substack{j \in V \setminus \{i\} \\ k \in N_j(i)}} + \frac{(H+D)\varepsilon_{rc}}{cap_i} x_{jk} \quad \forall i \in V \end{aligned} \tag{23}$$

Variable y_i models the lifetime (expressed in terms of number of transmitting cycles, see Section 5.3) of node *i*. Notice once more that we refer to model M2, but it is straightforward to translate it for M1 and M3.

If the adaptation of the method suggested in [16] is applied to the example of **Figure 3**(**a**), the same topology depicted in **Figure 3**(**b**) is obtained. Such an approach clearly goes in the opposite direction of the speed-up strategy discussed in Section 6. It is interesting to notice in [16] the authors did not have any need for a speed-up strategy because they proposed a model, and not an algorithm for efficiently solving it.

In order to overcome the practical problem discussed before, we developed a post-optimization technique, where we shrink an optimal solution obtained by applying the speed-up rule discussed in Section 6. The objective is to increase as much as possible the lifetime of all the nodes of the network.

The strategy we propose does not lead to an optimal solution (like the method suggested in [16] does), but it provides suboptimal solutions in a very short time. The rationale behind a heuristic approach is that the main target of the whole optimization is to maximize the network lifetime, and we achieve this in the first phase (see Section 6). The second phase discussed here is therefore less crucial, and is suitable for a very fast heuristic method like the one we suggest.

The second phase optimization method has a polynomial execution time, and is based on the solution of a minimum cost arborescence problem on the connected structure obtained during phase one. Costs of arcs are represented by power requirements. Tohe algorithm described in [22] is adopted. Once the arborescence has been identified, the transmission powers of the nodes are regulated based on the arborescence, saving as much power as possible. The lifetime of single nodes is therefore increased with respect to the solution obtained in phase one. Notice that the application of the post-optimization procedure to the example of **Figure 3**(**c**) leads to the topology of **Figure 3**(**b**).

The post-optimization phase is run in polynomial time, and has a negligible computation time. Therefore the total execution times of phase one plus phase two remain similar to those presented in **Table 2**. On the contrary, the adaptation of the approach proposed in [6] leads to solving times greater than or equal to those of **Table 1**, which are longer than those of the method we suggest.

8. Conclusions

The aim of this paper was to show that tools like integer linear programming, often regarded as over-theoretical and unrealistic, are indeed suitable frameworks to include the latest advances in energy consumption and communication models in wireless sensor networks. Three models of increasing realism have been presented. Experimental results suggest that integer linear programming can be used not only as an effective modeling tool, but also as an efficient solving method for problems of realistic size. A surprising result also indicates that the easiest models to solve (in terms of computation times) are the most realistic ones, suggesting that they should be preferred in general.

A speed-up technique, based on the characteristics of the problem, has been discussed and experimentally shown to be effective on many of the problems considered. A practical drawback introduced by the speed-up technique has been finally identified and a method to overcome it has been introduced.

9. References

[1] L. Negri, D. Zanetti, R. Montemanni and S. Giordano, "Power-optimized Topology Formation and Configuration in Bluetooth Sensor Networks: An Experimental App- roach," *Ad-Hoc and Sensor Wireless Networks*, Vol. 6, No. 1-2, 2008, pp. 145-175.

[2] W. Ye, J. Heidemann and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," *IEEE Infocom Proceedings*, New York, 23-27 June 2002, pp. 1567-1576.

[3] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk and J. Anderson, "Wireless Sensor Networks for Habitat Monitoring," *Proceedings of the* 1*st ACM international Workshop on Wireless Sensor Networks And Applications ConFerence*, Atlanta, 28 Sepmeber 2002, pp. 88-97

[4] S. Kim, S. Pakzad, D. E. Culler, J. Demmel, G. Fenves, S. Glaser and M. Turon, "Health monitoring of civil infrastructures using wireless sensor networks," *Proceedings of the IEEE Information Processing in Sensor Networks Conference*, Cambridge, 25-27 April 2007, pp. 254-263.

[5] D. M. Doolin and N. Sitar, "Wireless Sensors for Wildfire Monitoring," *Proceedings of the Sensors and Smart Structures Technologies For Civil, Mechanical, and Aerospace Systems Conference*, San Diego, 7-10 March 2005, pp. 477-484.

[6] I. Chlamtac, C. Petrioli and J. Redi, "Energy-Conserving Access Protocols for Identification Networks," *IEEE Transactions on Networking*, Vol. 7, No. 1, 1999, pp. 51-59.

[7] D. Estrin, R. Govindan, J. Heidemann and S. Kumar, "Next Century Challanges: Scalable Coordination in Sensor Networks," *Mobicom Proceedings*, Seattle, 15-19 August 1999, pp. 263-270.

[8] S. Madden, M. J. Franklin, J. M. Hellerstein and W. Hong, "TAG: A Tiny Aggregation Service for Ad-hoc Sensor Networks," *Proceedings of the* 5*th Annual Symposium on Operating Systems Design and Implementation conference*, Boston, 9-11 December 2002, pp. 131-146.

[9] W. Heinzelman, A. Chandrakasan and H. Balakrish- nan, "Energy-Efficient Communication Protocols for Wireless Microsensor Networks," *Proceedings of the* 33*th Hawaii International International Conference on Systems Science*, Hawaii, 4-7 January 2000, p. 8020.

[10] S. Singh and C. S. Raghavendra, "Power Aware Multi-access Protocol with Signalling for Ad Hoc Networks," *Computer Communication Review*, Vol. 25, No. 3, 1998, pp. 5-26.

[11] R. Montemanni, L. M. Gambardella and A. K. Das, "The Minimum Power Broadcast Problem in Wireless Networks: A Simulated Annealing Approach," *Proceedings of the IEEE Wireless and Communications and Networking Con- ference*, New Orleans, 13-17 March 2005, pp. 2057- 2062.

[12] I. Stojmenovic, M. Seddigh and J. Zunic, "Internal Nodes Based Broadcasting in Wireless Networks," *Proceedings of the* 34*th Hawaii International International Conference on Systems Science*, Maui, 3-6 January 2001, p. 9005.

[13] J. Wieselthier, G. Nguyen and A. Ephremides, "On the Construction of Energy-efficient Broadcast and Multicast Trees in Wireless Networks," *Proceedings of the IEEE Infocom conference*, Tel-Aviv, 26-30 March 2000, pp. 585-594.

[14] P. Basu and J. Redi, "Effect of Overhearing Transmissions on Energy Efficiency in Dense Sensor Networks," *Proceedings of the IEEE Information Processing in Sensor Networks Conference*, California, 26-27 April 2004, pp. 196-204.

[15] E. S. Biagioni and G. Sasaki, "Wireless Sensor Placement for Reliable and Efficient Data Collection," *Proceedings of the 36th Hawaii International International Conference on Systems Science*, Hawaii, 6-9 January 2003, p. 127.2.

[16] A. K. Das, M. El-Sharkawi, *et al*., "Maximization of Time-to-first-failure for Multicasting in Wireless Networks: Optimal Solution," *IEEE Milcom Proceedings*, Vol 3, 2004, pp. 1358-1363.

[17] R. Montemanni, "Maximum Lifetime Broadcasting Topologies in Wireless Sensor Networks: Advanced Mathematical Programming Models," *Proceedings of the 42nd Hawaii International International Conference on Systems Science*, Hawaii, 5-8 January 2009.

[18] V. Leggieri, P. Nobili and C. Triki, "Minimum Power Multicasting Problem in Wireless Networks," *Mathematical Methods of Operations Research*, Vol. 68, No. 2, 2008, pp. 295-311.

[19] G. B. Dantzig, D. R. Funkerson and S. Johnson. "Solution of a Large-scale Traveling Salesman Problem," *Journal of the Operations Research Society of America*, Vol. 2, No. 4, 1954, pp. 393-410.

[20] J. Bauer, D. Haugland and D. Yuan, "Analysis and Computational Study of Several Integer Programming Formulations for Minimum-Energy Multicasting in Wireless Ad Hoc Networks," *Networks*, Vol. 52, No. 2, 2008, pp. 57-68.

[21] R. Montemanni and L. M. Gambardella, "Minimum Power Symmetric Connectivity Problem in Wireless Networks: A New Approach," *Proceedings of the Mobile and Wireless Communication Networks Conference*, Paris, 25-27 October, 2004, pp. 496-508.

[22] H. N. Gabow, Z. Galil, T. Spencer and R. E. Tarjan, "Efficient Algorithms for Finding Minimum Spanning Trees in Undirected and Directed Graphs," *Combinatorica*, Vol. 6, No. 2, 1986, pp. 109-122.