Accepted Manuscript

Mixed Integer Formulations for the Probabilistic Minimum Energy Broadcast Problem in Wireless Networks

Roberto Montemanni, Valeria Leggieri, Chefi Triki

PII:	S0377-2217(07)00624-8
DOI:	10.1016/j.ejor.2007.06.031
Reference:	EOR 8516

To appear in: *European Journal of Operational Research*

Received Date:	12 July 2006
Revised Date:	28 March 2007
Accepted Date:	20 June 2007

Please cite this article as: Montemanni, R., Leggieri, V., Triki, C., Mixed Integer Formulations for the Probabilistic Minimum Energy Broadcast Problem in Wireless Networks, *European Journal of Operational Research* (2007), doi: 10.1016/j.ejor.2007.06.031

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

ACCEPTED MANUSCRIPT

Mixed Integer Formulations for the Probabilistic Minimum Energy Broadcast Problem in Wireless Networks

Roberto Montemanni[a,1,2], **Valeria Leggieri**[b], **Chefi Triki**[b,3]

[a] *Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA), Galleria 2, CH-6928 Manno-Lugano, Switzerland*

[b] *Dipartimento di Matematica, Università degli Studi di Lecce, 73100 Lecce, Italy*

Abstract

In this paper we study a new variant of the Minimum Energy Broadcast (MEB) problem, namely the Probabilistic MEB (PMEB). The objective of the classic MEB problem is to assign transmission powers to the nodes of a wireless network is such a way that the total energy dissipated on the network is minimized, while a connected broadcasting structure is guaranteed by the assigned transmission powers. In the new variant of the problem treated in this paper, node failure is taken into account, aiming at providing solutions with a chosen reliability level for the broadcasting structure. Three mixed integer linear programming formulations for the new problem are presented, together with efficient formulation-dependent methods for their solution. Computational results are proposed and discussed. One method emerges as the most promising one under realistic settings. It is able to handle problems with up to fifty nodes.

Keywords: Minimum Energy Broadcasting, Mixed Integer Linear Programming, Ad-Hoc Networks.

ACCEPTED MANUSCRIPT

1 Introduction

Ad-Hoc wireless and sensor networks consist of a set of radio terminals that communicate through wireless channels without relying on any network infrastructure. One terminal can communicate with another one using a single hop if the second terminal is within the transmission range of the first one, otherwise a multi-hop communication is required. In this latter case, the transmission is performed by using some intermediate terminals, that therefore play the role of routers.

A crucial issue in this context consists in assigning a transmission power to each node in order to ensure the connectivity of the network while minimizing the total power expenditure. Determining the optimal transmission power for each node is, indeed, desirable since a high power value will achieve a wide transmission range and therefore reach many nodes via a direct link, but at the same time will require higher transmission power and will increase the interference level. On the other hand, low energy value may isolate one or more nodes

[1]Corresponding author. Tel +41 58 666 666 7, fax +41 58 666 666 1, email: roberto@idsia.ch.
[2]Partially through grant 200020-109854/1 from NSF, Switzerland.
[3]Partially supported through grant 2003014039 from MIUR, Italy.

ACCEPTED MANUSCRIPT

causing the network to be disconnected. We consider here the broadcast case, in which a designated source terminal has to communicate with all the other active nodes, and we work on a static network, i.e. distances among terminals are known in advance, together with the characteristics of the environment in which the terminals are operating.

The Minimum Energy Broadcast (MEB) problem and its variants have already been the subject of many works. Clementi et al. have shown its NP-hardness in [3]. Althaus et al. have proposed a mixed integer linear programming model and have developed a branch and bound approach for its solution [1]. Alternative formulations have been suggested and solved to optimality by Das et al. in [5]. Montemanni and Gambardella have proposed in [10] two mixed integer programming formulations together with a preprocessing rule and some valid inequalities. Several heuristic methods have been also proposed in the literature. Wieselthier et al. have developed in [4] the well known BIP (Broadcast Incremental Power) algorithm. Metaheuristic approaches have been suggested by Marks et al. in [6], Das et al. in [4], Montemanni et al. [9] and Li et al. in [7]. More recently, Lagrangian relaxation procedures have been proposed by Altinkemer et al. in [2] and by Yuan in [11].

However, all the above works deal with deterministic models for the MEB problem and none of them has considered node realiability. This assumption represents a distorted approximation of reality; the terminals are, indeed, electronic devices that may be subject to temporary damage or a permanent failure. This remark suggests the appropriateness of solving the problem as an optimization problem that takes into account the uncertain nature of nodes availability. This is a salient characteristic that makes the problem much more complex to solve than its classic, fully deterministic counterpart.

To the best of our knowledge, no mathematical models explicitly incorporating the uncertain availability of the nodes have been proposed so far. This paper attempts to provide an original contribution in this direction. More specifically, it presents formulations for a variant of the MEB problem in which nodes failure is taken into account, and the optimal solution not only minimizes the total transmission power over the network, but also guarantees a certain reliability level for the whole network. The rationale is that in reality one implicitly accepts that failures will happen in the terminals deployed in the environment, and therefore the aim of the PMEB problem is to provide broadcasting structures robust enough to guarantee, in case of failure of some terminals, a reliable connectivity for the remaining terminals.

2 Network Model

The mathematical formulation of the MEB problem can be given in graph theoretical terms using the notation already adopted, e.g. in [10]. Let $G = (V, A)$ be a directed complete graph, where V represents the set of nodes corresponding to the terminals of the network and A is the set of arcs containing all the possible pairs (i, j), with $i, j \in V$ and $i \neq j$.

ACCEPTED MANUSCRIPT

A cost p_{ij} is associated with each arc $(i,j) \in A$. It corresponds to the power required to establish a link from node i to j. In this paper a simple signal propagation model is used. The power required for a signal originated at node i to be received at node j is defined as $p_{ij} := (d_{ij})^\kappa$, where d_{ij} is the distance between nodes i and j, and κ is an environment-dependent coefficient (typically between 2 and 5). Notice that the results presented in this paper remain valid also in case more complex signal propagation models are considered.

Since we consider wireless networks where nodes are equipped with omnidirectional antennae, the following important broadcasting property is valid: if node i is transmitting directly (single-hop transmission) to node j, each other node k such that $p_{ik} \leq p_{ij}$ will be also reached by the signal transmitted by node i. This property is at the basis of the power-optimized topologies we propose.

The MEB problem can be thus formalized by introducing the notion of *range assignment* defined as a function $r : V \to I\!R^+$ that associates to any $i \in V$ the transmission power assigned to node i. A transmission link from nodes i to node j is established under the range assignment r if $r(i) \geq p_{ij}$. The MEB is therefore the problem of defining a range assignment r minimizing $\sum_{i \in V} r(i)$, subject to the constraints that a directed path exists from the source node s to all the other nodes in the network.

Another definition of the MEB problem can be given in terms of optimal arborescence rooted at node s: for a node i and an arborescence T of G, let (i, i_T) be the maximum cost arc originated from i in T, i.e. $(i, i_T) \in T$ and $p_{ii_T} \geq p_{ij}$, for all $(i,j) \in T$. Due to the broadcasting property, the *power cost* of an arborescence T is then $c(T) = \sum_{i \in V} p_{ii_T}$. It is now easy to observe that an arborescence rooted at s is contained (not necessarily strictly contained) in any valid broadcasting structure. The MEB problem can therefore be described as the problem of finding the arborescence T with the minimum power cost $c(T)$.

In reality, some nodes of the network may fail due to technical problems or battery draining. In the present paper we aim at presenting a model where a concept of reliability, connected with node failures, is taken into account. It is important to stress that it is not our aim to model situations where nodes temporarily switch off (sleeping mode) in order to save battery. This aspect, which is however an important issue, is beyond the scope of this work, and deserves future research.

In order to consider node failures, we associate with each node i of the network a value $q_i \in (0, 1]$ representing the probability that node i will remain active (i.e. it does not fail) for the whole operating time of the network. The value of q_i has to be assigned by the decision makers, and reflects the reliability of each node. Typically it will depend on the physical characteristics of the area where each node is deployed. For example, in military applications a node i close to the enemy will have a high probability to be destroyed, and should consequently have a small value of q_i. Based on the same idea, a node i deployed in an impervious territory (e.g. a pluvial forest) will have again a small value for q_i.

ACCEPTED MANUSCRIPT

We can now formally define the Probabilistic Minimum Energy Broadcast (PMEB) problem as a MEB problem where a given minimum reliability level $\alpha \in (0, 1]$ has to be achieved. Specifically, the reliability level of the multi-hop transmissions from s to each other node i of the network will have to be at least α.

The uncertain events characterizing our problem (i.e. node failures) are independent from each other, that is, if a node happens to fail, this does not affect the correct functioning of the other terminals of the network. Therefore the probabilities q_i can be regarded as independent from each other. It is also possible to observe that if nodes i and j have a probability values of functioning q_i and q_j respectively, then link (i, j) has a probability value of being available equal to the product $q_i q_j$. The same reasoning can be extended to paths: the probability of a multi-hop transmission (path) from node i to node j of being active is equal to the product of the probabilities q_v associated with the nodes involved (i.e. belonging to the path itself). In mathematical terms we have $\mathcal{P}(P_{ij}) := \prod_{v \in P_{ij}} q_v$, where P_{ij} represents the path connecting i to j under investigation, and $\mathcal{P}$ is the probability function.

3 Mixed integer linear programming formulations

Even though the PMEB problem consists in assigning transmission powers to the nodes of the network, link states will be used as decision variables. More specifically, we define continuous variables y containing the transmission power of each node, i.e. $y_i := r(i)$ for each $i \in V$, and boolean variables z, that describe the optimal arborescence structure: $z_{ij} := 1$ if $(i, j) \in T$ and 0 otherwise, where T represents the arborescence connecting the source s with all the other nodes of the network.

3.1 Path-Based formulation

Let $\mathcal{U}$ represent the set of all infeasible paths originated in s. The generic element P of $\mathcal{U}$ verifies the condition that the product of the probabilities of the nodes involved in path P is less than the reliability level α, i.e. $\mathcal{U} := \{P : P \text{ is an } s - k \text{ path for } k \in V, \text{ s. t. } \prod_{i \in P} q_i < \alpha\}$. The first mixed integer linear programming formulation F_1 we propose for the PMEB is as follows:

$$
\begin{aligned}
& min \sum_{i \in V} y_i & & (1)\\
& s.t. \;\; y_i \geq p_{ij} z_{ij} & \forall (i,j) \in A & \quad (2)\\
& \sum_{\substack{(i,j) \in A, \\ i \in S,\, j \in V \setminus S}} z_{ij} \geq 1 & \forall S \subset V, \; s \in S & \quad (3)\\
& \sum_{(i,j) \in P} z_{ij} \leq |P| - 1 & \forall P \in \mathcal{U} & \quad (4)\\
& z_{ij} \in \{0, 1\} & \forall (i,j) \in A & \quad (5)\\
& y_i \in I\!R^+ & \forall i \in V. & \quad (6)
\end{aligned}
$$

ACCEPTED MANUSCRIPT

Constraints (2) establish the relation between variables z and y. Constraints (3) represent the connectivity requirements: for each proper subset S of V there is at least one outgoing arc. Denoting by $|P|$ the number of arcs belonging to path P, inequalities (4) ensure the reliability constraints across all the source-destination paths. Constraints (5) and (6) define the domain of y and z variables.

Algorithm. The drawback of formulation F_1 is represented by the sets of constraints (3) and (4) that are in an intractable number, from a practical point of view. However, since only a small fraction of these constraints is saturated at optimality, we choose to solve the problem by the means of an iterative approach. Namely, constraints (3) and (4) are initially not considered, and a subset of them will be inserted into the formulation only in case the current optimal solution violates them. The algorithm stops when no violated constraint (3) or (4) exists.

Pseudo-code:

Step 1: Let F_1' be formulation F_1 for problem $PMEB$ without constraints (3) and (4);
Step 2: Solve F_1', and let $(\overline{y}, \overline{z})$ be the optimal solution;
Step 3: If $\overline{z}$ violates a constraint ctr_3 of type (3), then add ctr_3 to F_1' and **go to Step 2**;
Step 4: If $\overline{z}$ violates a constraint ctr_4 of type (4), then add ctr_4 to F_1' and **go to Step 2**;
Step 5: $(\overline{y}, \overline{z})$ is the optimal solution of F_1 (and not only of F_1'). **Stop**.

Notice that the procedure converges after a limited number of iterations since the number of inequalities (3) and (4) is finite.

It is important to observe that a speed-up may be obtained by first considering the linear relaxation of F_1' in Step 3, and adding the corresponding violated constraints of type (3). In this way, many of the constrains might be added before considering the (more time consuming) integer program F_1'. In Section 4 some results that confirm the correctness of this idea will be presented.

Separation of inequalities (3). Once a solution $(\overline{y}, \overline{z})$ of F_1' is available, the presence of violated inequalities of type (3) of F_1 not inserted into F_1' is detected as follows. We use a set M containing marked nodes. Initially M will contain node s only. The set is iteratively expanded by adding, at each iteration, each node j such that $\overline{z}_{ij} = 1$, with $i \in M$ and $j \in V \backslash M$. The iterative procedure is stopped when an iteration terminates without expansions (notice that a maximum of $|V| - 1$ iterations is possible). At this point, if $|M| = |V|$, no violated constraint of type (3) exists in the current solution $(\overline{y}, \overline{z})$. Otherwise, if $|M| < |V|$, a violated constraint of type (3) has been identified. Therefore we can add the violated inequality $\sum_{i \in M, j \in V \backslash M} z_{ij} \geq 1$ to F_1'.

ACCEPTED MANUSCRIPT

Separation of inequalities (4). Once a solution $(\overline{y}, \overline{z})$ of F_1' is available, the presence of violated inequalities of type (4) of F_1 not inserted into F_1' is detected as follows. Since variables $\overline{z}$ define an arborescence (no violated constraint of type (3) exists because of the structure of the algorithm), it is enough to calculate, for each $k \in V \backslash S$, the value $R_{sk}^{\overline{z}} := \prod_{i \in P_{sk}^{\overline{z}}} q_i$, where $P_{sk}^{\overline{z}}$ is the set of nodes encountered along the (sole) path from s to k on the arborescence defined by variables $\overline{z}$. We visit the arborescence defined by variables $\overline{z}$, and as soon as we identify a path from s to k (with k possibly not a leaf) with $R_{sk}^{\overline{z}} < \alpha$, we add the constraint of type (4) corresponding to $P_{sk}^{\overline{z}}$ to model F_1'. More than one constraint (4) can be added at each iteration.

3.2 Cumulative Probability formulation

The idea behind our second PMEB model is to get rid of $\mathcal{U}$ introducing a new variable associated with each node k of the network expressing the probability value accumulated till that node along the arborescence. Such a variable can be defined as the product of the probability values of the nodes along the $s-k$ path. Instead, in order to have a linear model, in formulation F_2 we use a continuous variable τ_k, for $k \in V$, equivalently defined as the sum of the logarithm of the probability values of the nodes along an $s-k$ path. The model F_2 is as follows:

$$\begin{aligned}
& min \sum_{i \in V} y_i && && (7)\\
& s.t.\ \ y_i \geq p_{ij} z_{ij} && \forall (i,j) \in A && (8)\\
& \sum_{\substack{(i,j) \in A,\\ i \in S,\, j \in V \setminus S}} z_{ij} \geq 1 && \forall S \subset V,\ s \in S && (9)\\
& \tau_i \leq \tau_j + \log q_i - (\log \alpha + \log q_i)(1 - z_{ji}) && \forall (i,j) \in A && (10)\\
& \tau_s = \log q_s && && (11)\\
& z_{ij} \in \{0, 1\} && \forall (i,j) \in A && (12)\\
& y_i \in I\!R^+ && \forall i \in V. && (13)\\
& \log \alpha \leq \tau_i \leq 0 && \forall i \in V && (14)
\end{aligned}$$

For each arc $(i,j) \in A$ constraint (10) updates, through a recursive process, the value of τ_i whenever node i is reached directly from node j. Clearly, such a constraint should be meaningful only if arc (i,j) belongs to the arborescence otherwise it should become redundant. This is guaranteed by the coefficient $-(\log \alpha + \log q_i)$. Constraint (11) initializes the recursive process by assigning $\log q_s$ to τ_s. The set of constraints (14) imposes the reliability requirement on each node of the network and defines the domain of τ_i variables.

Algorithm. With respect to the algorithm for the Path-Based formulation, only constraints (9) are dynamically separated when violated here.
Pseudo-code:

ACCEPTED MANUSCRIPT

Step 1: Let F_2' be formulation F_2 for problem $PMEB$ without constraints (9);
Step 2: Solve F_2', and let $(\overline{y}, \overline{z}, \overline{\tau})$ be the optimal solution;
Step 3: If $\overline{z}$ violates a constraint ctr_9 of type (9), then add ctr_9 to F_2' and **go to Step 2**;
Step 4: $(\overline{y}, \overline{z}, \overline{\tau})$ is the optimal solution of F_2 (and not only of F_2'). **Stop**.

A theoretical speed-up for the method might be obtained by considering first the linear relaxation of F_2' in Step 3. However, the experiments reported in Section 4, suggest that this is not the case.

3.3 Multicommodity Flow formulation

The formulation presented in this section is based on a multicommodity flow model as described, for example, in [8]. It includes into the model an explicit representation of all the paths connecting the source s to each node $d \in V$. For this goal, we decline the use of the spanning arborescence variables z and we introduce, for each node $d \in V$ and each arc $(i, j) \in A$, a new variable denoted by t_{ij}^d that takes value 1 if arc (i, j) is on the path $s - d$, and 0 otherwise. Variables y remains, and have the same meaning as in Sections 3.1. Defining $e_i := \sum_{j \in V,\, j \neq i} t_{ij}^d$, the model F_3 is as follows:

$$min \sum_{i \in V} y_i \tag{15}$$

$$s.t.\ \ y_i \geq p_{ij} t_{ij}^d \qquad \forall (i,j) \in A, \forall d \in V \setminus \{s\} \tag{16}$$

$$\sum_{j \in V,\, j \neq s} t_{sj}^d = 1 \qquad \forall d \in V \setminus \{s\} \tag{17}$$

$$\sum_{i \in V,\, i \neq d} t_{id}^d = 1 \qquad \forall d \in V \setminus \{s\} \tag{18}$$

$$\sum_{i \in V,\, i \neq j} t_{ij}^d - \sum_{i \in V,\, i \neq j} t_{ji}^d = 0 \quad \forall d \in V \setminus \{s\},\ \forall j \in V \setminus \{s, d\} \tag{19}$$

$$q_d \prod_{i \in V} q_i^{e_i} \geq \alpha \qquad \forall d \in V \setminus \{s\} \tag{20}$$

$$y_i \in I\!R^+ \qquad \forall i \in V. \tag{21}$$

$$t_{ij}^d \in \{0, 1\} \qquad \forall (i,j) \in A,\ \forall d \in V \setminus \{s\}. \tag{22}$$

Constraints (16) regulates the power emitted by node i based on the value of t variables. The sets of constraints (17)–(19) are the usual flow equations on the source node, on all the destination nodes, and on any intermediate node, respectively. Constraints (20) are the reliability requirements, and constraints (22) are the limitations on the decision variables. The Multicommodity Flow formulation F_3 is a non-linear programming model because of the presence of the set of constraints (20). Such constraints could be, however, linearized by making use of the following logarithmic properties: $\log \left(q_d \prod_{i \in V} q_i^{e_i} \right) = \log q_d + \sum_{i \in V} \log \left(q_i^{e_i} \right) = \log q_d + \sum_{i,j \in V,\, j \neq i} t_{ij}^d \log q_i$. (We remind the reader that $e_i = 1$ if i is on the active path from s to d.) Constraints (20) can be, thus, replaced by the following linear constraints:

$$\log q_d + \sum_{i,j \in V,\, j \neq i} t_{ij}^d \log q_i \geq \log \alpha \qquad \forall d \in V \setminus \{s\}. \tag{23}$$

ACCEPTED MANUSCRIPT

Table 1: Average number of constraints generated while solving the Path-Based formulation F_1 and the Cumulative Probability formulation F_2.

$\|V\|$	α	Path-Based F_1 (3)	(4)	Cumulative Probability F_2 (9)	$\|V\|$	α	Path-Based F_1 (3)	(4)	Cumulative Probability F_2 (9)
10	0.50	2.75	0.50	0.00	15	0.50	11.50	0.60	0.00
10	0.60	7.75	4.10	0.00	15	0.60	42.75	44.60	0.00
10	0.70	28.50	39.60	0.00	20	0.50	17.00	23.00	0.00
10	0.80	94.00	46.60	0.00	20	0.60	43.75	82.40	0.00

Algorithm. Formulation F_3 can be directly attacked by any mixed integer linear programming solver.

4 Experimental Results

This section presents the computational experience carried out with the exact methods described in Section 3. We investigate how many constraints of type (3), (4) and (9) are generated during the execution of the methods based on formulation F_1 and F_2. Then we evaluate the computation times of the three methods, aiming at estimating the largest problem which is possible to solve, and to understand which is the most promising approach, depending on the characteristics of the problem under investigation.

The three methods described in Section 3 have been implemented in C and the experiments have been carried out on an Intel Celeron 1.3 GHz / 256 MB machine. The callable library version of CPLEX[4] 9.0 has been used as mixed integer programming solver. Ten random instances have been generated for each problem considered, and a maximum computation time of 3600 seconds has been allowed for each instance.

Benchmark description. No benchmark is available from the literature, being the problem treated here for the first time. We have therefore generated a set of random instances, trying to produce realistic scenarios. The nodes have been chosen uniformly in a 5000×5000 grid and the probability that any of the nodes is functioning is assumed to be uniformly distributed in the interval $[0,85\,;\,0,95]$. These values should be reasonable for real-life applications. Moreover, the value of the coefficient κ, which models signal propagation, has been set to 2.

Number of constraints added (iterative methods). In Table 1 we present, for some problems, the average number of constraints (3) and (4) generated while solving the Path-Based formulation F_1 as described in Section 3.1, and the average number of constraints (9) generated while solving the Cumulative Probability formulation F_2 as described in Section 3.2.

[4]http://www.cplex.com.

ACCEPTED MANUSCRIPT

Table 2: Computational results for the methods based on the three formulations discussed in Section 3.

$\|V\|$	α	**Path–Based F_1**			**Cumulative Probability F_2**			**Multicommodity Flow F_3**		
		T (sec)	σ (sec)	OOT	T (sec)	σ (sec)	OOT	T (sec)	σ (sec)	OOT
10	0.50	**0.58**	0.71	-	4.67	10.92	-	1.56	1.54	-
10	0.60	**1.55**	2.14	-	3.11	5.71	-	2.26	2.08	-
10	0.70	41.38	52.21	-	14.67	18.71	-	**0.46**	0.70	-
10	0.80	309.84	536.32	-	54.55	51.43	-	**0.05**	0.04	-
15	0.50	**4.64**	3.02	-	65.68	91.15	-	58.10	28.78	-
15	0.60	237.11	540.40	-	459.35	593.40	-	**109.66**	80.38	-
15	0.70	2338.25	1558.93	5	2097.16	1580.71	4	**4.02**	5.56	-
15	0.80	-	-	10	2935.27	1330.43	8	**0.074**	0.01	-
20	0.50	**365.95**	626.78	-	2017.29	1630.16	5	2863.30	952.29	5
20	0.60	**2032.40**	1555.29	5	2710.02	1367.56	7	2267.56	1370.81	5
20	0.70	3364.93	964.74	9	3269.61	991.38	9	**93.72**	200.91	-
20	0.80	-	-	10	-	-	10	**0.21**	0.01	-
25	0.70	-	-	10	-	-	10	**949.33**	1240.36	1
25	0.80	-	-	10	-	-	10	**0.42**	0.02	-
30	0.70	-	-	10	-	-	10	**1809.67**	1791.00	5
30	0.80	-	-	10	-	-	10	**0.78**	0.05	-

From Table 1 it can be observed how, during the solution of formulation F_1, a considerable number of constraints (3) and (4) is generated. Moreover, a weak correlation seems to exist among the number of constraints generated for the two families. This result suggests that a speed-up for the solution method described in Section 3.1 may be obtained by considering the linear relaxation of F_1 for the generation of constraints (3) (as suggested in Section 3.1). More interesting is the situation for constraints (9) generated while solving formulation F_2: none of these constraints is generated during the experiments summarized in Table 1. This result suggests that considering the linear relaxation of F_2 first, to generate constraints (9) in the algorithm discussed in Section 3.2, is not likely to speed up the method.

Computation times. Execution times of the algorithms discussed in Section 3 are summarized in Table 2. For each method, for each problem and for each reliability level considered, we report the average T and standard deviation σ for the execution time (in seconds) and the number of instances not solved to optimality in the given time limit (OOT, out of time). When not all the problems are solved to optimality, only the instances solved to optimality concur to the calculation of T and σ.

From the results reported in Table 2, the exact method based on the Path-Based formulation F_1 appears to be the most efficient approach for small networks (i.e. with at most 15 nodes) and for low values of the reliability threshold α. On the other hand, as the value of α increases, the approach based on the Multicommodity Flow formulation F_3 outperforms by far the other methods, reaching the point of becoming the only method able to solve many of the large problems in the given time limit.

It is also interesting to observe how, for most of the problems, the average computational time required to solve the Multicommodity Flow model F_3 decreases as the value of α

ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

Table 3: Additional computational results for the Multicommodity Flow formulation F_3.

$\|V\|$	α	T	OOT	$\|V\|$	α	T	OOT	$\|V\|$	α	T	OOT
25	0.75	2.923	-	35	0.75	90.59	-	45	0.75	1810.50	3
30	0.75	47.47	-	40	0.75	936.22	2	50	0.75	2788.13	5

increases.

A final remark is about the potential speed-up for the method based on model F_1, based on linear relaxation as discussed in Section 3.1. Even if such a speed-up is likely to exist, it would definitely not close the gap between the performance of the methods based on F_1 and F_3 for the problems where the latter is the fastest method, i.e. when α is large. In these problems, many (even short) paths are automatically discarded by constraints (23), and therefore F_3 becomes extremely easy to solve.

This attractive performance of the Multicommodity Flow model F_3 suggests to test it on larger problems. Indeed, Table 3 summarizes the average computational times (and number of instances not solved to optimality) for test problems with up to 50 nodes and $\alpha = 0.75$. The results show how both the computational times T and the number of instances not solved within the required amount of time OOT increase quite drastically as the number of nodes increases. This is related to the explosion in size of formulation F_3. Nevertheless, the method based on model F_3 remains the only one, among those considered, able to handle problems with up to 50 nodes in the given time.

5 Conclusions

In this paper we have introduced and studied the minimum broadcast problem for Ad-Hoc wireless networks in probabilistic settings, where the possible failure of any node in the network is taken into account, and a given level of reliability has to be achieved. Three formulations of the problem have been proposed, together with algorithms. Experimental results, carried out on instances with up to 50 nodes, suggest that one method dominates the other two when reasonable reliability levels are considered.

6 Acknowledgments

The authors are indebted to the reviewers for their helpful comments and constructive suggestions.

ACCEPTED MANUSCRIPT

ACCEPTED MANUSCRIPT

References

[1] E. Althaus, G. Călinescu, I.I. Măndoiu, S. Prasad, N. Tchervenski, and A. Zelikovsky. Power efficient range assignment in ad-hoc wireless networks. In *Proceedings of the IEEE WCNC'03*, pages 1889–1894, 2003.

[2] K. Altinkemer, F. S. Salman, and P. Bellur. Solving the minimum energy broadcasting problem in ad hoc wireless networks by integer programming. In *Proceedings of the INOC 2005*, pages B2.635–B2.642, 2005.

[3] A. Clementi, P. Penna, and R. Silvestri. On the power assignment problem in radio networks. Technical Report TR00-054, Electronic Colloquium on Computational Complexity (ECCC), 2000.

[4] A.K. Das, R.J. Marks II, M. El-Sharkawi, P. Arabshahi, and A. Gray. The minimum power broadcast problem in wireless networks: an ant colony system approach. In *Proceedings of the IEEE Workshop on Wireless Communications and Networking*, 2002.

[5] A.K. Das, R.J. Marks, M. El-Sharkawi, P. Arabshani, and A. Gray. Minimum power broadcast trees for wireless networks: integer programming formulations. In *Proceedings of the IEEE INFOCOM 2003 Conference*, 2003.

[6] R.J. Marks II, A.K. Das, M. El-Sharkawi, P. Arabshani, and A. Gray. Minimum power broadcast trees for wireless networks: optimizing using the viability lemma. In *Proceedings of the IEEE ISCAS 2002*, 2002.

[7] F. Li, S. Mannor, and A. Lippman. Probabilistic optimization for energy-efficient broadcast in all-wireless networks. In *Proceedings of the 2005 Conference on Information Sciences and Systems*, 2005.

[8] T.L. Magnanti and L. Wolsey. Optimal trees. In *Network Models, Handbook in Operations Research and Management Science* (M.O. Ball et al. eds.), volume 7, pages 503–615. North-Holland, 1995.

[9] R. Montemanni, L.M. Gambardella, and A.K. Das. The minimum power broadcast tree problem in wireless networks: a simulated annealing approach. In *Proceedings of the IEEE WCNC 2005*, 2005.

[10] R. Montemanni and L.M. Gambardella. Exact algorithms for the minimum power symmetric connectivity problem in wireless networks. *Computers and Operations Research*, 31(10):1667–1680, 2004.

[11] D. Yuan. An integer programming approach for the minimum-energy broadcast problem in wireless networks. In *Proceedings of the INOC 2005*, pages B2.643–B2.650, 2005.

ACCEPTED MANUSCRIPT