SWARM APPROACH FOR A CONNECTIVITY PROBLEM IN WIRELESS NETWORKS

Roberto Montemanni, Luca Maria Gambardella

Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA)
Galleria 2, CH-6928 Manno-Lugano, Switzerland
{roberto, luca}@idsia.ch

ABSTRACT

We consider the problem of assigning transmission powers to the nodes of a wireless network in such a way that all the nodes are connected by bidirectional links and the total power consumption is minimized.

Since no central authority (with a global vision of the network) exists in wireless networks, only distributed, swarm approaches can be used. We present a distributed protocol that embeds well-known centralized techniques for power minimization, here used in a local, distributed fashion. The result can be seen as a complex adaptive system (the global network), where global optimization emerges as a result of the behavior of local nodes, each one carrying out a myopic, local optimization.

Computational results, proving the effectiveness of the new protocol, are finally presented.

1. INTRODUCTION

Wireless sensor networks have received significant attention in recent years due to their potential applications in battlefield, emergency disasters relief, and other application scenarios (see, for example, [2], [3], [5], [7], [11], [12], [14] and [15]). Unlike wired networks of cellular networks, no wired backbone infrastructure is installed in wireless sensor networks. A communication session is achieved either through single-hop transmission if the recipient is within the transmission range of the source node, or by relaying through intermediate nodes otherwise.

We consider wireless networks where individual nodes are equipped with omnidirectional antennae. Typically these nodes are also equipped with limited capacity batteries and have a restricted communication radius. Topology control is one of the most fundamental and critical issues in multi-hop wireless networks which directly affects the network performance. In wireless networks, topology control essentially involves choosing the right set of transmitter power to maintain adequate network connectivity. In energy-constrained networks where replacement or periodic maintenance of node batteries is not feasible, the issue is all the more critical since it directly impacts the network lifetime.

The work was partially supported by the Future & Emerging Technologies unit of the European Commission through project "BISON: Biology-Inspired techniques for Self Organization in dynamic Networks" (IST-2001-38923) and by the Swiss National Science Foundation through project "Approximation Algorithms for Machine Scheduling trough Theory and Experiments" (200021-100539).

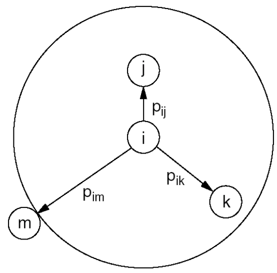

Figure 1. Communication model.

For a given set of transmitters spatially located in the network's area (nodes), the *minimum power topology* (MPT) *problem* is to assign transmission powers to the nodes of the network in such a way that all the nodes are connected by bidirectional links and the total power consumption over the network is minimized. Having bidirectional links simplifies one-hop transmission protocols by allowing acknowledgement messages to be sent back for every packet (see [1]). It is assumed that no power expenditure is involved in reception/processing activities.

Unlike in wired networks, where a transmission from i to m generally reaches only node m, in wireless sensor networks with omnidirectional antennae it is possible to reach several nodes with a single transmission (this is the so-called *wireless multi-cast advantage*, see [16]). In the example of Figure 1 nodes j and k receive the signal originated from node i and directed to node m because j and k are closer to i than m, i.e. they are within the transmission range of a communication from i to m. This property is used to minimize the total transmission power required to connect all the nodes of the network.

Some exact algorithms, all based on mixed integer programming formulations of the problem, have been recently proposed for the (unrealistic) centralized version of the problem, where all information about network topology are available to a central authority. We refer the interested reader to [11] for an overview, comprehensive of theoretical and experimental comparison, of these methods. Turning into real world, it is very unlikely that a centralized authority with global knowledge of the network exists. For this reason distributed protocols, i.e. protocols that run at each node of the network, with a partial knowledge of the network - namely the set of neighbors of each node - have to be developed.

Some distributed protocols that guarantee full connectivity while minimizing the number of neighbors of each node (an indirect measure of the required transmission power) have been proposed in the literature (see [4], [6]). The aim of this paper is to extend these protocols in order to preserve connectivity, while directly optimizing the global transmission power over the network.

The new methodology we propose is somehow novel. The approach to distributed optimization problems has been always approached almost completely by studying the distributed version of the problem itself, without taking into account progresses already developed for the centralized version of the same problem.

On the other hand, many distributed problems, like the one treated in this paper, can be tackled by starting from a different point of view. Often for these problems efficient algorithms for the centralized version are known. We believe that in these circumstances it is often possible to directly transfer centralized algorithms to the distributed case (eventually, like in our case, with some small adjustments).

In our approach a centralized algorithm is run locally at each node of the network, on a subproblem of the whole problem given by the local knowledge of the network available at the node. It also communicates its results to its neighbors, that will use them. Following this approach, each node ends up carrying out a non-egoistic, but somehow myopic, optimization.

The same idea can be applied, with some adjustments, to other distributed problems in wireless networks that have a centralized counterpart, e.g. the minimum power broadcast problem and the channel assignment problem.

It is finally important to observe that in the context we have lined up, very effective algorithms can be used locally (like exact algorithms), since problems are generally of small size, representing a local view.

2. PROBLEM DESCRIPTION

As stated above, it is very unlikely that a central authority with full knowledge about the topology of the network exists. Nevertheless, to formally define the problem and the optimization target, it is useful to imagine that it exists (the central authority can be seen as an oracle).

To represent the MPT problem in mathematical terms, a model for signal propagation has to be selected. We adopt the model presented in [13], and used for most of the work appeared in the literature (see, for example, [1], [10] and [11]). According to this model, signal power falls as $\frac{1}{d^\kappa}$, where d is the distance from the transmitter to the receiver and κ is a environment-dependent coefficient, typically between 2 and 4. Under this model, and adopting the usual convention that every node has the same transmission efficiency and the same detection sensitivity threshold, the power requirement for supporting a link from node i to node j, separated by a distance d_{ij}, is then given by

$$p_{ij} = (d_{ij})^\kappa \tag{1}$$

The MPT problem can be formally described as follows. Given the set V of the nodes of the network, a *range assignment* is a function $r : V \rightarrow \mathcal{R}^+$. A *bidirectional link* between nodes i and j is said to be established under the range assignment r if $r(i) \geq p_{ij}$ and $r(j) \geq p_{ij}$. Let now $B(r)$ denote the set of all bidirectional links established under the range assignment r. The MPT problem is the problem of finding a range assignment r minimizing $\sum_{i \in V} r(i)$, subject to constraints on minimum and maximum transmission powers and to the constraint that the graph $(V, B(r))$ must be connected.

3. DISTRIBUTED PROTOCOLS

We now go back to the realistic situation where no central authority exists, and each node has only a local view of the network. To solve the minimum power connectivity problem in these conditions, distributed protocols that run locally at each node represent the only possible alternative.

In [4] a detailed study showing that there is a very close correlation between the (minimum) number of neighbors of the nodes of a network and the probability of the network to be fully connected is conducted. In particular they observed that this indicator (number of neighbors) is more interesting than a threshold on transmission power when connectivity issues are studied. Following this observation they propose a simple protocol able to provide full connectivity (with high probability) with a much smaller total transmission power expenditure than methods working directly on global thresholds on power.

This protocol will be here extended in order to locally optimize transmission powers while maintaining the good theoretical properties of the original protocol. The original protocol is sketched in Section 3.1, while the new extended version is presented in Section 3.2.

3.1. Protocol MLD ([4])

The MLD (*Minimum Link Degree*) protocol, originally proposed in [4], works as follows. Each node, starting from 0, increases its transmission power by a small amount once it has not reached a minimum number of neighbors (link degree) ngb. Whenever another node, which so far does not belong to the neighborhood list, hears the hello message of the original node for the first time, it realizes that the latter has too few neighbors, either sets its power equal to the transmission power of the hello-sending node or leaves it as before, whichever is larger, and answers the hello message. Now the original and new node are able to communicate back and forth and have established a new link. The original node adds one new node to its neighborhood list. Only once the required minimum link degree is reached, the original node stops increasing its power for its hello transmissions. At the end each node has at least ngb neighbors. Some have more because they have been forced to answer nodes too low in ngb; their transmission power is larger than necessary to obtain only ngb neighbors for themselves.

In [4] it is shown that small values of parameter ngb (e.g. 10) already guarantee, from a theoretical and practical point of view, full connectivity with probability almost 1 for very large networks (e.g. 1600 nodes). It is however interesting to observe that parameter ngb is strictly related to the (global) topology of the network under investigation. Since this parameter can only be estimated in absence of global topology information, it would be important to develop protocols as much independent as possible from it. We will show in Section 4 that the enriched protocol that we will describe in Section 3.2 is able to obtain better results than MLD, in terms of total power consumption, even when parameter ngb is strongly overestimated (conservative values, necessary when the network topology is unknown).

3.2. Protocol LMPT

Our aim here is to enrich the MLD protocol described in the previous section by introducing explicit local transmission power minimization. In order to do this, we need a little bit more of local information about neighbors, and a slightly more articulated protocol. We will refer to this new protocol as the LMPT protocol, which stands for *Local Mimimum Power Topology* protocol.

Similarly to the MLD protocol sketched in Section 3.1, where each node is, in turn, in charge of establishing links with ngb neighbors, here each node is, in turn, in charge of local optimization. We will refer to this node as the (temporarily) head node. It needs to know the list of neighbors (at the time of the local optimization) for each of its ngb potential neighbors. Moreover, each node has to send to the head node the power required to reach each one of its neighbors (it collected these information while it incrementally increased its power in order to reach a minimum number of neighbors or when it receives a connection request by another node).

Once the head node has collected these information for the ngb nodes (same parameter of MLD protocol) closest to it, it solves a local optimization problem involving itself and these nodes. Details about the local optimization problem and the technique we use to solve it, can be found in Section 3.2.1. In the meantime the nodes in its neighborhood wait for the optimization to be concluded. At these point, according to the solution of the optimization, the head node distributes the new neighbors lists and the new transmission powers for its (current) neighbors. Once they receive this information they update their state and lists.

The overhead introduced for information exchange (and for solving the local optimization problem) is justified by the efficiency gained in terms of transmission power expenditure.

It is very important to stress that when the new protocol LMPT is applied, all the theoretical results of Glauche et al, that guarantee connectivity "almost for sure" for proper values of ngb, are still completely valid. In the optimized network provided by LMPT there will be much more multi-hop communications than in the one produced by MLD. The power saving we guarantee is consequently directly related to the acceptance of multi-hop transmission instead of direct one-hop transmissions only.

Figure 2 illustrates the algorithmic implementation of the distributed rule in more detail. Initially, all nodes come with a minimum transmission power $P_i = 0$ and an empty neighborhood list $\mathcal{N}_i = \emptyset$ (with the respective list of required transmission powers $\mathcal{I}_i$ empty as well). All of them start in the receive mode. Then, at random, one of the nodes switches into the discovery mode. By subsequently sending Ask4Info messages and receiving replies, the picked node increases its power until it has discovered enough neighbors to guarantee connectivity with high probability. At this point it uses the collected information to set up the optimization problem MPT^L (see below) and solves it.

Once MPT^L has been solved, the optimal solution of MPT^L is distributed to the set of neighbors that sent their information in order to set up problem MPT^L. The head node can now set up its new transmission power P_i, its set of neighbors $\mathcal{N}_i$ with the respective required transmission powers $\mathcal{I}_i$.

The other nodes will use the information received to set up their power and their new neighbor lists. The node returns then into the receive mode. For simplicity we assume that only one node at a time is in the discovery mode.

In the receive mode a node listens to incoming Req4Info messages. Upon receipt of such a message, the node first checks whether it already belongs to the incoming neighborhood list. If yes, the requesting node has already asked

```
LMPT()
P_i := 0;
N_i := ∅;
ReceiveMode();
DiscoveryMode();
ReceiveMode();

DiscoveryMode()
P_i^disc := 0;
N_i^disc := ∅;
I_i := ∅;
While ( |N_i^disc| < ngb )
     P_i^disc := P_i^disc(1 + Δ );
     Ask4Info(i, P_i^disc, N_i^disc);
     (j_1, info_{j_1}, i),... := ReceiveInfo();
     N_i^disc := N_i^disc ∪ {j_1,...} ;
     Update I_i according to info_{j_1}, ...;
EndWhile
Create MPT^L according to I_i;
Sol := Optimal solution of MPT^L;
SendSol(i, N_i^disc, sol);
Set P_i, N_i and I_i according to sol;

ReceiveMode()
(j, P_j, N_j) := ReceiveReq4Info();
If ( i ∉ N_j )
     P_i := max(P_i, P_j);
     info_i := combination of N_i and I_i;
     SendInfo(i, info_i, j);
     sol := ReceiveSol();
     Set P_i, N_i and I_i according to sol;
EndIf
```

Figure 2. Pseudo-code for the Local Minimum Power Topology (LMPT) protocol.

before with a smaller discovery power and there is no need for the receiving node to react. Otherwise, it updates its transmission power to $\max(P_i, P_j)$. Then it sends back information about its neighbors and the respective transmission powers required to reach them. The node then waits for the head node j to solve MPT^L and collects the results. These results are used to update transmission power P_i, the set of neighbors $\mathcal{N}_i$ and the respective transmission powers $\mathcal{I}_i$.

3.2.1. Local optimization

In our approach, the local optimization is carried out by a method originally developed for the centralized version of the MPT problem, where a full knowledge of the topology of the network is supposed to exist. Here we describe how these techniques can be adapted and used while dealing with the restricted information available in a distributed environment.

Local optimization is carried out on the weighted graph $G^L = \{V^L, E^L, p^l\}$, defined as follows. The set of nodes V_L for the head node i is given by the elements of $\mathcal{N}_i^{disc}$, E^L is the set of all possible edges, and power requirements between nodes are set according to the following rule: if $j \in V \cap \mathcal{N}_i$ then p_{ij}^L is given by the respective power requirement (contained in $\mathcal{I}_i$, according to equation (1)), otherwise $p_{ij}^L := +\infty$. This last assignment is equivalent to state that node i will never reach node j in the optimal solution of MPT^L (since they are not aware of each other and do not know the required power to reach each other).

A weighted, directed graph $G_D^L = (V^L, A^L, p)$ is derived from G^L by defining $A^L = \{(i,j),(j,i)|\{i,j\} \in E^L\} \cup \{(i,i)|i \in V^L\}$, i.e. for each edge in E^L there are the respective two (oriented) arcs in A^L, and a dummy arc (i,i) with $p_{ii}^L = 0$ is inserted for each $i \in V^L$. In order to describe the new integer programming formulation for MPC, we also need the following definition.

Given $(i,j) \in A^L$, we define the *ancestor* of (i,j) as

$$a_j^i = \begin{cases} i & \text{if } p_{ij} = \min_{\{i,k\}\in E}\{p_{ik}\} \\ arg\max_{k \in V^L}\{p_{ik}|p_{ik} < p_{ij}\} & \text{otherwise} \end{cases} \quad (2)$$

According to this definition, (i, a_j^i) is the arc originated in node i with the highest cost such that $p_{ia_j^i} < p_{ij}$. In case an *ancestor* does not exist for arc (i,j), vertex i is returned, i.e. the dummy arc (i,i) is addressed.

In formulation MPT^L a spanning tree (eventually augmented) is defined by z variables: $z_{ij} = 1$ if edge $\{i,j\}$ is on the spanning tree, $z_{ij} = 0$ otherwise. Variable y_{ij} is 1 when node i has a transmission power which allows it to reach node j, $y_{ij} = 0$ otherwise.

$$(MPT^L) \quad \text{Min} \sum_{(i,j)\in A^L} c_{ij} y_{ij} \quad (3)$$

$$\text{s.t. } y_{ij} \le y_{ia_j^i} \quad \forall (i,j) \in A^L, a_j^i \ne i \quad (4)$$

$$z_{ij} \le y_{ij} \quad \forall \{i,j\} \in E^L \quad (5)$$

$$z_{ij} \le y_{ji} \quad \forall \{i,j\} \in E^L \quad (6)$$

$$\sum_{i\in S, j \in V\setminus S, \{i,j\}\in E^L} z_{ij} \ge 1 \quad \forall S \subset V^L \quad (7)$$

$$z_{ij} \in \{0,1\} \quad \forall \{i,j\} \in E^L \quad (8)$$

$$y_{ij} \in \{0,1\} \quad \forall (i,j) \in A^L \quad (9)$$

In formulation MPT^L an incremental mechanism is established over y variables (i.e. transmission powers). The costs associated with y variables in the objective function

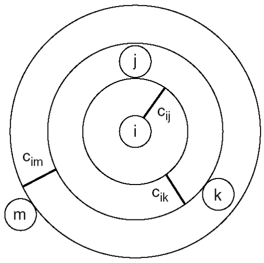

Figure 3. Incremental mechanism for the costs of formulation MPT^L.

(3) are given by the following formula:

$$c_{ij} = p_{ij} - p_{ia^i_j} \quad \forall (i,j) \in A^L \tag{10}$$

The coefficient c_{ij} is equal to the power required to establish a transmission from node i to node j (p_{ij}) minus the power required by node i to reach node a^i_j ($p_{ia^i_j}$). In Figure 3 a pictorial representation of the costs arising from the example of Figure 1 is given.

Constraints (4) realize the incremental mechanism by forcing the variable associated with arc (i, a^i_j) to assume value 1 when the variable associated with arc (i,j) has value 1, i.e. the arcs originated in the same node are activated in increasing order of p. Inequalities (5) and (6) connect the spanning tree variables z to transmission power variables y. Basically, given edge $\{i,j\} \in E$, z_{ij} can assume value 1 if and only if both y_{ij} and y_{ji} have value 1. Equations (7) state that all the vertices have to be mutually connected in the subgraph induced by z variables, i.e. the (eventually augmented) spanning tree. Constraints (8) and (9) define variable domains.

An important issue, not present for the centralized version of the problem, has to be taken into account while setting up the distributed problem MTP^L. In case there exists a node $k \in \mathcal{N}_j \backslash V^L, j \in \mathcal{N}_i^{disc}$ (k is not a neighbor of i, but k is a neighbor of j, that in turn is a neighbor of i), we have to force j to keep transmitting to k in order to ensure global connectivity. This can happen when j has already been replying to Ask4Info messages before the current round. In the situation depicted, we have to force node j to reach (at least) node k. We then add the following constraints to MPT^L.

$$y_{jl} = 1 \quad \forall j,l \in V^L, k \in \mathcal{N}_j \backslash V^L \text{ s.t. } p^L_{jl} \le p^L_{jk} \tag{11}$$

Constraints (11) are enough to directly transfer also to LMPT the nice property about global connectivity emerged for protocol MLD, i.e. very small values of ngb are enough for guarantee full connectivity almost for sure.

It is finally interesting to observe that constraints (11) also reduce the complexity of MPT^L (new facets are added), making the mixed-integer program MPT^L easier to solve.

In [9] (see also [8]) a set of facet defining valid inequalities, all based on the incremental mechanism used within formulation MPT^L, is presented. These inequalities, that are strongly based on the incremental mechanism described by equations (2), (3) and constraints (4), are able to better define the polytope associated with the linear relaxation of MPT^L, which is obtained by substituting constraints (8) and (9) with the following ones:

$$0 \le z_{ij} \le 1 \qquad \forall \{i,j\} \in E^L \tag{12}$$

$$0 \le y_{ij} \le 1 \qquad \forall (i,j) \in A^L \tag{13}$$

Since methods to solve integer programs are based on the iterative refinement of the solution of the linear relaxation, a tighter relaxation usually produces a speed up. In [9] it is shown that for the MPT problem the speed up factor can reach 1200. For this reason it is convenient to incorporate these extra inequalities into formulation (MPT^L).

It is very difficult to deal with constraints (7) of formulation MPT^L, because they are in a huge number. For this reason some techniques which leave some of them out have to be considered. We present an *iterative exact algorithm* (IEX) which in the beginning does not consider constraints (7) at all, and then adds them step by step only in case they are violated.

In order to speed up the approach, the following inequality should also be added to the initial integer problem MPT^L:

$$\sum_{\{i,j\} \in E^L} z_{ij} \ge |V^L| - 1 \tag{14}$$

Inequality (14) forces the number of active z variables to be at least $|V| - 1$ - this condition is necessary in order to have a spanning tree - already at the very first iterations of the algorithm.

The integer program defined as MPT^L without constraints (7) but with inequality (14), is solved and the values of the z variables in the solution are examined. If the edges corresponding to z variables with value 1 form a spanning tree then the problem has been solved to optimality, otherwise constraints (15), described below, are added to the integer program and the process is repeated.

At the end of each iteration, the last available solution is examined and, if edges corresponding to z variables with value 1 generate a set $\mathcal{CC}$ of connected components with $|\mathcal{CC}| > 1$, then the following inequalities are added to the formulation:

$$\sum_{i \in C, j \in V \backslash C, \ \{i,j\} \in E} z_{ij} \ge 1 \ \forall C \in \mathcal{CC} \tag{15}$$

Inequalities (15) force z variables with value 1 to connect the (elsewhere disjoint) connected components in $\mathcal{CC}$ to each other.

```
IEX()
Build integer program MPT^L;
sol := optimal solution of MPT^L;
CC := connected components defined by
    variables z of sol;
While (|CC| > 1)
    Add inequalities (15) to MPT^L;
    sol := optimal solution of MPT^L;
    CC := connected components defined
        by variables z of sol;
EndWhile
return sol.
```

Figure 4. Pseudo-code for the exact algorithm IEX, used for local optimization.

The IEX algorithm is summarized by the pseudo-code presented in Figure 4.

4. EXPERIMENTAL RESULTS

In this section we aim to compare the results obtained by the original distributed protocol MLD, described in 3.1 with those achieved by the enriched protocol $LMPT$, discussed in Section 3.2.

The following three indicators are taken into account for the comparison:

- **total power:** the sum of the transmission power of all the nodes of the network;
- **avg no of neighbors:** the average number of one-hop connections each node has to maintain in the solution generated by the protocols;
- **max no of neighbors:** the maximum number of one-hop connections a node within the network has to maintain.

The network topologies considered are those already adopted in [4]. Namely, we consider *homogeneous*, *multifractal* and *Manhattan* topologies. We refer the interested reader to [4] for details about these topologies and how to generate the respective networks. All the networks considered here have 1600 nodes, path loss exponent $\kappa = 2$ and are generated according to [4]. Different values for parameter ngb will be finally considered. Namely, for each family of networks, we will report experiments for the smallest value of ngb that guarantees connectivity with probability "almost 1 (according to [4]) and for an heavily overestimated value of the parameter.

Average results over 50 networks are summarized in Table 1, 2 and 3 for the three indicators considered, for each of the three families of networks analyzed.

Table 1. **Homogeneous networks**. Averages over 50 networks.

	$ngb = 6$		$ngb = 16$	
	MLD	LMTP	MLD	LMPT
Total power	2.55	1.40	7.42	2.33
Avg no of neighbors	7.09	2.88	19.06	2.90
Max no of neighbors	12.32	7.68	29.24	12.22

Table 2. **Multifractal networks**. Averages over 50 networks.

	$ngb = 7$		$ngb = 17$	
	MLD	LMTP	MLD	LMPT
Total power	4.31	2.05	11.44	3.07
Avg no of neighbors	8.32	3.39	18.24	2.84
Max no of neighbors	14.15	9.33	27.92	10.36

For all the experiments reported in Table 1, 2 and 3, the use of the extended protocol $LMPT$ brings a substantial gain over the original protocol MLD, in terms of both the total transmission power and the number of neighbors (average and maximum). It is particularly interesting to analyze column 2 and 3 of each table, i.e. the results of the experiments where the smallest value of ngb that guarantees connectivity is considered. The measure of the strong improvements provided by $LMPT$ can be estimated. On the other hand, the analysis of columns 4 and 5 of the tables clearly suggests that protocol $LMPT$ is much less sensitive to the value of parameter ngb than protocol LMD. As observed earlier, this is a very good property, since conservative values of the parameter can then be used when the network topology is (partially) unknown. A comparison of columns 3 and 5 gives an estimate for the loss in the quality of the solutions provided by protocol $LMPT$, due to an overestimation of parameter ngb. It is finally important to observe that protocol $LMPT$ is better than protocol LMD even when the first is run with an overestimated value of ngb, and the latter uses the smallest possible value of ngb.

We can conclude the marginal overhead generated by the extra operations carried out by protocol $LMPT$ with respect to LMD are completely justified by the results.

Table 3. **Manhattan networks**. Averages over 50 networks.

	$ngb = 10$		$ngb = 20$	
	MLD	LMTP	MLD	LMPT
Total power	3.42	0.62	8.68	0.87
Avg no of neighbors	11.89	3.51	23.71	3.42
Max no of neighbors	24.79	12.68	44.40	12.02

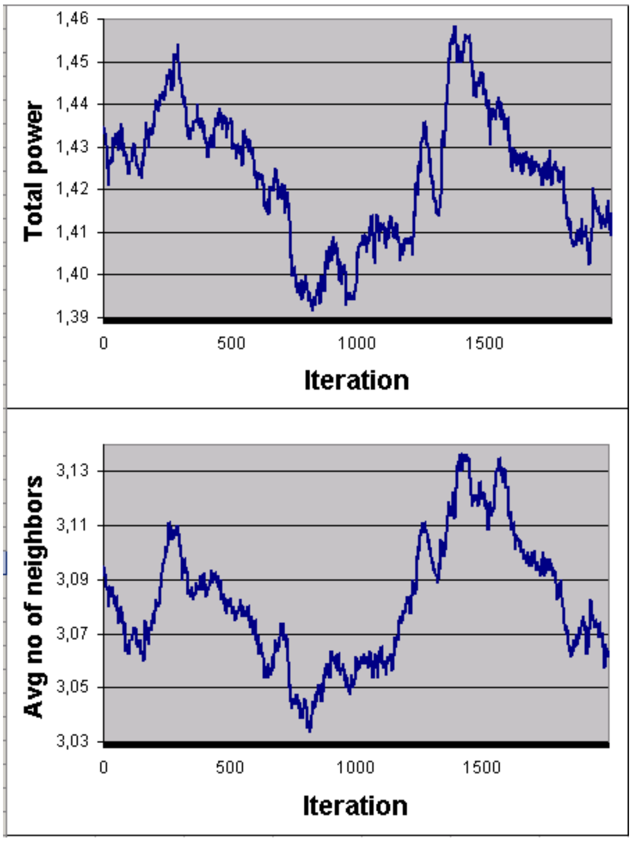

Figure 5. Adjustment phase of protocol $LMPT$. Homogeneous network with $ngb = 6$.

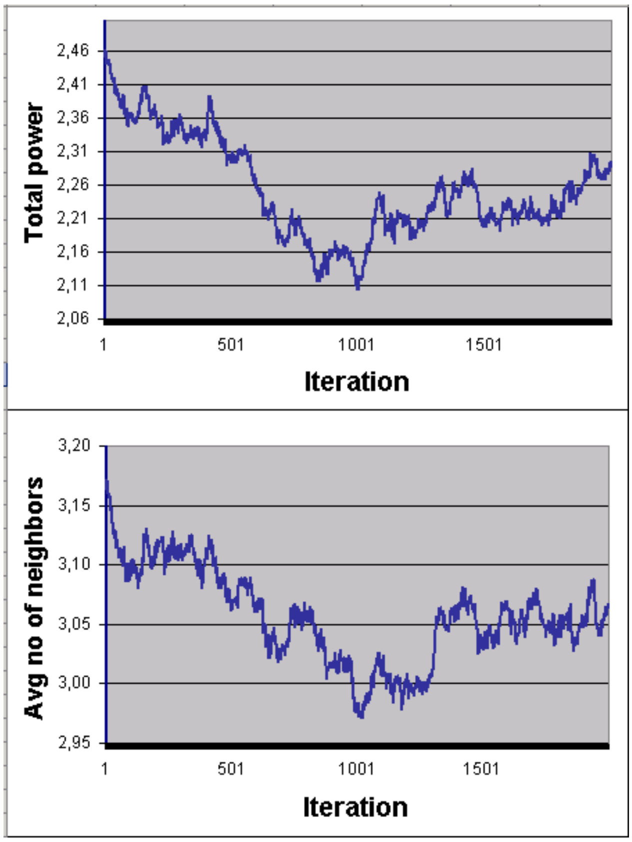

Figure 6. Adjustment phase of protocol $LMPT$. Homogeneous network with $ngb = 16$.

Another interesting issue is related to the behavior of the $LMPT$ protocol in case it is modified by adding a second, *adjustment phase* after the completion of the *set-up phase* described by procedure $LMPT$ of Figure 2. Namely, after the set-up phase is concluded[1], we make nodes iteratively switching back (at random) to the *discovery* mode, and consequently to revise their power and that of their local neighbors.

We conducted some experiments aiming to understand the effects of the adjustment phase. We run the $LMPT$ protocol as described in Section 3.2 on some networks, and after the completion of the set-up phase, we run 2000 iterations of the adjustment phase. Namely, at each iteration, a random node is selected and it is switched into *discovery* mode (in order to adjust its power and that of its neighbors). The results of these experiments carried out on two homogeneous networks, are summarized in Figures 5 and 6. Each figure contains two charts, one for the total power consumption, and one for the average number of one-hop neighbors.

[1]The completion of the (global) set-up phase is difficult to perceive at a local level, but the phase can be considered as concluded, for example, after T_{su} seconds from the starting of the sensors deployment process.

Each chart reports the value of the respective indicator during the execution of the 2000 iterations of the adjustment phase. The experiments reported in Figure 5 have been obtained with $ngb = 6$, while for those of Figure 6 the parameter ngb was overestimated at 16.

From Figure 5 and 6 we can observe that during the adjustment phase there is an oscillation of both the indicators considered. Furthermore, a clear correlation between the global power consumption and the average number of one-hop neighbors emerges. It is finally interesting to observe that while the oscillation do not produce any improvement in the values of the indications in the long-time when $ngb = 6$ (smallest possible value -Figure 5), it is possible to identify a descendent trendline for the experiments carried out with ngb overestimated at 16 (Figure 6). This aspect is very positive, since it indicates that the adjustment phase is able to improve the solutions provided by protocol $LMPT$ when parameter ngb is to large, with respect to the network topology. This makes the protocol even more independent from the (global) parameter ngb.

Charts similar to those presented in Figures 5 and 6 can be obtained also for multifractal and Manhattan networks, for which the same conclusions lined up for homogeneous networks hold.

5. CONCLUSION

In this paper we have considered the problem of assigning transmission powers to the nodes of a wireless network in such a way that all the nodes are connected by bidirectional links and the total power consumption is minimized.

We have presented a distributed protocol that embeds well-known centralized techniques for power minimization, here used in a local, distributed fashion.

Computational experiments show the effectiveness of the new protocol, and rise the question whether an approach similar to the one we have followed, i.e. using in a distributed fashion well know centralized techniques, can lead to a successful implementation also for other distributed problems. Our future research will try to answer this question.

6. 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 Wireless Communications and Networking Conference (WCNC 2003)*, pages 1889–1894, 2003.

[2] D. Blough, M. Leoncini, G. Resta, and P. Santi. On the symmetric range assignment problem in wireless ad hoc networks. In *Proceedings of the 2nd IFIP International Conference on Theoretical Computer Science (TCS 2002)*, pages 71–82, 2002.

[3] A.K. Das, R.J. Marks, M. El-Sharkawi, P. Arabshani, and A. Gray. Optimization methods for minimum power bidirectional topology construction in wireless networks with sectored antennas. Submitted for publication.

[4] I. Glauche, W. Krause, R. Sollacher, and M. Greiner. Continuum percolation of wireless ad hoc communication networks. *Physica A*, 325:577–600, 2003.

[5] L. Kirousis, E. Kranakis, D. Krizanc, and A. Pelc. Power consumption in packet radio networks. *Theoretical Computer Science*, 243:289–305, 2000.

[6] W. Krause, R. Sollacher, and M. Greiner. Self-⋆ topology control in wireless multihop ad hoc communication networks. In *Post-proceedings of the Self-⋆ 2004 Conference*, to appear.

[7] E. Lloyd, R. Liu, M. Marathe, R. Ramanathan, and S. Ravi. Algorithmic aspects of topology control problems for ad hoc networks. In *Proceedings of the Third ACS International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2002)*, pages 123–134, 2002.

[8] R. Montemanni and L.M. Gambardella. Minimum power symmetric connectivity problem in wireless networks: a new approach. In *Mobile and wireless communications networks* (E.M. Belding-Royer et al. eds.), pages 496–508. Springer, 2004.

[9] R. Montemanni and L.M. Gambardella. Exact algorithms for the minimum power symmetric connectivity problem in wireless networks. *Computers and Operations Research*, 32(11):2891–2904, 2005.

[10] R. Montemanni, L.M. Gambardella, and A.K. Das. The minimum power broadcast problem in wireless networks: a simulated annealing approach. In *Proceedings of the IEEE Wireless Communication & Networking Conference (WCNC 2005)*, 2005.

[11] R. Montemanni, L.M. Gambardella, and A.K. Das. Models and algorithms for the MPSCP: an overview. In *Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Wireless, and Peer-to-Peer Networks* (J. Wu ed.). CRC Press, to appear.

[12] R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. In *Proceedings of the IEEE Infocom 2000 Conference*, pages 404–413, 2000.

[13] T. Rappaport. *Wireless Communications: Principles and Practices*. Prentice Hall, 1996.

[14] S. Singh, C. Raghavendra, and J. Stepanek. Power-aware broadcasting in mobile ad hoc networks. In *Proceedings of the IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 1999)*, 1999.

[15] P.-J. Wan, G. Călinescu, X.-Y. Li, and O. Frieder. Minimum energy broadcast routing in static ad hoc wireless networks. In *Proceedings of the IEEE Infocom 2001 Conference*, pages 1162–1171, 2001.

[16] J. Wieselthier, G. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wireless networks. In *Proceedings of the IEEE Infocom 2000 Conference*, pages 585–594, 2000.