On the Construction of Broadcast and Multicast Trees in Wireless Networks – Global vs. Local Energy Efficiency

Leonardo Badia and Michele Zorzi
CNIT and Dipartimento di Ingegneria, Università di Ferrara
via Saragat, 1 – Ferrara, Italy – {lbadia,zorzi}@ing.unife.it

***Abstract*—Energy efficient communications in ad hoc and sensor wireless networks is a very important topic. We study the problem of creating spanning trees of low cost, where *low cost* can be viewed in terms of global or local energy efficiency. We refer to the algorithms and techniques presented in the literature, and we show an extension to them, called TDPC, which takes into account both global and local efficiency. We show that these techniques can be applied to existing algorithms, improving the performance, and addressing both needs. We show how it is possible in general to cut a trade-off between the two contrasting requests of global and local energy efficiency, by using TDPC with a particular class of algorithms.**

I. Introduction

In this paper, we study ad hoc networks from the point of view of energy efficient broadcast and multicast. An ad hoc network is characterized by two basic requirements: reliability even without pre-existing infrastructure, and completely distributed network control [1]. Strong connectivity of an ad hoc network can be managed with appropriate algorithms which consider properly the need to build such networks in low power environments.

We analyze the problem of network generation for a topology of N terminals in a designated area. We can represent each user and his terminal as a point. The network generation must be done (as fast as possible, ideally instantaneously) when a particular user (called in the following “information source”) starts a transmission to other users. We speak of broadcast (one–to–all) if the communication is direct from the source to all users, or multicast (one–to–many) if, in the communication area, only a subset of the users, called multicast group, is interested in receiving information from the source. However, we let open the possibility of using other users as relays (multihop operation). We assume to be in a power limited network, where we can speak indifferently of *power* or *energy* efficiency, assuming implicitly that we consider transmission time as constant.

In this situation, we would like to find the lowest energy spanning tree, rooted at the source, that reaches all the desired destinations. The analysis of this problem in wireless networks, which can be found in [2], [4] and [5], is different from the wireline cases, for which solutions are well known [8]. In fact, there are substantial differences between wired and wireless situations: if we consider power expenditure as a metric for the evaluation and comparison of the solutions, in wired networks this metric shows a linear behavior (the cost of spanning tree is the sum of the costs of the branches), whereas this is no longer true for wireless networks.

A trade-off can be immediately identified, which an algorithm must evaluate, between reaching from the source a large number of nodes in a single hop (with consequent higher power consumption), and reaching directly only few nodes, by using lower power, and using them as relays to reach other destinations. An important aspect of this trade-off is the non-linear attenuation characteristics of the radio channel, that plays a major role in determining power consumption.

For simplicity, we assume that each terminal in the network has sufficient transceiver resources to handle every call without blocking, and that the limiting factor is energy consumption as opposed to channel capacity. Even if it is suitable that each terminal requires low power consumption, we do not put particular bounds to the power that a node can expend for transmissions. Potentially, in our model every node is reachable from every other, as long as enough power is used. This hypothesis is only a way to take into account all possible choices for a connection. In fact, the request of a low cost tree by itself makes it unlikely to establish links with unreasonably high expenditure. Then, for sake of generality, it is more suitable that the algorithm itself discards links that need high power to be established, e.g., between very far nodes, rather than having an additional bound that would be different according to specific network.

We assume that node mobility occurs on a time scale which is significantly larger than that of data transmission, so that during the transmission we consider the positions of the terminals as fixed. On the other hand, the network may need to be “rearranged” if a terminal changes its position so that it is no longer reachable efficiently. This consideration plays a role in determining the computational complexity: since we need to quickly generate (or re-arrange) networks due to mobility, we take into account that the complexity of the algorithm we use must not be too high.

The key point of this paper is to highlight the trade-off between global and local efficiency, that implies different possible choices in establishing links and choosing transmitting nodes. Moreover, the Time Division Path Changing (TDPC) concept is presented: this technique optimizes both efficiencies, because it considers several spanning multicast trees to cover the network, that are as disjoint as possible. Then, by cyclically rotating the transmitting nodes, global efficiency is kept but high stress to single nodes is avoided.

This paper is organized as follows: in Section II the model used for the wireless environment is described. In Sections III and IV we present techniques to obtain global and local efficiency, respectively, and we introduce the TDPC concept. In Sections V we show examples of application of TDPC with different algorithms. In Section VI we evaluate the performance of the considered algorithms. Finally, Section VII presents the conclusions and summarizes the advantages of using TDPC.

II. Wireless Environment Model

We consider a network of N nodes, one of which plays the role of source of information (in the examples we denote this as node number 0). The multicast group consists of the source of information and of a number of destination nodes (at least one). We assume that each node can choose its power level without limit: we must remember however that the primary target of the

Work partially supported by the European Commission under the EYES project, IST-2001-34734.

algorithm is to search for efficient (i.e., low power) solutions, so that high power transmissions are avoided.

The power consumption in a wireless environment is strictly related to the attenuation of the radio channel. Popular mathematical models (e.g., see [6] and [7]), have three basic terms: path loss, large scale variations and small scale variations. Large scale variations are usually described with a log-normal distribution, with mean in dB equal to the path loss. Small scale variations are modeled with a Rayleigh (or Rician) distribution, where the received signal is a wide sense stationary complex Gaussian process whose envelope is a Rayleigh (or Rice) random variable.

Practical considerations allow us to simplify this model: as a matter of fact, the effect of small scale variations is mitigated by designing a receiver with diversity (e.g., Rake receiver with maximal ratio combining [6]). So we can assume that small scale variations are managed by these techniques and do not affect the link cost, except possibly for a constant factor. Large scale variations are instead often accounted for in wireless systems by an outage probability [7]: these variations affect the transmitted power, that must be sufficiently high so that the transmission is correct in a given percentage of cases (e.g., 99%). In both cases, a fixed power margin is introduced in order to meet the desired quality requirements, and this margin, being applied to all transmissions, is irrelevant in the energy optimization problem considered in the sequel.

In view of the above discussion, we consider path loss as the only significant parameter: from experimental measurements it follows that the received signal is proportional to $d^{-\alpha}$, where d is the distance between transmitting and receiving antennas, and α is an exponent with typical value between 2.0 and 4.0. A value of 2.0 means that the propagation is as in free space, whereas more realistic values for urban areas are $2.7 \sim 4.0$ [6]. We assume that α is fixed in the environment, i.e., the propagation medium is uniform.

We then have a simplified model in which the cost function of a link is proportional to d^{α}. Because we have assumed omnidirectional antennas and no upper bounds to power expenditure for a node, we can say that a transmitter that uses a power proportional to d^{α} can reach with adequate quality every point within a radius of d, and every node is reachable if sufficient power is used.

In order to simplify our environment (but this approximation is not restrictive) we always consider a 2-dimensional distribution of the users in the network: this means that we do not consider issues such as the curvature of earth, or different heights of the antennas.

We emphasize an important property that results from the broadcast nature of the wireless environment. It was called *wireless multicast advantage* by the authors of [2] and [3], and can be explained by means of the following example.

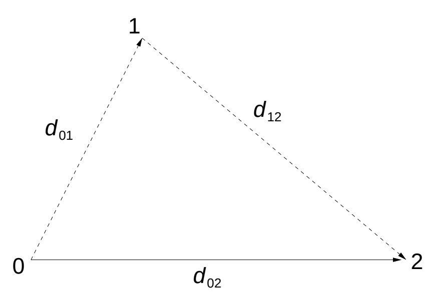

Fig. 1. Simple 3-node network (0 is the source)

Consider Figure 1, in which a very simple network is presented: a transmitter (indicated as 0) must reach two destination nodes (1 and 2). 0 can choose to either reach only 1 (the nearest) and let it reach 2, or (using a higher power level) reach both 1 and 2 with a single transmission. The first case has total power expenditure equal to ${d_{01}}^{\alpha} + {d_{12}}^{\alpha}$. The second case requires only the power needed to reach 2, since node 1, being closer, is also reached by this transmission. So in this case we have a global cost of the transmission equal to ${d_{02}}^{\alpha}$. In a wireline environment we have in both cases a global cost equal to the sum of two parts, one for each link. On the other hand, in a wireless environment the power expended by one transmitter is only the power needed to reach the farthest receiver, i.e., the maximum of the link powers. In other words, if a transmitter t is connected to a set R of receivers, its power consumption is equal to:

$$p(t) = \max_{r \in R} {d_{tr}}^{\alpha} \tag{1}$$

This property makes cost functions like global power expenditure not additive and it explains why classic algorithms for wireline networks do not work correctly for wireless networks. We show in the next section which type of approach can be taken to account for these differences.

III. Global energy efficiency

The generation of a network, given the spatial distribution of the terminals (or, equivalently, the distance between each pair of nodes), can be viewed as a spanning tree creation problem. If we want to consider the global power consumption, we associate a cost to each link. Based on the considerations of the previous section, we choose a cost function equal to d^{α}. Various algorithms to minimize global cost can be found in the literature, i.e., Prim's algorithm [8]. These algorithms were proposed to be applied to wireline networks, whereas, as already mentioned, in the wireless environment costs are not additive, because of the *wireless multicast advantage*. Therefore the global cost of the tree is the sum of the expenditures of single nodes, but this is a non-linear function of the individual link costs. The problem of finding the minimum cost spanning tree in this node-based version of the problem is harder than the link-based wireline formulation.

In this paper we focus our attention on node-based algorithms: two such algorithms were proposed in [3], one for the broadcast case, BIP (*Broadcast Incremental Power*), and one for the multicast case, MIP (*Multicast Incremental Power*). Because the difference between BIP and MIP consists only in an additional operation of pruning of the tree to remove all unutilized connections, we speak indifferently of MIP or BIP, implying that BIP is only a version of MIP in which the multicast group size is N, i.e., it includes all nodes in the network. In Sections 4 and 5 we show that it is possible to modify the energy efficiency objective by introducing a concept of efficiency related to local power consumption.

The MIP algorithm acts in steps, starting with only the source node included in the tree as the root. At every step a node is added to the tree, building a link between a node of the tree and a node not yet included. Because we start with only the source, and the network has N nodes, we complete the algorithm in $N - 1$ steps. The choice of which node to add is done through a heuristic function: the new link chosen at every step is the one with minimum additional cost in terms of power consumption. Adding this link to the spanning tree causes the lowest increase of the total network cost.

An important clarification must be made: by just following the above description it is possible that the generated spanning tree has redundant links. We must then operate a pruning of the tree in order to remove unnecessary branches. We will show later an example of this pruning.

The MIP algorithm can be seen as an evolution of algorithms such as Prim's, which provide a solution to the problem of minimum-global-cost spanning tree in the linear case [8]. Prim's algorithm builds a network with a step-by-step procedure as well, but the trees generated by the two algorithms are generally different. Because Prim's and MIP algorithms scan the set of N points in the same way, the order of complexity is $O(N^3)$ for both. Although implementation techniques that can lower this value have been proposed, here it is enough to note that the complexity is polynomial.

Moreover, there are differences in the step-by-step evaluation performed by the algorithms. The cost function c_{ij}^n (i.e., the cost of the link from i to j at the nth step) is defined in the two cases as follows:

$$\text{Prim's:} \qquad c_{ij}^n = d_{ij}^{\alpha} \tag{2}$$

$$\text{MIP:} \qquad c_{ij}^n = d_{ij}^{\alpha} - p_a^n(i) \tag{3}$$

where $p_a^n(i)$ is the power that node i needs in order to sustain links towards already reached nodes (as a consequence of the wireless environment properties discussed in Section 2, this term is the maximum of the power expenditure on the links that node i maintains after the first n steps). That is, while Prim's algorithm uses a cost function that is simply equal to the absolute cost of the link in terms of power dissipation, in choosing the link to add to the network the MIP algorithm also considers the power already expended by the transmitter node in links previously added.

This approach does not give us the optimal spanning tree, however it is very simple and requires short computation time, and the asymptotic performance is not far from optimal [3]. For this reason we do not seek further improvements of the global consumption, which would still be possible with more complicated heuristics, but do not appear significant enough.

In Section 5, we will show that the approach of this algorithm (called in [3] "node-based" approach) allows useful extensions to other performance metrics, e.g., the local consumption.

We now compare MIP and Prim's algorithms with an example. Figure 2 shows 10 points that represent a distribution of terminals in a wireless scenario. Let the exponent of the distance d in the cost function (related to the propagation) be $\alpha = 2$, i.e., we are assuming propagation as we have in open space.

Figures 2a and 2c show the network topology generated by MIP and Prim's algorithm respectively. As we have said before, Figure 2a does not represent correctly the result of MIP algorithm, because links $2 \rightarrow 4$ and $9 \rightarrow 1$ are unnecessary. In order to maintain the link to 6, node 0 uses in fact a power level sufficient to reach both nodes 4 and 1 directly. By pruning these links, we obtain correctly the solution given by MIP algorithm, shown in figure 2b.

Assuming the side of the square area to be equal to 10 units, we can compute the global cost of the trees which is (expressing results in reference power units, rpu):

$$P_{MIP} = 44,01 \text{ rpu} \quad \text{for MIP algorithm}$$
$$P_{Prim} = 71,94 \text{ rpu} \quad \text{for Prim's algorithm}$$

To illustrate the multicast case rather than broadcast, we can simply prune the trees generated, by eliminating branches that reach leaf-nodes that are not part of the multicast group. This procedure must be repeated until all leaves in the tree belong to the multicast group.

Qualitatively, we can explain the higher cost of the tree generated by Prim's algorithm by observing that we have a higher number of transmitting nodes. Comparing the trees generated by MIP and Prim's algorithms, it is easy to see that in the case

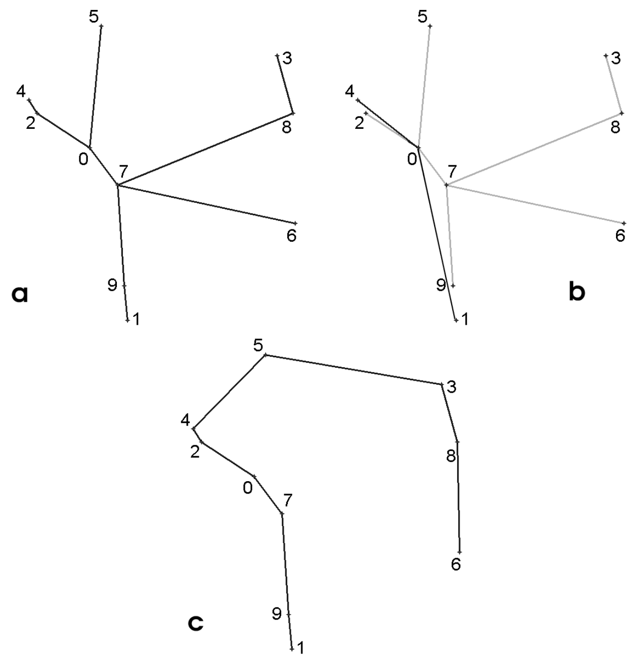

Fig. 2. A 10-node network (0 is the source)

of MIP only node 0 is involved in transmissions to relatively far nodes, so it is the only node with high consumption. In Prim's algorithm tree we have several nodes that support heavy (in term of dissipation) links, e.g. 5, 7, 8.

This criterion is often verified: the *wireless multicast advantage* principle implies that if we have few transmitting nodes (with consequently higher but concentrated power expenditure), we often obtain a lower global power consumption. We could conclude that under the aspect of global efficiency a node-based approach appears to be suitable: it allows in a simple way the construction of ad hoc networks where the number of transmitting nodes is limited and the global power dissipation is low and not far from the minimum possible.

This problem can be solved with a TDPC approach, where, instead of using a fixed set of transmitting nodes, a cyclic rotation between a small number of relays is performed.

IV. Local energy efficiency and TDPC technique

The search for an energy-efficient network can be alternatively seen from a local standpoint, as in [9] and [10]. In a real network, such as one of laptop terminals, the energy is supplied to each terminal by a battery, which can only contain a finite amount of charge. As previously observed, the fact that a spanning tree is efficient under the aspect of global consumption may correspond to situations in which few nodes consume relatively large amounts of powers, and this results in poor local efficiency since few node are highly stressed.

The requirement of avoiding to stress the single nodes leads to the consideration of the maximum of the consumption at each single node as a measure of local efficiency. Thus, we measure the local efficiency as the reciprocal of the maximum of the single node expenditures: the higher this value, the more efficient the network in a local sense. If we assume that the network reconfiguration due to exhausted battery in any node is to be avoided, we can define the lifetime of a network as the average time until an outage occurs at any node. It is easy to see that this time is proportional to the local efficiency as previously defined. So we characterize the behavior of a spanning tree under the aspect of local efficiency by the value t, called in the following *network lifetime* and defined as follows:

$$t = \frac{1}{p(\mu)} \quad \text{where} \;\; \mu = \underset{i=1..N}{\text{argmax}}\; p(i)$$

where $p(i)$ is the *average* power consumption of node i (possibly taking into account the fact that the node is not transmitting continuously, as explained later).

In fact, if even only one node is overloaded, after a short time it is impossible to sustain the network topology, because the link requests are too heavy for the charge left in the batteries, and the spanning tree has to be recomputed after the discharged node is removed. This may make it impossible to establish some links in the network, and is undesirable especially in the case of multicast: in fact, the node with exhausted battery may not even be part of the multicast group, being simply a relay.

As mentioned in Section 3, a network is "globally" efficient when the number of transmitting nodes is relatively low, which on the other hand is undesirable for local efficiency, as some nodes may have to perform a high number of transmissions over a large area of coverage, and the network lifetime decreases. In other words, it may be better to have several transmitters with low consumption than few transmitters with high energy request. This, on the other hand, works against global efficiency. Therefore, when we take into account both global and local efficiencies, we have to face two contrasting needs, and the correct trade-off needs to be identified.

In this paper we illustrate a possible extension of the node-based algorithms which also considers the local energy efficiency. The basic idea is the following: instead of using a single spanning tree for the total connection time, several trees are used, characterized by similar global costs, but maximally disjoint set of relays. By alternating among these trees we can greatly decrease local consumption, since most nodes will cyclically rest and the average power consumption. The mechanism we propose to reach our target is called Time Division Path Changing (TDPC), similar to TDMA techniques.

In practice, if M is the number of trees found, we divide the time axis into frames, and each frame into M slots. During the ith slot of each frame, the ith tree is used. [1] The benefit of acting in this way is even more significant if one considers that in a realistic model of laptop batteries ([11] and [12]) some recharge phenomena take place when the battery is not supplying power to the terminal. So the change of spanning tree is very useful: not only does it allow to share of power dissipation among a greater number of nodes, but it also makes it possible some recharge of the terminals, which we "put at rest" cyclically.

It is important to notice that the TDPC technique acts well when it is used jointly with a node-based algorithm: in fact, we can obtain M different spanning trees simply by an iterative mechanism, by just varying the heuristic of the algorithm. By doing so, we combine the local efficiency of alternating minimally overlapping trees with the global efficiency of each one of them.

In the following Sections, we illustrate these results and we show how the node-based algorithms can be extended.

V. Weighted Multicast Incremental Power Algorithm (WMIP)

A node-based algorithm, (as described in [3]) can be easily modified to incorporate a local energy efficiency objective. In fact such algorithm is based on the evaluation of a heuristic function that represents the estimated cost for the unlinked nodes: at each step, the node with the lowest value of the heuristic function is chosen and added to the tree.

[1] Note that this may apply to successive blocks of data transmitted, thereby avoiding the synchronization issues involved in time division.

We operate as follows: the MIP algorithm is used to find a global low-cost spanning tree, that is included as first tree in a set of M. Furthermore, other $M-1$ trees of the TDPC set are to be determined, with small power consumption and relays that are as different as possible.

To do so, the initial research of global low-cost spanning tree is perturbed, so that a different solution is found. In practice, we assign a weight to each node: this weight is multiplied at each step by a factor that is a function of the expended power of the node in the tree previously found, and is used to calculate the next tree. That is, in looking for the $m+1$st tree, we penalize transmissions from the nodes with high power consumption in the mth solution. This procedure has to be repeated by re-evaluating the weights and computing the next solution: by this way M different spanning trees are found.

By means of an iterative approach of this sort, called in the following *Weighted Multicast Incremental Power* (WMIP), the M generated spanning trees tend to be as non-overlapping as possible, while having similar global energy cost. Obviously the cost of solution $m+1$ is generally expected to be higher than the cost of solution m, because the higher the index of the solution, the more restrictive the conditions in which the spanning tree is found. However, as shown in the sequel, this increase turns out to be limited.

We can modify the expression of the heuristic of the MIP algorithm (3) as follows:

$$\text{WMIP:} \quad c_{ij}^{n} = w_i^{(m)}[d_{ij}^{\,\alpha} - p_a^n(i)] \tag{4}$$

A term, $w_i^{(m)}$, has been added: it is the "weight-in-transmission" of the ith node, in fact its meaning is to make the transmission from the ith node heavier. The value of $w_i^{(m)}$ changes according to the index m, that is the index of the particular solution in the set, so that we account for the node consumption in the trees already found, by encouraging the high consumption nodes in the mth solution to be inactive or to have low consumption in the $(m+1)$st solution. The $w_i^{(m)}$ coefficient can be defined in a recursive approach, that can be applied to the studied case in a simple and intuitive way. In this case it must be observed that $w_i^{(m)}$ can be interpreted as a cumulative weight, i.e., in it is kept memory of the past weight, since $w_i^{(m)}$ is defined from $w_i^{(m-1)}$ and so on.

Formally, if $p_m(i)$ is the expended power by node i in the mth tree, we evaluate $w_i^{(m)}$ as follows:

$$w_i^{(0)} = \psi(i) \tag{5}$$

$$w_i^{(m+1)} = F[\widetilde{p}_m(i)] \cdot w_i^{(m)} \tag{6}$$

$\psi(i)$ is the initial condition at node i. It can play many roles: if we start with equal battery level for all nodes, we may define $\psi(i)$ as follows:

$$\begin{cases} \psi(0) = \psi_0 & \\ \psi(i) = \psi_e & \text{if } i \text{ is not in the multicast group} \\ \psi(i) = 1 & \text{otherwise} \end{cases}$$

where $\psi_0 \geq 1$, $\psi_e \geq 1$. This is because we desire to avoid using the nodes which are not part of the multicast group, and also to avoid stressing the source, which is always involved in transmission. So we choose a higher initial weight-in-transmission for these nodes.

$F[x]$ is a multiplicative term and is a function of $\widetilde{p}_m(i)$, i.e., the normalized value of the power spent by node i in the mth tree, $p_m(i)$. The normalization, useful to keep the argument of function F between 0 and 1, is referred to the maximum power expenditure of a single node in the mth solution. Formally:

$$\widetilde{p}_m(i) = \frac{p_m(i)}{p_m(\mu)} \quad \text{where} \;\; \mu = \operatorname*{argmax}_{i=1..N} p_m(i)$$

Concerning the particular type of function $F[x]$, we request that it leaves the weight of leaf nodes unchanged, while penalizing transmissions from nodes with high power expenditure. Therefore, we need a monotonically increasing function for which $F[0] = 1$. A possible choice of $F[x]$ is an exponential-type function:

$$F[x] = A^{\beta x} \qquad \text{with } A > 1, \beta > 0$$

where the coefficient A and β may be tuned in order to have better performance. Our simulations studies have shown that, although the optimality of the performance is dependent on the values of these parameters, similar results, that show the advantage of applying TDPC to this algorithm, can however be highlighted even with a choice of A and β in large ranges, for example $A = 1.1 \sim 1.5, \beta = 1 \sim 3$.

From the previous formulation of WMIP algorithm we can expect that it will be impossible to have M spanning trees whose sets of relays are disjoint partitions of the set of N nodes of the network. This happens first of all because the network has a source node and the transmission from this node is unavoidable. Furthermore, as we still build trees based on global energy efficiency, depending on the topology some key nodes may have to be active most of the time.

It is therefore very difficult not to use nodes that are in strategic positions and can cover large parts of the network: if we avoid using these nodes we must expect a very high increase of the total power expenditure. It is important to notice that the node-based approach does not increase the computational complexity of the algorithm: the only significant variation under this aspect is a scan for every node and the evaluation of the weight in transmission. This additional complexity is however negligible compared to the original term $O(N^3)$.

VI. Simulation Results

We can afford a performance comparison, and at the same time gain some deeper understanding of TDPC's advantages, by considering a very simplistic approach to generate the set of M solutions, i.e., an algorithm in which we use a random tree generation, rather than a low-cost based algorithm. In order for this to be meaningful, we must put bounds to the randomly operated choices, thereby discarding particularly bad options that can lead to spanning trees with many very high-power branches, because such trees are bad under both global and local efficiency. With this variation, the random algorithm (denoted in the figures as Increasing Bound Random algorithm, IBR) does not operate in a completely random way, but it discards a randomly chosen link if its power request is greater than some predefined upper bound. The choice is repeated, and the upper bound is increased by an assigned percentage, in order to avoid situations in which we have no links with acceptable cost. By appropriately choosing this bound we can practically fix the global power consumption of the entire tree. In fact, by repeating the algorithm, we usually find another solution with similar global consumption, but in general with power expenditure assigned to other nodes.

So the value of global consumption of an IBR solution is constrained into almost fixed ranges, that are generally not efficient, and iterations can not change significantly this value: TDPC can improve the efficiency, but only under local aspect, i.e., by lowering the local consumption of single nodes, with the principle of using several spanning trees, rather than only one. In thios case, we only exploit TDPC's ability to share the cost among nodes, without any explicit attempt to choose energy-efficient solutions.

We have chosen a square area of 10×10 units in which we randomly place N nodes. The simulations allow the possibility to vary the placement of the nodes in the area according to several random distributions, but we found that the performance of the algorithms is only marginally affected by the specific statistics of the node distribution, so in the following we will only refer to the case of nodes uniformly distributed within the square area. The exponent in the path loss expression α has been set equal to 4 as a default value.

We assign the internal parameters of the algorithms as follows. The ψ-parameters defining the initial conditions in the WMIP algorithm are selected by setting: $\psi_0 = 1.4$, $\psi_e = 1$. This choice is meant to give to the source more "endurance", and to not use special care against using nodes that are not part of the multicast group. The $F[x]$ function is an exponential-type function. A definition of parameters A and β of broad validity, useful in the comparison of different situations, was found to be $A = 1.2$, $\beta = 1.5$. This choice led to generally good performance for various types of environment and different networks. More accurate choices of these parameters would be still possible in each specific situation, although only marginal improvements can be expected.

Figures 3-5 represent the trade-off between global consumption and network lifetime in various scenarios: each point in the graphs corresponds to a pair of global efficiency/network lifetime values, obtained with the specified algorithm and a given choice of the number M of considered paths in TDPC approach. The values are normalized to the solution of minimum global cost (that is usually the first solution found by the WMIP algorithm). The lifetime t is the reciprocal of the local consumption of the node that suffers the highest expenditure, averaged over all trees, because as explained in Section 4, this is the node that can lead to a change in network topology, due to insufficient charge left in battery.

Thus the best points in the graphs are the ones with low global consumption and increased lifetime, i.e., in the upper left corner: ideally, a perfect optimized situation, that can be obtained only in a very special network, has the global consumption equal to 1, i.e., not increased, and the local consumption perfectly distributed among all nodes, so that the normalized lifetime is N.

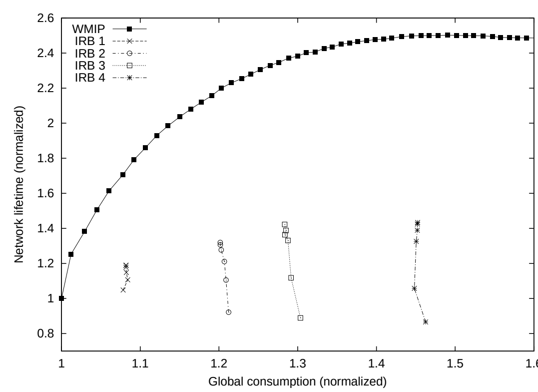

Fig. 3. 100-node network, 10-user multicast group

The first solution of WMIP algorithm corresponds to point $(1, 1)$: values of M greater than 1 attempt to increase lifetime while slightly increasing global dissipation. As a function of M, we can cut the trade-off between these contrasting requests, by choosing the operating point that gives the best conditions.

Consider Figure 3: it refers to a 100-node network, in which

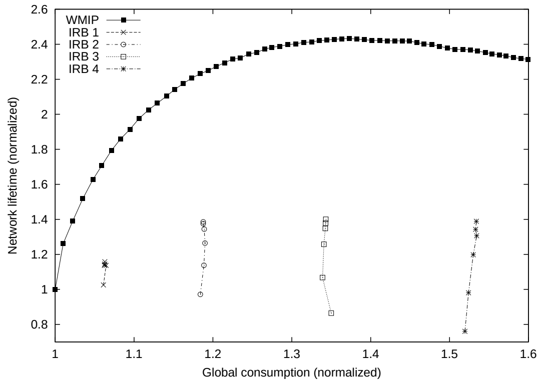

Fig. 4. 100-node network, 50-user multicast group

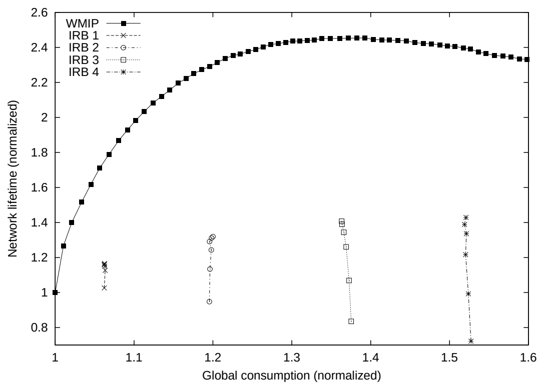

Fig. 5. 100-node network, broadcast

the multicast group is composed of 10 elements (including the source). Each point in the curve labelled "WMIP" corresponds to a TDPC solution with $M = 1, 2, 3 \ldots$. It can be seen that by increasing M the global consumption is increased while the local consumption is decreased.

We can observe that if we let M equal to 4, we obtain a global normalized consumption of 1.05, i.e., 5% higher than for the best global case, while the network lifetime is increased by 50%. In the same scenario, it is possible to use higher values of M, thereby achieving an even lower local consumption but at the cost of a slightly higher total consumption. For example, $M = 10$ leads to a global normalized consumption of 1.15, but allows to double the normalized lifetime, whereas by taking a higher value of M it is possible to obtain a lifetime about equal to 2.5, while paying a global cost increase of at least 40%.

Thus, it is easy to choose a point of trade-off, by simply setting M equal to a value which meets the desired consumption requirements. An interesting conclusion that can be drawn from these results is that by only increasing the whole network expenditure by about 15 percent, the local consumption can be halved, i.e., the lifetime of the network, as limited by the battery charge, can be doubled. Depending on the number of nodes in the network, the size of the multicast group, and other parameters that we can simulate, it is possible to choose a correct point of trade-off for a network scenario in a very general way. We only have to run the WMIP algorithm M times and then implement a TDPC-mechanism with the appropriate number M of solutions to be used in a TDMA-like fashion.

As a final remark, notice that the performance of the IBR algorithm, also reported in the plots, is significantly poorer than that obtained with WMIP; however even a random tree generator obtains an improved performance, only by using TDPC principle. It is possible to conclude that a consistent performance improvement is allowed, in both cases, but this improvement is more useful if TDPC is applied along with a power aware algorithm to build the trees, so that energy consumption, both in global and local sense, can be kept small.

VII. Conclusions

In this paper, we have addressed issues arising when designing algorithms to generate spanning trees for broadcast and multicast comunications in wireless networks. In particular, attention has been given to the issue of energy efficiency, both in global terms (total network consumption) and in local terms (focusing on the most highly stressed nodes).

We have proposed a novel approach which combines global and local considerations, thereby providing a flexible tool to design network topologies under energy constraints. In addition, the proposed approach is an a priori technique, where the computational cost can be paid all at once, and run-time changes of the spanning tree (e.g., to continuously take into account the discharge of the nodes as in [9]) are not necessary, unless a node is completely discharged and needs to be removed.

TDPC also gives the necessary flexibility to trade-off between low global cost and long network lifetime. This can be done in a very easy way, from tradeoff curves such as those shown in this paper.

For the future, possible extensions of this work could be first of all an implementation of TDPC with distributed algorithms, that are more interesting for real implementations over ad hoc networks. Moreover, the generation of many spanning trees can be used as mesh generation, i.e., to introduce a desired amount of redundancy, by using a set of spanning trees instead of only one at same time. This may be useful or necessary, when the possibility of outage for the nodes of the network is accounted. Another quite natural sequel of the work of this paper could be the specialization to practical cases of interest, like sensor networks or other types of network in low power environments, by simply modifying the network parameters, or distinguishing between different properties of the nodes.

References

[1] Z.J. Haas, M. Gerla, D.B. Johnson, C. E. Perkins, M.B. Pursley, M. Steenstrup, C.-K. Toh, "Wireless Ad Hoc Networks", *Guest Editorial IEEE JSAC*, vol. 17, no.8, August 1999.

[2] J.E. Wieselthier, G.D. Nguyen, A. Ephremides, "Algorithms for energy-efficient multicasting in ad hoc wireless networks", *Proc. MILCOM 1999 IEEE Military Communications Conference*.

[3] J.E. Wieselthier, G.D. Nguyen, A. Ephremides, "On the construction of energy-efficient broadcast and multicast trees in wireless networks." *Proc. INFOCOM 2000 IEEE Conference on Computer Communications*.

[4] V. Rodoplu, T.H. Meng, "Minimum Energy Mobile Wireless Network", *IEEE JSAC*, vol. 17, no.8, August 1999.

[5] S. Singh, M. Woo, C.S. Raghavendra, "Power-aware routing in mobile ad hoc networks", *Proceedings of 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, 1998.

[6] T.S. Rappaport, "Wireless Communications: Principles and Practice", Englewood Cliffs, NJ: Prentice Hall, 1996.

[7] Proakis, "Digital Communications", 3rd ed. New York: McGraw-Hill, 1995.

[8] N.A. Lynch, "Distributed Algorithms", San Mateo CA: Morgan Kaufmann, 1996.

[9] C.F. Chiasserini, R.R. Rao, "Routing protocols to maximize battery efficiency", *Proc. MILCOM 2000 IEEE Military Communications Conference*.

[10] J.H. Chang, L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks", *Proceedings INFOCOM 2000. 19th Annual Joint Conference of the IEEE Computer and Communications Societies*.

[11] T.F. Fuller, M. Doyle, J.S. Newman, "Relaxation phenomena in lithium-ion-insertion-cells", *J.Electrochem. Soc.*, vol. 141, 1994.

[12] E.J. Podlaha, H.Y. Cheh, "Modeling of Cylindrical Alkaline Cells. VI: Variable Discharge Conditions", *J.Electrochem Soc.*, vol. 141, 1994.