

# Adaptive Broadcast Consumption (ABC), a New Heuristic and New Bounds for the Minimum Energy Broadcast Routing Problem\*

Ralf Klasing<sup>1</sup>, Alfredo Navarra<sup>1,2</sup>, Aris Papadopoulos<sup>1,3</sup>, and  
Stéphane Pérennes<sup>1</sup>

<sup>1</sup> MASCOTTE project, I3S-CNRS/INRIA/Univ. Nice, Sophia Antipolis, France.

{FirstName.LastName}@sophia.inria.fr

<sup>2</sup> Computer Science Department, University of L'Aquila, Italy.

navarra@di.univaq.it

<sup>3</sup> Department of Computing, Imperial College London, UK.

aris.papadopoulos@imperial.ac.uk

**Abstract.** In this paper we present a new heuristic called *Adaptive Broadcast Consumption* (ABC for short) for the *Minimum-Energy Broadcast Routing* (MEBR) problem. We first investigate the problem trying to understand which are the main properties not taken into account by the classic and well-studied MST and BIP heuristics, then we propose a new algorithm proving that it computes the MEBR with an approximation ratio less than or equal to MST, for which we prove an approximation ratio of at most 12.15 instead of the well-known 12 [10]. Finally we present experimental results supporting our intuitive ideas, comparing ABC with other heuristics presented in the literature and showing its good performance on random instances even compared to the optimum.

## 1 Introduction

The *Minimum-Energy Broadcast Routing* (MEBR) problem is one of the most intensively studied problems in recent years for the various applications in which it can be applied (see [3,4,5,9,10] for a survey). The best result presented in the literature concerning the approximation is a ratio of 12 thanks to [10], but we show in Section 3 that 12.15 is the actual value. In this paper, an analytical study of the performance of some basic heuristics such as Minimum Spanning Tree (MST), Shortest Path Tree (SPT) and Broadcast Incremental Power (BIP) is presented. According to the results we obtained, MST seems to be the most promising heuristic to reduce the gap between the best known lower and upper bounds. The Euclidean MST, in fact, has a lot of useful properties that can be applied to try to reduce the gap. On the other hand, MST seems to be very far

---

\* Work supported by the European projects RTN ARACNE and IST FET CRESCCO, by the French MASCOTTE project I3S-CNRS/INRIA/University of Nice, Sophia Antipolis and by the Italian project REAL-WINE.

from a good heuristic to solve MEBR due to the static properties of its construction. It is evident, in fact, that to refine a solution, a sort of backtracking is needed to adapt at each step the current solution to the newly considered configuration of nodes in the plane. In this paper, we investigate other kinds of heuristics, related to MST, adding a sort of backtracking in order to modify properly a current solution. We present a promising heuristic supported by experimental results and some basic theoretical results. In particular, after proving that our *Adaptive Broadcast Consumption* (ABC) algorithm outputs a solution whose value is at most equal to MST, we show the near optimal performance of ABC on random instances and the decrease, on average, of 5% on the total cost with respect to BIP.

The paper is organized as follows. In the next section, we introduce the problem of Minimum Energy Broadcast Routing with notations and the necessary definitions. In Section 3, we prove the mentioned upper bound 12.15 for MST refining the arguments of [10]. In Section 4, we present a new heuristic for the MEBR problem trying to trace some more specific property that is completely ignored by previous attempts. In Section 5, we present a formalization of the problem using integer linear programming to obtain optimal solutions. In Section 6, we provide experimental results obtained in various scenarios of random instances, comparing the developed heuristics. Finally, in Section 7, we give some conclusive remarks and discuss some open questions.

## 2 Minimum Energy Broadcast Routing

The Minimum Energy Broadcast Routing (MEBR) problem arises in the context of computing communication schedules in *Ad-Hoc* networks. These are usually special networks used in specific situations such as military operations, emergency disaster relief and so on. In contrast with *cellular*, *single-hop* networks, Ad Hoc wireless networks consist of a homogeneous network of mobile nodes and require no established, supporting backbone infrastructure. A communication session can be established through a series of wireless links involving any of the network mobile nodes and therefore Ad Hoc networks are *multi-hop* networks. In the problem we study, the nodes have the ability to adjust their transmission power as appropriate. Thus every mobile node is assigned a transmission range and every node inside this range receives its message. In simple terms, the aim is to assign ranges in such a way that the total energy that is consumed is minimum. Considering the fact that the mobile nodes operate using batteries of limited capability and the nature of the operations for which such kind of networks are used, it is easy to understand the importance of the energy conservation.

According to the mostly used power attenuation model [11] when a node that we call  $s$  transmits to a node  $r$  with power  $P_s$ , the power at the point that node  $r$  lies will be:

$$P_r = \frac{P_s}{\|s, r\|^\kappa} \quad (1)$$

where  $\|s, r\|$  is the Euclidean distance between the source node  $s$  and the receiving node  $r$ , and  $\kappa$  is the *distance power gradient*. In the real world it holds that  $2 \leq \kappa \leq 6$  according to the topology of the space, but for an even, flat surface  $\kappa \approx 2$  (see [8]). Therefore for our analysis we assume that  $\kappa = 2$ . Also assuming normalization to 1, in order to establish communication between  $s$  and  $r$ , the following expression must be valid:

$$\frac{P_s}{\|s, r\|^\kappa} > 1 \tag{2}$$

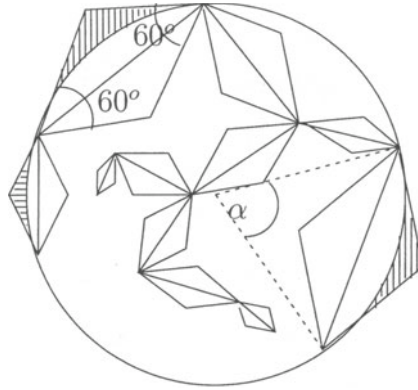
Note that in order to send the message from  $s$  to  $r$  there is the possibility of using a node  $i$  as an intermediate node, if one exists, resulting in a decrement of the power needed. With the assumption that  $\kappa = 2$  choosing the path  $\{s, i, r\}$  is cheaper only if  $i$  lies inside the circle around the middle point of the line segment  $s, r$  with diameter  $\|s, r\|$ . Otherwise using  $i$  makes the solution more expensive.

As mentioned, the most common algorithms that are proposed in the literature to solve MEBR are MST and its improved variations, SPT and BIP (see [1,10,11]). In this paper we propose a new algorithm, *Adaptive Broadcast Consumption* (ABC for short), and we provide an analytical comparison of the algorithms through extensive experiments. We do that for all the mentioned algorithms except for SPT. The reason is that SPT is clearly more suitable for the *Minimum-Energy Unicast Routing Problem* (MEUR, see [7] for a survey) than for the MEBR. Moreover, the lower bound presented in [10] for SPT is very high. Although MST seems to suit the MEBR much better, the reality is that MEBR is different and much more challenging than the typical MST problem.

### 3 Upper Bound for the MST Heuristic

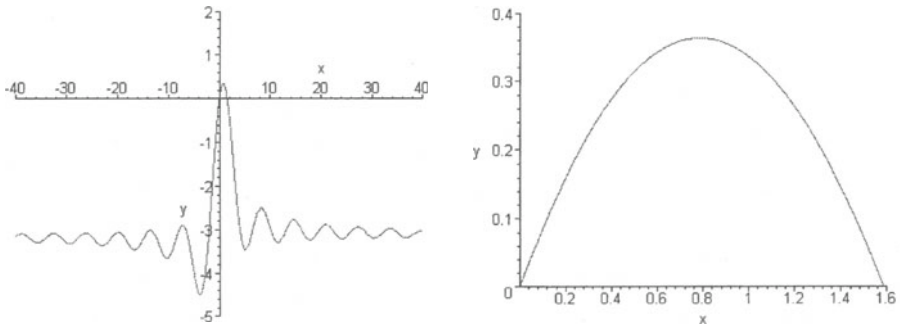
In this section, we prove the upper bound for the MST heuristic that will be useful also to bound the other heuristics. First of all, we briefly describe the technique used in the [10] where the authors claimed that the approximation ratio of *MST* is at most 12. In that paper, the authors used some geometric properties of the Euclidean MST.

Informally speaking, they associated the so-called “diamond” to each edge of the produced MST to estimate its cost comparing it with the covered area. The diamond is a rhombus whose bigger diagonal coincides with an edge of the MST and the angles at its endpoints are 60 degree (see Figure 1). After proving that there are no superposition between each pair of diamonds associated to an MST computed over an instance of MEBR inside a circle of radius 1, they calculate the “sticking-out area” ( $S(\alpha)$ ) of those diamonds, that is, the outgoing area from the circle where  $\alpha$  is the angle between the two radii to the endpoints of a diamond placed over the circumference (an example can be seen in Figure 1). The sticking-out area of a diamond is  $S(\alpha) = \frac{1}{2} \sin \alpha + \frac{\sqrt{3}}{6}(1 - \cos \alpha) - \frac{\alpha}{2}$ , the maximum is obtained for  $\alpha = \frac{\pi}{3}$  (see [10]). In their calculation they maximized properly the biggest sticking-out area of a single diamond but there is some problem with the global scenario. In the following we show the erroneous argumentation



**Fig. 1.** Diamonds associated to an MST inside a circle.

of [10] that, in any case, remains a milestone for the MEBR problem. Actually, the quantity that must be maximized to obtain the upper bound is  $\sum_{i=1}^k S(\alpha_i)$  with  $\sum_{i=1}^k \alpha_i \leq 2\pi$ . Roughly speaking, this quantity represents the maximum sticking-out area of  $k$  diamonds that can be placed around the circumference of the circle of radius 1 compatible with the Euclidean MST properties.



**Fig. 2.**  $y = \frac{2\pi}{x} \left( \frac{1}{2} \sin x + \frac{\sqrt{3}}{6}(1 - \cos x) - \frac{\pi}{2} \right)$  and its enlargement.

**Theorem 1.** *The approximation ratio of the Euclidean minimum spanning tree heuristic for the minimum energy broadcast routing problem is at most 12.15.*

*Proof.*

$$\sum_{i=1}^k S(\alpha_i) = \sum_{i=1}^k \alpha_i \frac{S(\alpha_i)}{\alpha_i} \leq \sum_{i=1}^k \alpha_i \max_{0 \leq \alpha \leq 2\pi} \left\{ \frac{S(\alpha)}{\alpha} \right\} \leq 2\pi \max_{0 \leq \alpha \leq 2\pi} \left\{ \frac{S(\alpha)}{\alpha} \right\}$$

In Figure 2, we show the graph of the function  $y = 2\pi \frac{S(x)}{x}$  whose maximum is  $y_{\max} = 0.36324\dots$  and this maximum is obtained for  $x = \frac{\pi}{4.00458\dots} \approx \frac{\pi}{4}$ . Adding  $\pi$  (the area of the circle of radius 1) to the maximum sticking-out area, we obtain the upper bound for MST. From [10],

$$\sum_{e \in \text{mst}(V)} \|e\|^2 \leq \frac{\pi + y_{\max}}{\frac{\sqrt{3}}{6}} = 12.14110\dots < 12.15.$$

□

The proof of Theorem 1 shows that the quantity  $\sum_{i=1}^k S(\alpha_i)$  (i.e. the maximum sticking-out area of  $k$  diamonds that can be placed around the circumference of the circle of radius 1 compatible with the Euclidean MST properties) is maximized by 8 diamonds of area  $S(\frac{\pi}{4})$  instead of 6 of area  $S(\frac{\pi}{3})$  (as was assumed in [10]). In fact, considering the maximum value for  $S(\alpha)$  that is obtained for  $\alpha = \frac{\pi}{3}$ , only 6 diamonds can be placed around the circumference.

## 4 The ABC Algorithm

A first easy observation in order to improve the basic MST algorithm is that for each node we need at most only one outgoing edge, i.e. the longest one.

This technique is also used in [1,3,4,9,11] for the Broadcast Incremental Power (BIP) where the nodes are discovered one by one according to the improvement of the energy needed to cover the new node. But also this algorithm does not take into account another property that is really important from a practical point of view. What we really need is an algorithm that is able to decide how to remove some previous edge(s) (or circle(s)) included in the current solution, adding a new one according to the considered points over the plane that we want to cover. What is important to decide is the order of the discovered nodes. The BIP order or more precisely the technique used to order the nodes seems to be inappropriate, since finding the cheapest node for the algorithm, that at each step needs to compute the deletion of previous circles, is very expensive in terms of time and resources. A more appropriate order seems to be choosing at each step the closest node to the current solution. This intuition is supported by our experimental results. In practice, we order the nodes in Prim's MST order. Due to this property, we will show that in general the cost of ABC is lower than or equal to the one of MST. After having decided the order, we describe the algorithm that tries to add the newly discovered node, at each step, according to its Euclidean distance from the current solution, in the cheapest way. This means that we try to reach the new node from each node already present in the solution considering that we can remove previous circles that become useless because of the new one. Let  $V$  be a set of nodes and  $s \in V$  the given source, denoting with  $i$  the  $i$ -th node discovered at step  $i$  ( $s \equiv 0$ ), the two invariants that we need to guarantee a valid solution at each step  $i$  are:

1. Every node from 1 to  $i - 1$  is covered by some circle.
2. Every node from 1 to  $i - 1$  admits an induced path back to the source 0.

Now we can describe more precisely how to design the ABC algorithm:

```

procedure  $ABC(s, V)$ 
1:  $S \leftarrow s$       \* Set of covered nodes
2:  $V \leftarrow V \setminus s$   \* Set of uncovered nodes
3:  $Sol \leftarrow null$   \* Set of used circles
4: while  $|S| < |V|$  do
5:   Let  $v$  be the closest node of  $V$  to  $S$ 
6:   if  $v$  is not already covered by  $S$  then
7:      $Sol \leftarrow Sol + Cost(v, S, Sol)$ 
8:   end if
9:    $V \leftarrow V \setminus v$ 
10:   $S \leftarrow S + v$ 
11: end while

```

The Function  $Cost(v, S, Sol)$  outputs the minimum cost circle needed to reach the current node  $v$ . As we said before this calculation is made according to the 2 previous invariants.

```

procedure  $Cost(v, S, Sol)$ 
1: for each node  $x \in S$  do
2:    $C_{xv} \leftarrow C_{xv} - C_{xy} - P$ 
3:   where  $C_{xy} \in Sol$  is the previous circle
4:   used by  $x$  to reach some other node in  $S$ ,
5:   unless it does not exists and
6:    $P$  is the set of the circles that now
7:   can be removed due to  $C_{xv}$  coverage.
8: end for
   Return  $\min_{x \in S} \{C_{xv}\}$ 

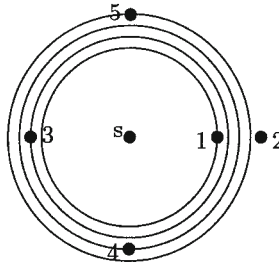
```

The most complicated part of the algorithm is how to compute the set  $P$ . In our implementation we associated a set of covered nodes to each circle, so that the first step to compute  $P$  is to verify if the set of covered nodes of a circle is a subset of the new circle. If not, we cannot remove that circle, otherwise we have to check the invariant number 2. This means, in practice that we have to verify if without the candidate circle to be removed, a path from the source 0 to the center of the new circle there exists.

By construction it is easy to see that the above algorithm is a refinement of Prim's MST algorithm in which at each step the closest node to the existing structure is chosen. The difference is that, due to the specific circle area coverage, the algorithm outputs a forest of paths refining the cost of the MST. Hence the following lemma holds.

**Lemma 1.** *Given a set of nodes  $V$  over the Euclidean 2-dimensional space and a source  $s \in V$ ,  $cost(ABC(s, V)) \leq cost(MST(s, V))$ .*

ABC can also be considered as an extension of the BIP program since at each step we try to increase the power of each node, with the main difference that in ABC we can also reduce it due to some different kind of coverage. This process can be easily tested on the example of ten nodes presented in [11] for BIP in which, applying ABC, we obtain the optimal solution. Unfortunately we cannot claim a similar lemma for BIP (as for MST) because it can happen in some specific case that the BIP solution is cheaper. An example of this case is shown in Figure 3.



**Fig. 3.** Example in which ABC can lose with respect to BIP.

In this figure, ABC adds the circle  $C_{1,2}$  just after  $C_{s,1}$  because of the distance. This way, when it realizes that a bigger circle centered in  $s$  is needed, it does not cover also the node 2. BIP, instead, increasing the solution by the cost discovers node 2 at the end of its computation.

In terms of complexity, ABC takes  $O(n^3)$  time, in fact the *while* loop inside the main procedure is executed exactly  $n - 1$  times, the *for* loop in the Procedure  $Cost()$  is executed at most  $n - 1$  times and to find the set  $P$  we need at most  $n - 1$  comparisons. Therefore in practice the complexity is the same as for BIP. Also for the approximation ratio we cannot say in general that ABC is better than BIP although the experimental results show better values on random instances but in general, we have as approximation ratio again 12.15 since it is bounded by the performances of MST.

The idea behind this algorithm is the result of our effort to solve all the practical problems presented in the literature. Every example and mainly worst-case example presented for each heuristic in [2,6,10,11] is optimally solved by ABC. This is due to the further property of this new algorithm for which, according to the order of the nodes, every path or subpath of the output solution is less expensive than the circle with radius equal to its endpoints when it covers the same set of nodes.

Before concluding the section, we show in Figure 4 the worst example for ABC that we found by studying the problem. Notice that in this example the



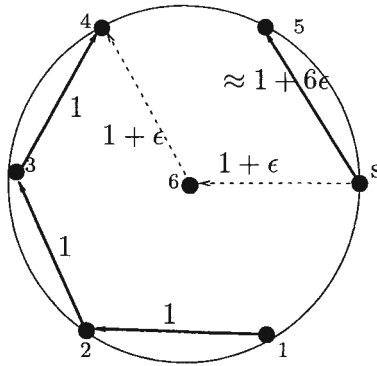


Fig. 4. Worst-case example for ABC.

approximation ratio obtained by ABC is 2 that is very far from 12.15. In the first four steps the ABC solution is composed by the path  $\{s, 1, 2, 3, 4\}$ . When node 5 is discovered ABC removes the edge  $(s, 1)$  and adds  $(s, 5)$ , finally node 6 is already covered. The same solution is obtained by applying BIP. MST, instead, is even worse (its reduced version as well) as it outputs the path  $\{s, 1, 2, 3, 4, 5, 6\}$ . The optimal solution costs  $2(1 + \epsilon)^2$  as it is composed by the two edges  $(s, 6)$ ,  $(6, 4)$ .

The problem for ABC in the example of Figure 4 is that the central node is discovered at the end of the computation, so it does not test the circles centered in that node. On the other hand, the central node is discovered by BIP as the second node but the algorithm is not “smart” enough to realize that after it could cover all the remaining nodes with only one circle. As a consequence of the previous result and Theorem 1, the following corollary holds.

**Corollary 1.** *The approximation ratio  $\rho$  of ABC is  $2 \leq \rho(ABC) \leq 12.15$ .*

### 5 ILP Formulation

In this section we present an integer linear programming formulation to solve optimally the MEBR problem. We use it for our experiments to compare the results of ABC, MST and the other heuristics to the optimal solution in “small” instances. Formally we describe the problem as the research of the minimum cost of the sum of the squared radii of the used disks to perform a flow problem.

Given  $n$  nodes  $\{s \equiv v_1, v_2, \dots, v_n\}$ , distributed on the 2-dimensional space, let  $d_{i,j}$  be the Euclidean distance between nodes  $v_i$  and  $v_j$ ; let  $y_{i,j} \in \{0, 1\}$  be the variable equal to 1 if the solution contains the disk centered at  $v_i$  and radius  $d_{i,j}$ , 0 otherwise; let  $x_{i,j} \in \mathbb{N}$  be the variable describing the number of units of flow going from  $v_i$  to  $v_j$ . We add a further node,  $t$ , as the destination node, setting  $\forall i \neq s, d_{i,t} = 0$  and  $x_{i,t} = 1$ .



$$\begin{aligned}
 &\min \sum_{i,j} d_{i,j}^2 y_{i,j} \\
 &\quad \sum_{j \neq t} x_{s,j} = n - 1 \\
 &\quad \forall k, \sum_{i \neq k} x_{i,k} - \sum_{j \neq k} x_{k,j} = 0 \\
 &\quad \sum_{i \neq s} x_{i,t} = n - 1 \\
 &\quad \forall i \neq t \text{ and } j \neq s, x_{i,j} \leq (n - 1) \sum_{k: d_{i,j} \leq d_{i,k}} y_{i,k} \\
 &\quad \forall i, j \quad y_{i,j} \in \{0, 1\}, x_{i,j} \in \mathbb{N}
 \end{aligned}$$

The first condition describes that the source  $s$  outputs  $n - 1$  units of flow; the second is the standard condition for the flow conservation. The third describes that each node, except  $s$ , sends a unit of flow to  $t$ . The last means that if a disk  $y_{i,k}$  covers more than one node, then  $v_i$  is allowed to send units of flow to each covered node. This is not the first time the the linear programming method is used to formalize the MEBR problem (see [6] for instance) but, to the best of our knowledge, it is the first time that experimental results comparing several heuristics with the real optimal solution appear.

## 6 Some Experimental Results

In this section we briefly present a comparison between the heuristics that we took into consideration which are the MST, BIP, ABC and the optimal algorithm obtained by the ILP formulation, as part of the extensive experiments we ran. We show the good performance of the ABC algorithm compared to the others. The experiments were performed among 500 instances for each different size. The number of nodes is between 5 and 50. The distribution of the nodes is random in a 2-dimensional space inside a square of dimension  $5 \times 5$ , even the source node is randomly chosen. The metric used to perform the comparisons is the total power of the broadcast tree. In the following pictures the X-axis represents the instances and the Y-axis represents the correspondent costs of each heuristic.

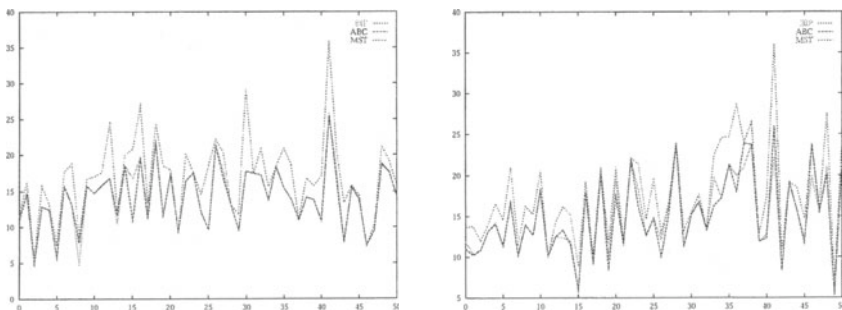
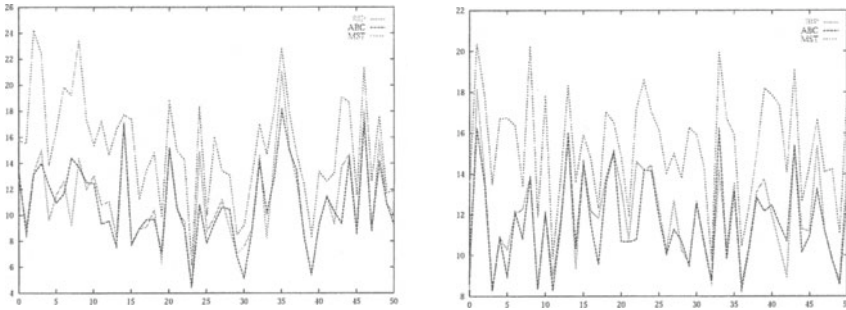


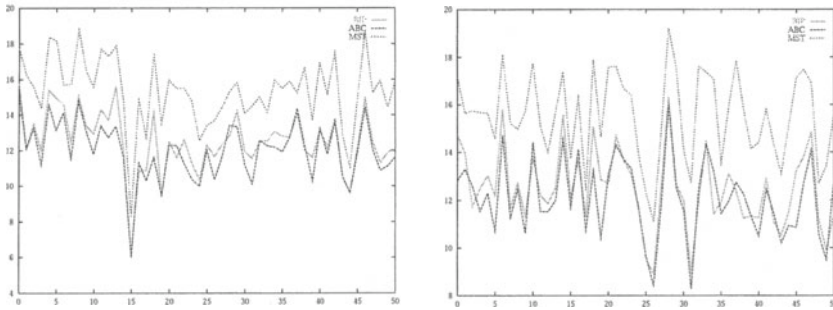
Fig. 5. ABC, BIP, MST with 5 and 7 nodes

In small instances, such as in Figure 5, of 5 and 7 nodes, the distance between the heuristics is on the average almost zero, mainly between ABC and BIP the solutions are almost the same.



**Fig. 6.** ABC, BIP, MST with 10 and 15 nodes

Increasing the number of nodes we start observing some relevant difference between the heuristics. From Figure 6, using 10 and 15 nodes, the bad performance of the classic MST algorithm are already evident. This clearly shows again the intuition that *MST* is not the appropriate heuristic to solve the *MEBR* problem. Anyway the properties of the Euclidean *MST* lead to the upper bound of Section 3 that remains the only valid upper bound for all the other heuristics. ABC is, on average, better than BIP but there are some peaks in which BIP is better.



**Fig. 7.** ABC, BIP, MST with 25 and 30 nodes

Experimenting using 25 and 30 nodes (see Figure 7), the fact that ABC performs better than BIP becomes clear as it results in solutions that are, on the average 3.5% better than the ones calculated by BIP. Moreover, even if MST is far from the other heuristics, this distance is nothing compared to the known bounds.

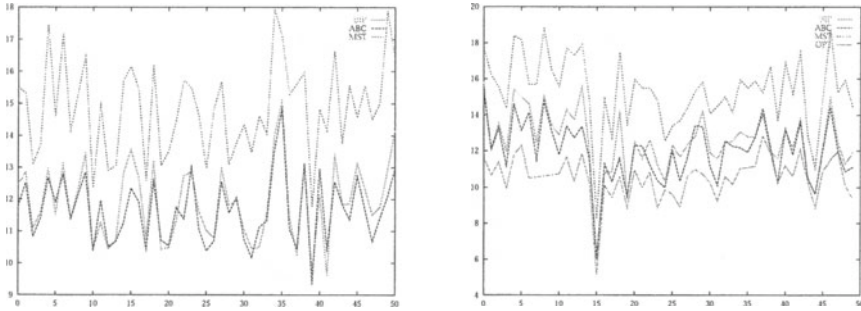


Fig. 8. ABC, BIP, MST with 45 nodes and ABC, BIP, MST, OPT with 50 nodes

Increasing further the number of nodes up to 45 and 50 (see Figure 8) the average of the power saved by ABC with respect to BIP reaches 5%. Moreover, in the last figure we also show the graph of the optimal solution obtained by the integer linear program proposed in Section 5. The optimal solution is, on average, 12% better than ABC so, in practice, on random instances we do not lose much with respect to the optimal and this is in contrast with the discussed upper bound of 12.15. By these experiments we can conclude that the advantage of the 5% obtained with ABC with respect to BIP can be considered to be very significant.

## 7 Conclusion

The aim of the paper was to produce a new heuristic able to solve some evident problems arising from experimental and analytical results about the MEBR problem. We first proved the upper bound for the MST heuristic, with which every other heuristic is compared. Then we showed, by experimental results, that the proposed ABC algorithm computes better solutions over several random instances of various sizes comparing it with the known heuristics of the literature. The performed computations produced solutions which values are on average 5% better than BIP using the same time complexity of  $O(n^3)$ . The main question left open in this paper is if it is possible to reduce the big gap of the ABC bounds. Anyway the new heuristic shows improvements on random instances and makes it difficult to find an ad hoc instance for which ABC outputs a solution worse than a 2-approximation.

We also developed an implementation of the optimal algorithm using integer linear programming. To the best of our knowledge this is the first work where the heuristics are compared with the real costs. With this method we showed that on random instances even the easy MST is not too far from the optimal and hence that our improvement of 5% is a very good result. Another way to try implementing other good approximation algorithms could be to study the relaxed formulation of the integer linear program. By our first tests there are good hopes in this direction. Another interesting issue could be to evaluate the

heuristics for other kind of distributions such as the spatial Poisson one (see [2] for instance).

## References

1. BANERJEE, S., MISRA, A., YEO, J., AND AGRAWALA, A. Energy-efficient broadcast and multicast trees for reliable wireless communication. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)* (2000), IEEE Computer Society.
2. CAGALJ, M., HUBAUX, J. P., AND ENZ, C. Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues. In *Proceedings of MO-BICOM* (2002), pp. 172–182.
3. CLEMENTI, A., CRESCENZI, P., PENNA, P., ROSSI, G., AND VOCCA, P. On the complexity of computing minimum energy consumption broadcast subgraph. In *Proceedings of the 18th Annual Symposium on Theoretical Aspects of Computer Science* (2001), vol. 2010, Lecture Notes in Computer Science, Springer-Verlag, pp. 121–131.
4. CLEMENTI, A., HUIBAN, G., PENNA, P., ROSSI, G., AND VERHOEVEN, Y. C. On the approximation ratio of the mst-based heuristic for the energy-efficient broadcast problem in static ad-hoc radio networks. In *Proceedings of the 3rd IEEE IPDPS Workshop on Wireless, Mobile and Ad Hoc Networks (WMAN)* (2003), p. 222.
5. CLEMENTI, A., PENNA, P., AND SILVESTRI, R. On the power assignment problem in radio networks. *Mobile Networks and Applications (ACM MONET), Special Issue on Discrete Algorithms and Methods for Mobile Computing and Communications 9* (2004), 125–140.
6. DAS, A. K., MARKAS, R. J., EL-SHARKAWI, M., ARABSHAHI, P., AND GRAY, A. Minimum energy broadcast trees for wireless networks: Integer programming formulations. In *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)* (2003), IEEE Computer Society.
7. KASHYAP, A., NISHAR, H., AND AGARWAL, P. Survey on unicast routing in mobile ad hoc networks, <http://citeseer.nj.nec.com/481660.html>, 2003.
8. RAPPAPORT, T. *Wireless communications: principles and practice*. Prentice-Hall, Englewood Cliffs, NY, 1996.
9. ROSSI, G. *The Range Assignment Problem in Static Ad-Hoc Wireless Networks*. PhD thesis, Università degli Studi di Siena, 2002.
10. WAN, P. J., CALINESCU, G., LI, X., AND FRIEDER, O. Minimum energy broadcast routing in static ad hoc wireless networks. *Wireless Networks 8* (2002), 607–617.
11. WIESELTHIER, J. E., NGUYEN, G. D., AND EPHREMIDES, A. On the construction of energy-efficient broadcast and multicast trees in wireless networks. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)* (2000), IEEE Computer Society, pp. 585–594.