Caspian Journal of Mathematical Sciences (CJMS)
University of Mazandaran, Iran
http://cjms.journals.umz.ac.ir
ISSN: 1735-0611

CJMS. **7**(1)(2018), 46-67

Broadcast Routing in Wireless Ad-Hoc Networks: A Particle Swarm optimization Approach

Ahmad Moradi [1]
[1] Department of Computer Science, Faculty of Mathematical Sciences, University of Mazandaran, Babolsar, Iran

Abstract. While routing in multi-hop packet radio networks (static ad hoc wireless networks), it is crucial to minimize power consumption since nodes are powered by batteries of limited capacity and it is expensive to recharge the device. This paper studies the problem of broadcast routing in radio networks. Given a network with an identified source node, any broadcast routing is considered as a directed tree rooted at the source node and spans all nodes.

Since the problem is known to be NP-Hard, we try to tackle it heuristically. First we propose an efficient Particle Swarm Optimization (PSO) based algorithm with a proper coding schema. Then we present the second algorithm which combines the global search of the first algorithm with a local search strategy based on noising methods. Comprehensive experimental study is devoted to compare the behavior of the algorithms and to show its priority over the best known previous results.

Keywords: Particle Swarm Optimization, Broadcast Routing, Wireless Ad Hoc Network, Noising method.

2000 Mathematics subject classification: xxxx, xxxx; Secondary xxxx.

[1] Corresponding author: a.moradi@umz.ac.ir
Received: 25 January 2016
Revised: 11 March 2016
Accepted: 11 March 2016

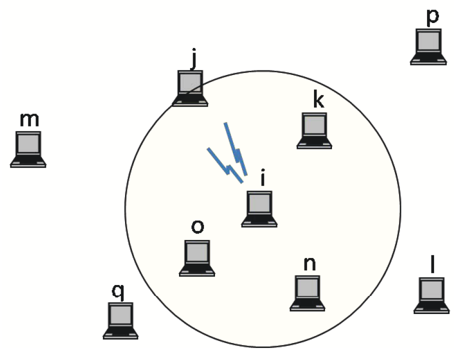

FIGURE 1. Re-transmission to the nodes k, n, o is not needed since they will receive the transmitted message from node i to node j.

1. INTRODUCTION

Due to its potential application in different situations such as battlefield, emergency relief, environment monitoring, and so on, wireless ad hoc network have emerged as premier research topic [1, 2]. Network elements in wireless ad hoc network are powered by batteries only and it is expensive and sometimes impossible to recharge the device [3]. Having such limitation over energy resources, it is crucial to design algorithms of minimum power consumption for typical network tasks such as fault-tolerance [3] and broadcast/multicast transmission [4,5].

Wireless ad hoc network is considered as a set of simple wireless nodes powered by batteries of limited capacity. Nodes are equipped by Omni-directional antenna - the case considered in this paper - and this is interesting because a single transmission could reach several nodes within its vicinity. This feature called *wireless multicast advantage* [6] is extremely useful for energy efficient broadcast routing. Figure .1 illustrates the so-called wireless multicast advantage in wireless networks.

Nodes in wireless ad hoc network can forward messages as well as initiate them. In such setting, a node could transmit a message to nodes which are not within its transmission range. The message must be routed to a destination through a multi-hop path. This is why wireless ad hoc network are also called *multi-hop* networks. In such a multi-hop network it is possible to transmit messages to other nodes without having every node transmitting at maximum power. In order to change its transmission range, each node has also the capability of varying the power with which it transmits a message. This gives the network *adjustable energy model* [7]. Since we are supposed to determine the transmission rang for each node, using the adjustable energy model will lead better results on power consumption at the cost of solving much harder problems.

In a wireless ad-hoc network, while a node is transmitting a message to another node a one-hop communication link will be established to every other node within its transmission range. So network topology of a wireless ad hoc network could be represented by a directed graph. For such a network with an identified source node, the minimum power broadcast (MPB) problem is to assign transmission range to the network nodes so that communication from source node to all other nodes is possible (directly or by hopping) and the total power consumption is minimized. The MPB problem is known to be NP-Hard even in the Euclidean plane [8,9]. The following paragraph will summarized important attempts proposed in the literature to tackle the MPB problem.

Wicselthier et al. [5] developed the so-called Broadcast Incremental Power (BIP) algorithm based on the observation that node based approaches are more suited for wireless environment. Similar to Prim's well-known minimum spanning tree algorithm [10], the BIP algorithm begins with the source node and in each iteration a single node with minimum incremental power will be added to the current subtree rooted at the source node. The algorithm terminates while it added all the network nodes and reports the rooted tree as broadcast tree. The BIP algorithm has an approximation ratio between 13/3 and 12 [4]. Das et al. [11] proposed an algorithm called CM to solve the MPB problem using ant colony optimization approach. The algorithm first clusters the network nodes to low-power rooted subtrees and then uses ant systems to merge root nodes in order to obtain a broadcast tree. Simulated annealing meta-heuristic was applied by [12] and [14] to design simple algorithms of better performance. The algorithm proposed by [12] is in fact the probabilistic version of r-shrink tree-improvement greedy heuristic proposed by Das et al [13]. Hashemi et al. [14] developed an algorithm called ESA which uses simulated annealing meta-heuristic with special neighborhood structure. For a transmitting node v in the current broadcast tree, they define the following two utility measures and try to find a neighbor broadcast tree by perturbing them. The first one is the ratio obtained by dividing power consumed by the node v to the total power consumption by the current broadcast tree. The second one is the ratio obtained by dividing the number of nodes receiving a message directly from v to the maximum number of nodes that can receive a message directly from a transmitting node. The ESA algorithm has best known mean results for the MPB problem.

Particle Swarm Optimization (PSO) developed by [15,16] is a meta-heuristic optimization approach inspired by the observation of the social behavior of animals, such as bird flocking, fish schooling. Due to the simple concept and easy implementation, the particle swarm optimization has now attracted lots of interests around the globe. Swarm intelligence based approaches including Ant Colony Optimization, PSO and ,... are widely used to successfully solve different routing problems in wireless ad hoc networks. The reader is referred to [33,34,35,36] for some related literature and also to the paper [32] in which a large number of publications dealing with PSO applications is categorized. To the best of our knowledge the MBP problem (As considered and defined in this paper in section 2) has not yet been tackled by particle swarm optimization. This paper is intended to make an efficient algorithm based on the particle swarm optimization to solve the MPB problem. The algorithm applies the standard PSO [15] in which a weighted coding of broadcast trees is used to represent particles of the swarm. The algorithm is then hybridized with a local search strategy based on the so-called noising method [24] to intensify the search around good regions.

The rest of the paper is organized as follows. Section .2 gives some basic concepts and definitions of the underlying network model and the MPB problem. Section .3 and Section .4 describes the PSO algorithm for the MPB problem in details. Section .5 hybrids the PSO with the noising method while Section .6 performs a comprehensive computational study.

2. Network Model and Problem Definition

In this paper we consider a static multi-hop ad hoc network which uses omni-directional antenna for its nodes. The model was previously used by [14,18]. In this model each node is able to adjust its transmitting power for the purpose of energy conservation. The nodes have also the same transmission efficiency and the same power threshold for signal detection [19,20]. Based on the common power attenuation model [19], the power *P(u)* required by the node u to transmit data to another node v must satisfy the inequality

$$P(u) \geq dist(u,v)^{\beta} \tag{2.1}$$

where *dist(u,v)* is the Euclidean distance between u, v and the parameter $\beta \geq 1$ is an environment dependent parameter called *distance-power gradient* (also referred to as *power-attenuation exponent*). In an ideal environment it holds that $\beta = 2$ but it may vary from 1 to 6 depending on the environment conditions of the place the network is located on [20]. The equation 2.1 comes from the fact that the signal power falls as $1/d^{\beta}$ at the distance d from the transmitting node.

Let the weighted directed graph $G = (V, E, dist)$ represents a wireless ad hoc network with the set V of network nodes, the set E of possible communication links and the positive distance function $dist : E \to R^+$. A function $r : V \to R^+$ is called a *range assignment* over the network. We say a directed link from node u to node v could be established under the range assignment r, if $r(u) \geq dist(u, v)$. A subgraph $G' = (V, E', dist)$ of the network graph G, where $E' \subseteq E$, defines a range assignment $r_{E'}$ where

$$r_{E'}(i) = max_{j|(i,j)\in E'}\{dist(i, j)\} \tag{2.2}$$

and

$$P_{r_{E'}}(i) = r_{E'}(i)^{\beta}, i \in V \tag{2.3}$$

as the power consumed by node i under the range assignment $r_{E'}$. The power cost of G' is the total power consumed through its corresponding rang assignment $r_{E'}$ i.e.

$$P(G') = \sum_{i\in V} P_{r_{E'}}(i) = \sum_{i\in V} r_{E'}(i)^{\beta}. \tag{2.4}$$

Since any connected graph contains a spanning tree, the MPB problem can be restated as follows: Let $G = (V, E, dist)$ be a directed complete graph with an identified source node s, find a spanning tree T^* with

$$P(T^*) = min\{P(T) : T \in \mathbf{ST}(\mathbf{s})\} \tag{2.5}$$

where $\mathbf{ST}(\mathbf{s})$ is the set of all spanning trees over V rooted at the source node s.

3. Particle Swarm Optimization

Over the years, metaheuristic algorithms have been widely used as robust techniques for solving hard combinatorial optimization problems. Their behavior is directed by the evolution of a population in the search for an optimum.

Particle Swarm Optimization (PSO) is a metaheuristic proposed by Kennedy and Eberhart [16]. It has been applied with success in many areas and appears to be a suitable approach for several optimization problems [21]. Similar to Genetic Algorithms, PSO is a population-based technique, inspired by the social behavior of individuals (or particles) inside swarms in nature (for example, flocks of birds or schools of fish). However, unlike Genetic Algorithms, it has no crossover and mutation operators and is easy to implement, requiring few parameter settings and computational memory.

3.1. **Standard PSO algorithm.** The standard PSO considers a swarm S containing n particles ($S = 1, 2, ..., n$) in a d-dimensional continuous solution space [21]. Each ith particle of the swarm has a position $\mathbf{x_i} = (x_{i1}, x_{i2}, ..., x_{ij}, ..., x_{id})$, and a velocity $\mathbf{v_i} = (v_{i1}, v_{i2}, ..., v_{ij}, ..., v_{id})$. The position $\mathbf{x_i}$ represents a solution to the problem, while the velocity $\mathbf{v_i}$ gives the rate of change in the position of particle i at the next iteration. Indeed, considering iteration k, the position of particle i is adjusted according to

$$x_i^k = x_i^{k-1} + v_i^k. \tag{3.1}$$

where x_i^k and v_i^k are position and velocity of the ith particle in kth iteration.

Each particle i of the swarm communicates with a social environment or neighborhood, $N(i) \subseteq S$,[2] representing the group of particles with which it communicates, and which could change dynamically. In nature, a bird adjusts its position in order to find a better position, according to its own experience and the experience of its companions. In the same manner, considering iteration k of the PSO algorithm, each particle i updates its velocity reflecting the attractiveness of its best position so far ($\mathbf{g}$) and the best position ($\mathbf{b_i}$) of its social neighborhood $N(i)$, according to the equation:

$$v_i^k = c_1\xi v_i^{k-1} + c_2\xi(b_i - x_i^{k-1}) + c_3\xi(g_i - x_i^{k-1}). \tag{3.2}$$

The parameters c_i are positive constant weights representing the degrees of confidence of particle i in the different positions that influence its dynamics, while the term ξ refers to a random number with uniform distribution [0, 1] that is independently generated at each iteration.

In order to control excessive roaming of particles outside the search space, the value of each element v_{ij} of $\mathbf{v_i}$ (by equation 3.2) is kept within the range $[v_{min}, v_{max}]$. The swarm fly toward new positions according to equation 3.1 and the PSO evaluate the desired optimization fitness function of the new positions of the particles. This process is repeated until a user-defined stopping criterion is reached. The stopping criterion is usually a sufficiently good fitness or a maximum number of iterations. A short pseudo-code of the standard PSO is given in the Table .1.

4. Coding of Spanning Trees

For a PSO algorithm parameter setting affects its performance, but the fundamental design choice of the algorithm is the coding by which its

[2] In this paper we apply the *Global Best PSO* in which $N(i)$ is considered to be $S \backslash i$, that is the particle i of the swarm will communicate information with all other particles in the swarm.

Table 1. PSO algorithm

```
Algorithm Standard PSO
{
  Initialize a population of particles with random
    positions and velocities in the search space.
  repeat
  {
      For i-th particle,
      {
        Evaluate the desired optimization fitness
          function and update each b_i (if applicable). }
        Identify the particle with the best success so far
          and update the variable g_i (if applicable).
        Change the velocity and position of the particle
          according to the equation 3.2 and equation 3.1.
  } until a pre-specified stopping condition is met
}
```

particles represent candidate solutions [22]. Most researchers [23] agree on the relevance of the following features of evolutionary coding:

- solution representation should not require extravagant amount of memory
- time considered of evaluating a solution or decoding a particle to represent a spanning tree should be small
- each position of a particle should represent feasible solutions
- representation of a solution should be equally likely, though bias may be an advantage if the favored solutions are near-optimal

In this study, to address the above mentioned features, we propose the following coding approach based on the idea of *weighted coding* [31] and *priority based coding* [29,30] to relate position of each particle in PSO to a broadcast routing tree.

4.1. **Coding and Decoding.** Here, spanning trees are encoded as strings of numerical weights associated with the network nodes. The numerical weights are in fact positions of particles in the PSO. Let $\mathbf{x_i} = (x_{i1}, ..., x_{iu}, ..., x_{iv}, ..., x_{id})$ be the position of i-th particle in the swarm, we give different meaning to the numerical weights. The node weight x_{ip} of the node p will be an indicator of the priority of the node p to decrease its transmitting range while creating a broadcast tree. The BIP algorithm is then modified to interpret the weights and to decode the string of numerical weights to a spanning tree as follows.

Similar to the BIP algorithm the modified version starts with the source node but in each iteration a single node with minimum incremental "biased" power will be added to the current subtree rooted at the source node. Let C be the set of network nodes spanned by the subtree T_C at the current iteration of the modified BIP algorithm. A new arc (u, v) with $u \in C$ and $v \in V \setminus C$ will be added to the T_C provided that

$$\triangle P(T_C, (u, v)) = min_{p \in C, q \in V \setminus C}\{\triangle P(T_C, (p, q))\} \tag{4.1}$$

where $\triangle P(T_C, (p, q))$ is the *biased power change* in the subtree T_C after adding (p, q) obtained by the

$$\triangle P(T_C, (p, q)) = \begin{cases} 0, \text{if } P_{T_C}(p) > dist(p, q)^{\beta} \\ x_{ip}.(dist(p, q)^{\beta} - P_{T_C}(p)), \quad \text{O.w} \end{cases} \tag{4.2}$$

where $P_{T_C}(p)$ is the power consumed by the node p in T_C.

Clearly, using these equations, the modified BIP algorithm will give the potential of adding more nodes to the node p if it has more priority (less value of x_{ip}) according to the position of the i-th particle of the swarm. The illustrated examples below will explain the coding/decoding process in more details.

In Figure .2, we depicted a network of size 10 generated by choosing 10 points uniformly at random from a grid of size 100×100. Given labels 1 to 10 to the nodes, the node 1 is considered as the root node. The node's ($\mathbf{x}$,$\mathbf{y}$)-coordinates in the grid are (3,24), (9,97), (24,37), (27,55), (46,99), (49,68), (58,40), (74,63), (81,53), (93,49) accordingly. Let ($\mathbf{x}_i$,$\mathbf{y}_i$) be the ($\mathbf{x}$,$\mathbf{y}$)-coordinates of node i, we set $\beta = 2$ and obtain the power consumed by the node i to establish a link to the node j as

$$dist(i, j)^{\beta} = (\mathbf{x}_j - \mathbf{x}_i)^2 + (\mathbf{y}_j - \mathbf{y}_i)^2.$$

Figure .3 shows the tree obtained by the BIP algorithm with power cost 4310. The performance of the BIP algorithm is equal to the performance of the modified BIP algorithm with the position

$$\mathbf{w_0} = (1, 1, 1, 1, 1, 1, 1, 1, 1, 1).$$

Now consider the position

$$\mathbf{w_1} = (1.5188, 2.2881, 2.0726, 1.1357, 1.5921, 2.2547, 1.2157, 1.4802, 1.6621, 1.1974).$$

Performing the modified BIP algorithm over w_1 decodes the position to a tree shown in figure .4 with better power cost equal to 4185. Even

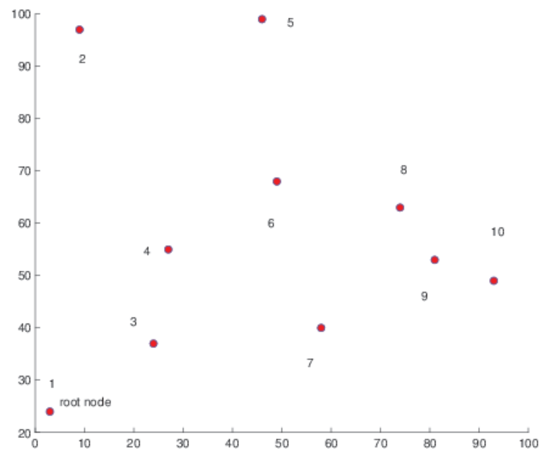

FIGURE 2. Random network with 10 nodes.

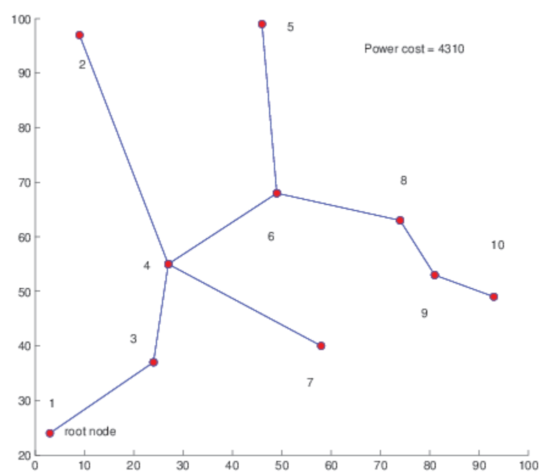

FIGURE 3. Broadcast tree obtained by the BIP algorithm.

better power cost will be obtained while performing the modified BIP over the position

$$\mathbf{w_2} = (1.3176, 2.8751, 3.0655, 1.1159, 1.6319, 2.2069, 1.5987, 1.5199, 2.1274, 2.2251).$$

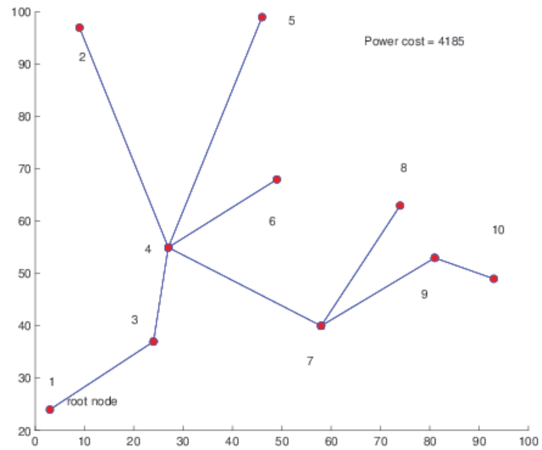

FIGURE 4. Broadcast tree obtained by the modified BIP according to the weight $\mathbf{w_1}$.

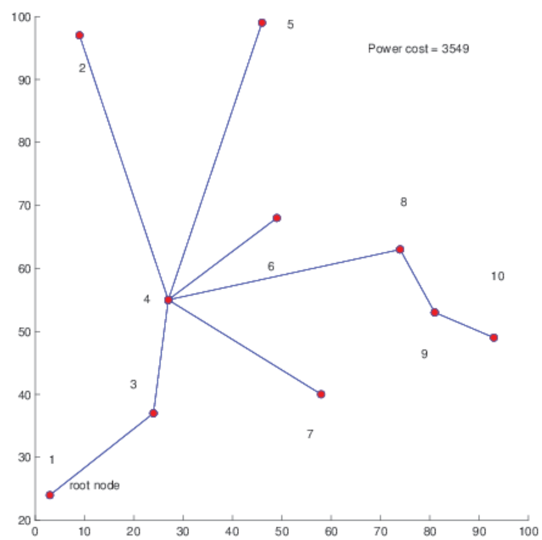

FIGURE 5. Broadcast tree obtained by the modified BIP according to the weight $\mathbf{w_2}$.

As shown in Figure.5, the tree has power cost equal to 3549. Note that this gives the 22% improvement over the power cost of the solution obtained by the BIP algorithm.

Having such weighted coding scheme, the PSO algorithm is now able to solve the MPB problem. As particles flown searching through hyperspace, their positions will be mapped to broadcast trees using the modified BIP algorithm and their power cost are then evaluated to determine good positions as well. This leads to a promising population-based search method over the space of broadcast trees.

5. Hybrid of PSO and noising method

Often an algorithm with stochastic search behavior can be improved by incorporating problem specific heuristics or local search optimization techniques [23]. For the proposed PSO algorithm, we use *noising method*, the recent optimization metaheuristic which has been successfully applied to different combinatorial optimization problems [24,25,26,27,28]. The PSO algorithm contributes to hybrid approach in a way to insure that the search is less likely to be trapped in a local optima, while the noising method makes the search to perform more exploitation in good regions. In fact, the hybrid approach can usually exploit a better trade-off between exploration efforts and global optimality of the solution found. This section, first describes the noising method in more details and then shows how the PSO algorithm could be hybridized by the noising method properly.

5.1. **Noising method: Design.** In order to use noising method for a specific optimization problem, an appropriate state space corresponding to the possible feasible solutions and a neighborhood-relation between the states are needed. The role of such relation is to express the similarity between the elements of the state space. The neighborhood of a state is typically defined as the set of the states that can be obtained by making some kind of local modifications on the current state. Given the source node s, the state space of the noising method is the set of all possible spanning trees rooted at s and the cost of a state is the power cost of it as defined in Section 2. For such a state T with corresponding position $\mathbf{x_i}$ a neighbor state T' is generated using the following procedure

1. chose a random network node u and increase/decrease[3] its priority x_{iu} by λ [4]. Denote the new position as $\mathbf{x_i'}$.
2. perform the modified BIP algorithm as stated in section 4.1 to obtain T'.

[3]The noising method decides to increase or decrease the x_{iu} at random

[4]λ is a constant parameter over the execution of the noising method

The idea behind the definition of such neighborhood relation between to broadcast trees is to let the noising method focuss on each node separately and to increase/decrease its power consumption by decreasing/increasing its priority at the aim of properly adjust its power consumption.

Performing step .2 above could be done more efficiently. Suppose that during the execution of the modified BIP algorithm over the network with the position $\mathbf{x_i}$, the network node u is the kth added node in the current tree. Using the equation 4.2, it is easy to see that the first k steps of performing the modified BIP over $\mathbf{x'_i}$ will have the same results as the first k steps of performing the algorithm over the network with the position $\mathbf{x_i}$. In average case, this leads to a significant improvement of performing step .2 above.

starting from a solution, T, obtained by the PSO, in each iteration the noising method generates a random neighbor, T', of the current state and performs the following noising process to be able to scape local minima. Instead of considering the value $\triangle P(T, T')$ the algorithm uses noised variation $\triangle P_{noised}(T, T')$ by adding a noise to $\triangle P(T, T')$,

$$\begin{aligned} &\triangle P_{noised}(T, T') = \\ &\quad \triangle P(T, T') + \rho = P(T') - P(T) + \rho \end{aligned} \tag{5.1}$$

where ρ is changing at each iteration. The new neighbor is then accepted as the current state if it satisfies the following equation

$$\triangle\, P_{noised}(T, T') = P(T') - P(T) + \rho < 0 \tag{5.2}$$

Usually, the noise is chosen from an interval containing negative values as well as positive values. As a consequence, it may accept a bad neighbor (a neighbor with more power consumption), but also it may reject a neighbor yielding a decrement of power consumption.

To intensify the search in good regions, we only perform the noising method when the value of $\mathbf{g}$ is updated. Together with the population-based search characteristic of the PSO which makes the search more diversified, the single-based neighborhood search of the noising method makes the search more intensified in good regions.

5.2. **Noising method: Implementation Issues.** For the proposed noising method, the following parameters need to be set. The noising method will be stopped when a predefined number of iterations, N is completed. We set $N = n$ where n is the node size of underlying ad-hoc network. As it was also mentioned above, the noises are usually drawn from an interval $[-r, +r]$ where r is called *noise rate*. The noise rate decreases during the running of the noising method and its initial value depends on the problem specific data (here the maximum power consumption needed to establish a link between a pair of network nodes).

TABLE 2. Hybrid PSO algorithm

```
Algorithm Hybrid PSO
{
  /* Initialization */
  For each particle i
  {
    Initialize x_i and v_i at random.
    Determine corresponding broadcast tree rooted at the source node s based
      on the modified BIP algorithm.
    Evaluate its power consumption.
  }
  Initialize b_i, g and their positions x_i^b and x^g
  Initialize r_max, r_min.
  repeat
  {
      For each particle i
      {
        Change the velocity and position of the particle
          according to the equattion 3.1 and equation 3.2.
        Determine corresponding broadcast tree rooted at s based on the modified
          BIP algorithm.
        Evaluate the desired power consumption
          and update b_i and x_i^b (if applicable).
      }
      Identify the particle with the best success so far
        and update the variable g and x^g (if applicable).
      If x^g is updated
      {
        Perform the noising method starting with x^g without changing the noise rate.
        Decrease the noise rate.
      }
  }until the evaluation value of g keeps fixed at a pre-specified consecutive iterations
  return the best solution x^g and its corresponding broadcast tree.
}
```

The value of the decreasing rate is obviously linked to the number N of iterations. Set r_{max} and r_{min} to be the extremal values for the noise rate. As it was also stated by [17] we let the noising method decreases the noise rate by $(r_{max} - r_{min})/N$ after each iteration.

5.3. **The hybrid algorithm for the MBP problem.** Based on the proposed decoding/coding of broadcast trees and the local search approach established using the noising method, the framework of a hybrid PSO algorithm for the MBP problem is proposed and illustrated in Table. 2.

It can be seen that the Hybrid PSO (HPSO) not only applies the PSO-based evolutionary searching mechanism to effectively perform exploration for promising solutions within the entire region, but also applies the noising method to perform exploitation in promising sub-regions. Since both exploration and exploitation are stressed and balanced, it is expected to achieve good performance for the problem. In the next section, we will investigate the performance of Hybrid PSO based on simulation and comparisons.

6. Computational Study

We have evaluated the performance of our algorithms for many network examples comparing with the best previous results. The experiments are run on randomly generated test cases. For each instance size n between 15 and 100, in increments of 5, an instance of size n is generated by choosing n points uniformly at random from a grid of size 100 $\times$ 100. One of the nodes is randomly chosen to be the source node of network. We have considered power attenuation exponent $\beta = 2$ in all cases (i.e. for a specified network size and algorithm), our results are based on the performance of 50 runs over the generated instance. All our computational experiments were performed on a Pentium with 1.6 GHz. The codes were written in C++ and run under Windows XP.

6.1. **Comparting PSO and HPSO.** In this section, we give numerical results to compare the HPSO algorithm with the standard PSO algorithm without noise. The results clearly shows that using noises together with the PSO algorithm has a significant influence on the performance of the PSO algorithm leading to broadcast trees of much lower power costs.

In the Hybrid PSO algorithm, we have used the following parameters; the swarm size $|S| = 30$, $c_1 = 0.9$, $c_2 = c_3 = 2$, and for each instance $r_{min} = -dist_{avg}$, $r_{max} = dist_{avg}$ is where $dist_{avg}$ is the average distance over the set of all possible distances between any pair of network nodes generated by the instance. We also set $v_{min} = 0.01$, $v_{max} = 5$, $\lambda = 0.5$ and the stopping condition of the hybrid PSO is that the evaluation value of g keeps fixed at 20 consecutive iterations. Parameters for the PSO have also chosen similar to their corresponding parameter in the HPSO. These values are fixed and considered to be our standard parameter

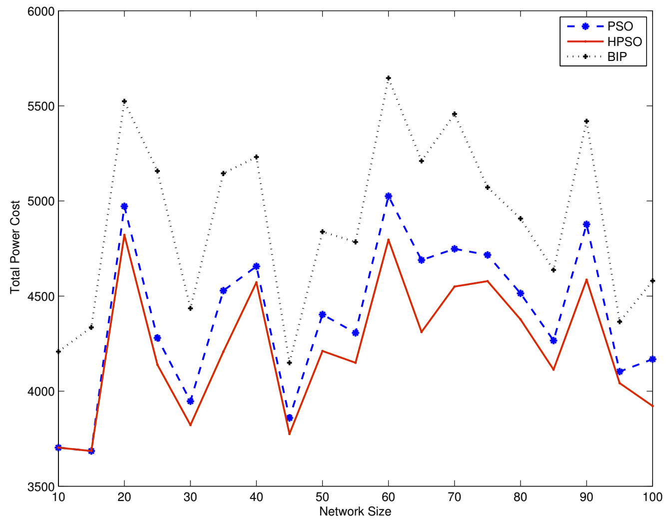

FIGURE 6. Total power cost obtained by the HPSO, PSO and the BIP algorithm over different network sizes.

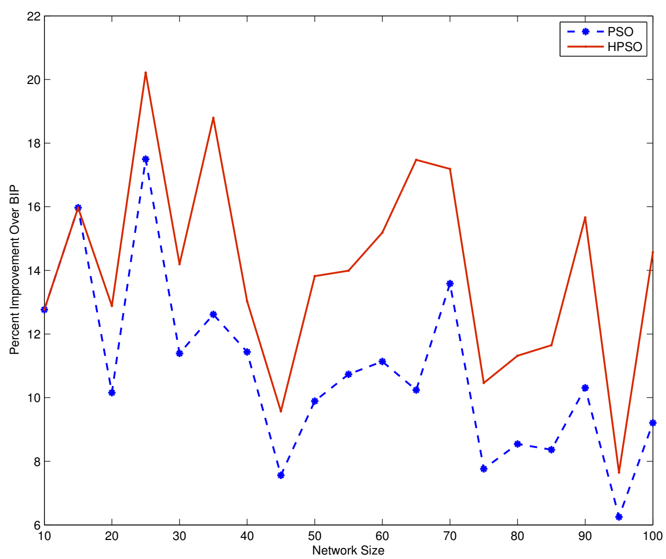

FIGURE 7. Percent improvement over the BIP algorithm.

TABLE 3. Performance of the hybrid PSO algorithm compared with the PSO algorithm

n	BIP	PSO, (%)	HPSO, (%)
10	4208	3703 , (12.7643)	3703 , (12.7643)
15	4335	3685 , (15.9679)	3685 , (15.9679)
20	5524	4972 , (10.1563)	4821 , (12.8848)
25	5157	4280 , (17.4988)	4139 , (20.2167)
30	4435	3947 , (11.3901)	3822 , (14.1964)
35	5145	4528 , (12.6174)	4208 , (18.7929)
40	5231	4656 , (11.4343)	4572 , (13.0322)
45	4149	3859 , (7.5574)	3775 , (9.5697)
50	4838	4403 , (9.8918)	4211 , (13.8211)
55	4784	4306 , (10.7359)	4149 , (13.9905)
60	5646	5025 , (11.1369)	4796 , (15.1866)
65	5210	4689 , (10.2388)	4311 , (17.4748)
70	5457	4748 , (13.5838)	4550 , (17.1875)
75	5071	4716 , (7.7611)	4578 , (10.4602)
80	4907	4514 , (8.5419)	4377 , (11.3176)
85	4637	4266 , (8.3621)	4113 , (11.6487)
90	5419	4877 , (10.3107)	4586 , (15.6623)
95	4365	4103 , (6.2508)	4042 , (7.6446)
100	4580	4168 , (9.2077)	3922 , (14.5664)

setting all over our computational results both in this and the next section.

Using the above mentioned parameter we have reported the *best* results obtained by the PSO algorithm and the *worst* results obtained by the HPSO (over 50 runs on randomly generated instances as we have mentioned before) as well as their percent improvement over the BIP algorithm in Table 3. Similar results are also depicted in Figure 6 and Figure 7. This demonstrate that the use of noising method together with the PSO algorithm will comprehensively improve the results obtained by the PSO algorithm. That is because using small noises, the PSO algorithm now has the the capability of better exploiting good regions.

6.2. **Comparing HPSO and other algorithm.** As we have stated before, Our performance metric is the power cost of a broadcast tree. To facilitate the comparison of our algorithms over a wide range of network examples, we present our results in terms of the normalized power

Table 4. Performance of the hybrid PSO algorithm compared with best previous algorithms

n	BIP	CM	SA	ESA	hybrid PSO
15	1.1695	1.1464	1.1235	1.0869	1.0451
20	1.2928	1.2318	1.2000	1.1669	1.1158
25	1.2084	1.1313	1.1116	1.0897	1.0499
30	1.2361	1.1708	1.1495	1.1261	1.0892
35	1.2586	1.2235	1.1915	1.1523	1.1188
40	1.3297	1.2802	1.2581	1.2386	1.1931
45	1.2352	1.1834	1.1456	1.0962	1.0509
50	1.3369	1.2446	1.1984	1.1536	1.1117
55	1.2143	1.1825	1.1482	1.0938	1.0618
60	1.2862	1.2058	1.1819	1.1240	1.0867
65	1.2799	1.2140	1.1825	1.1288	1.0939
70	1.2570	1.2098	1.1939	1.1792	1.1425
75	1.2857	1.2498	1.2078	1.1905	1.1499
80	1.2800	1.2284	1.1875	1.1542	1.1121
85	1.1921	1.1407	1.1044	1.0609	1.0229
90	1.1830	1.1543	1.1348	1.0842	1.0391
95	1.3398	1.2746	1.2413	1.2006	1.1567
100	1.1908	1.1554	1.1428	1.0821	1.0354

[6,14] for each network example. For each individual network example, say network $\mathtt{m}$, we compute the mean power cost associated with the broadcast tree generated by each of the algorithms over 50 runs. To determine a benchmark for each network instance let $\mathtt{I}$ be the set of algorithms, we define $Q_i(\mathtt{m})$ to be the mean power cost of broadcast tree for instance $\mathtt{m}$ generated by algorithm $\mathtt{i} \in \mathtt{I}$, over 50 runs and $Q_{\mathtt{best}}(\mathtt{m})$ to be the power cost of best solution found for instance $\mathtt{m}$. Thus, $Q_{\mathtt{best}}(\mathtt{m})$ is the power of the lowest-power tree among the set of algorithms (for the particular network instance $\mathtt{m}$). We then define the normalized mean power associated with algorithm $\mathtt{i}$ to be

$$Q'_{\mathtt{i}}(\mathtt{m}) = Q_{\mathtt{i}}(\mathtt{m})/Q_{\mathtt{best}}(\mathtt{m}) \tag{6.1}$$

this metric provides a measure of how close each algorithm comes to provide the lowest-power tree.

Now, we will report on a large number of experiments we performed to test our algorithms. Table 4 compares the performance of our algorithm with best previous results. The first column is the size of the network and

TABLE 5. Percent improvement of CM, SA, ESA and the hybrid PSO corresponding to results reported in Table. 3

n	CM	SA	ESA	hybrid PSO
15	1.9695%	3.9311%	7.0574%	10.6349%
20	4.7182%	7.1757%	9.7393%	13.6923%
25	6.3857%	8.0152%	9.8226%	13.1196%
30	5.2797%	7.0058%	8.8996%	11.8874%
35	2.7903%	5.3330%	8.4474%	11.1059%
40	3.7184%	5.3816%	6.8527%	10.2692%
45	4.1958%	7.2567%	11.2493%	14.9175%
50	6.9045%	10.3612%	13.7158%	16.8510%
55	2.6133%	5.4389%	9.9187%	12.5549%
60	6.2487%	8.1084%	12.6079%	15.5076%
65	5.1528%	7.6129%	11.8056%	14.5361%
70	3.7566%	5.0260%	6.1941%	9.1156%
75	2.7928%	6.0586%	7.4026%	10.5627%
80	4.0269%	7.2260%	9.8240%	13.1148%
85	4.3065%	7.3509%	11.0052%	14.1914%
90	2.4272%	4.0790%	8.3512%	12.1643%
95	4.8654%	7.3513%	10.3894%	13.6665%
100	2.9762%	4.0325%	9.1324%	13.0478%

the second one is the total power cost obtained by the BIP algorithm. In the next three columns we reported the normalized mean power obtained by CM algorithm [11], previous SA [12] and the ESA algorithm [14]. Similar results associated with the proposed hybrid PSO algorithm are also reported in the last column respectively. In all of the test cases the proposed hybrid PSO algorithm alone can improve the power cost obtained by CM and SA and ESA algorithms.

Corresponding to the results in Table 4, the Table 5 shows the percent improvement of the normalized mean power consumption over the BIP algorithm. The best results can be found by the hybrid PSO algorithm which gave an improvement of 9%, in extreme cases even of 17% compared to the BIP algorithm. Since all the SA, CM, ESA and the hybrid PSO algorithms are algorithms with stochastic search behavior, we have to analyzed the statistical behavior of the algorithms. To that end, variance of their results are also computed and reported in Table 6. The small variance values of the hybrid PSO algorithm also prove the stability of the algorithm compared with the other ones.

Table 6. Variance of results obtained by the CM, SA, ESA and hybrid PSO algorithms

n	CM	SA	ESA	hybrid PSO
15	0.0039	0.0043	0.0041	0.0036
20	0.0126	0.0125	0.0117	0.0109
25	0.0208	0.0236	0.0222	0.0177
30	0.0467	0.0472	0.0444	0.0420
35	0.0609	0.0710	0.0668	0.0525
40	0.0383	0.0383	0.0408	0.0424
45	0.0288	0.0311	0.0332	0.0308
50	0.0473	0.0463	0.0436	0.0431
55	0.0407	0.0409	0.0437	0.0370
60	0.0187	0.0193	0.0205	0.0164
65	0.0414	0.0418	0.0445	0.0414
70	0.0712	0.0819	0.0777	0.0803
75	0.0647	0.0664	0.0797	0.0644
80	0.0585	0.0598	0.0631	0.0525
85	0.0546	0.0574	0.0549	0.0518
90	0.0601	0.0681	0.0644	0.0564
95	0.0218	0.0251	0.0267	0.0220
100	0.0324	0.0326	0.0346	0.0286

7. conclusion

In this paper we have presented a hybrid PSO-based algorithm to solve the Minimum Power Broadcast (MPB) problem. Due to the hybridization of PSO and a local search, based on the noising method, searching behaviors can be enriched, searching ability can be enhanced, and exploitation and exploration are well balanced. Simulation and comparisons also demonstrate the effectiveness and robustness of the hybrid PSO algorithm. Using these results, we have then shown that the algorithm is competitive when compared with other state-of-the-art techniques for problems of real world size.

References

[1] A.E.F. Clementi, G. Huiban, P. Penna, G. Rossi, Y.C. Verhoeven, Some recent theoretical advances and open questions on energy consumption in ad-hoc wireless networks, in: Proceeding 3rd Workshop on Approximation and Randomization Algorithm in Communication Network (ARACNE), 2002, pp. 23 - 38.

[2] V. Rodoplu, T.H. Meng, Minimum energy mobile wireless networks, IEEE Journal on Selected Areas in Communications 17 (8) (1999) 1633 - 1639.
[3] M.T. Hajiaghayi, N. Immorlica, V.S. Mirrokni, Power optimization in fault-tolerant topology control algorithms for wireless multihop networks, in: Proceedings of the 9th Annual International Conference on Mobile Computing and Networking, 2003, pp. 300 - 312.
[4] P.J. Wan, G. Calinescu, X.Y. Li, O. Frieder, Minimum-energy broadcast routing in static ad hoc wireless networks, in: Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), 2001, pp. 1162 - 1171.
[5] J.E. Wieselthier, G.D. Nguyen, A. Ephremides, 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, pp. 585 - 594.
[6] J.E. Wieselthier, G.D. Nguyen, A. Ephremides, Energy-efficient broadcast and multicast trees in wireless networks, Mobile Networks and Applications, 7, 2002, 481 - 492.
[7] Li, Xiang-Yang, and Ivan Stojmenovic. "Broadcasting and topology control in wireless ad hoc networks." Handbook of Algorithms for Mobile and Wireless Networking and Computing (2004): 239-264.
[8] A.E.F. Clementi, P. Crescenzi, P. Penna, G. Rossi, P. Vocca, On the complexity of computing minimum energy consumption broadcast subgraphs, in: Symposium on Theoretical Aspects of Computer Science, 2001, pp. 121 - 131.
[9] A.E.F. Clementi, P. Penna, R. Silvestri, On the power assignment problem in radio networks, Electronic Colloquium on Computational Complexity (ECCC) (054) (2000).
[10] Ahuja, R.K., Magnanti, T.L., Orlin, J.B., 1993. Network Flows. Prentice Hall, Englewood Cliffs, NJ.
[11] A.K. Das, R.J. Marks, M. El-Sharkawi, P. Arabshahi, A. Gray, A cluster-merge algorithm for solving the minimum power broadcast problem in large scale wireless networks, in: Proceedings of the Milcom 2003 Conference, Boston, MA, 2003, pp. 13 - 16.
[12] A.K. Das, R.J. Marks, M. El-sharkawi, P. Arabshahi, A. Gray, The minimum power broadcast problem in wireless networks: a simulated annealing approach, in: Proceeding IEEE Wireless Communications and Networking Conference (WCNC), 2005, pp. 2057 - 2062.
[13] A.K. Das, R.J. Marks, M. El-Sharkawi, P. Arabshahi, A. Gray, r-shrink: a heuristic for improving minimum power broadcast trees in wireless networks, in: Proceedings of the IEEE Globecom 2003 Conference, San Francisco, CA, 2003, pp. 1 - 5.
[14] S. M. Hashemi, M. Rezapour, A. Moradi, Two new algorithms for the Min-Power Broadcast problem in static ad hoc networks, Applied Mathematics and Computation, 190, 2007, pp. 1657 - 1668
[15] R.C. Eberhar, J. Kennedy, A new optimizer using particle swarm theory, Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 1995, 39 - 43.
[16] J. Kennedy, R.C. Eberhart, Particle swarm optimization, Proceedings of the IEEE International Conference on Neural Networks, Piscataway, NJ, USA, 1995, 1942 - 1948.

[17] I. Charon, O. Hudry, The noising methods: a generalization of some meta heuristics, European Journal of Operational Research, 135, 2001, pp. 86 - 101.

[18] E. Althaus, G. Calinescu, I.I. Mandoiu, S. Prasad, N. Tchervenski, A.Z. Zelikovsky, Power efficient range assignment in ad-hoc wireless networks, in: Proceeding IEEE Wireless Communications and Networking Conference (WCNC), 2003, pp. 1889 - 1894.

[19] T.S. Rappaport, Wireless Communications: Principles and Practices, Prentice Hall, 1996.

[20] K. Pahlavan, A. Levesque, Wireless information network, Wily-Interscience, New York, 1995.

[21] J. Kennedy, R. Eberhart, Swarm Intelligence. Morgan Kaufmann Publishers, San Francisco, CA, 2001.

[22] G. R. Raidl, An efficient evolutionary algorithm for the degree-constrained minimum spanning tree problem, in Proceedings of the 2000 IEEE Congress on Evolutionary Computation, Carlos Fonseca, JongHwan Kim, and Alice Smith, Eds. 2000, pp. 104 - 111, IEEE Press.

[23] G. R. Raidl, B. A. Julstrom, Edge-Sets: An Effective Evolutionary Coding of Spanning Trees, TECHNISCHE UNIVERSITT WIEN, Institut fr Computergraphik und Algorithmen, Technical Report TR-186-1-01-01, 2002.

[24] I. charon, O. Hudry, The noising method: A new combinatorial optimization method, Operations Research Letters, 14, 1993, pp. 133 - 137.

[25] A. Freville, S. Hanafi, A. El. Abdellaoui, Noising method for the 0-1 knapsack problem, FRANCORO, Mons, Belgique, 1995.

[26] I. charon, O. Hudry, Lamarkian genetic algorithms applied to the aggregation of preference, Annals of Operational Research, 80, 1998, pp. 281 - 297.

[27] I. charon, O. Hudry, Variations on the noising schemes for a clustering problem, in The Proceeding of the Third Metaheuristics International Conference (MIC - 99), Angerados Reis, Brazil, 1999, pp. 147 - 150.

[28] I. charon, O. Hudry, A. Lobstein, Application of noising method to coding theory, IFORS 96, Vancouver, canada, 1996.

[29] M. Gen, R. Cheng, D. Wang, Genetic algorithms for solving shortest path problems, Proc. IEEE Int. Conf. Evol. Comput., 1997, pp. 401 - 406.

[30] M. Gen, L. Lin, R. Cheng, Bicriteria Network Optimization Problem using Priority-based Genetic Algorithm, IEEJ Transactions on Electronics, Information and Systems. 2004.

[31] G. Raidl and B. A. julstorm, A weighted coding in a genetic algorithm for the degree constrained minimum spanning tree problem, proceedig of the ACM Symposium on Applied Computing (SAC '00), Como, Italy, March 2000, Vol. 1, 440-445.

[32] R. Poli, An Analysis of Publications on Particle Swarm Optimisation Applications, Tech. Rep. CSM-469, Department of Computing and Electronic Systems, University of Essex, Colchester, Essex, UK, May-November 2007

[33] C-C. Shen, C. Jaikaeo, Ad hoc Multicast Routing Algorithm with Swarm Intelligence, Mobile Networks and Applications 10, 4759, 2005

[34] P. Janacik, O. Kao, U. Rerrer, A routing approach using swarm-intelligence for resource sharing in wireless ad hoc networks, IEEE Symposium on Trends in Communications, 170- 174, 2004

[35] M. Kirkpatrick, G. Madey, Using Swarm Intelligence to Broadcast Messages in Highly Mobile Ad Hoc Networks, Seventh Annual Swarm Researchers Meeting (Swarm2003), Notre Dame, IN, 2003

[36] G. Di Caro, F. Ducatelle, L.M. Gambardella, Swarm Intelligence for Routing in Mobile Ad Hoc Networks, Proceedings of the IEEE Swarm Intelligence Symposium, Pasadena, USA, June 8-10, 2005