

Predicted Energy-Efficient Bee-inspired Routing (PEEBR) Path Selection Optimization

Imane M. A. Fahmy^{1,*}, Laila Nassef², Hesham A. Hefny¹

¹Computer & Information Sciences department, Cairo University, Giza, 12613, Egypt

²Faculty of Computing & Information Technology, King Abdulaziz University, Jeddah, Kingdom of Saudi Arabia

Abstract In this paper, the on-demand Predictive Energy-Efficient Bee-inspired Routing (PEEBR) was modified in order to solve the Mobile Ad-hoc wireless Networks MANETs routing problem from an energetic point of view. PEEBR's main parameters to be optimized are: The energy consumption during communication between endpoint nodes to be minimized and the battery residual power of the intermediate nodes over the routing path to be maximized. The bee-inspired PEEBR employs two types of bee agents: the scouts and the foragers involved in the honey bees food search process. Based on the Artificial Bee Colony ABC optimization model, PEEBR assigns each potential path between a certain source-destination pair a fitness value and a goodness ratio using its energetic parameters. The experiments developed, using a self-made simulator, have shown the ability of PEEBR to optimize the path selection process between a certain source and destination nodes pair while increasing number of nodes in MANET and the topology variation. PEEBR aims to route packets over the path that not only achieves lower energy consumption but also with higher battery residual power of the route's intermediate nodes in order to maximize the network lifetime.

Keywords PEEBR, Energy Consumption, Battery Residual Power, ABC Optimization, Path Selection, Network Lifetime

1. Introduction

The Mobile Ad-hoc wireless Networks commonly referred to as MANETs are infrastructureless wireless mobile networks that permit mobile nodes hop-by-hop communication from the source to the destination. MANETs are characterized by their self-organization and decentralization nature. Thus, that dynamic nature of MANETs exhibits many challenging routing issues.

Moreover, one of the most important resources in MANETs is the limited battery power of the mobile nodes. The limited battery lifetime poses yet another challenge for the routing algorithms to distribute the packets on multiple paths in such a manner that the battery of different nodes deplete at an equal rate, as a result, the life time of the network could be increased [1].

There are four technically known routing protocols categories for MANETs: proactive, reactive, hybrid or hierarchical. Proactive routing protocols are mainly table-driven based on memorizing and updating routing tables containing all paths stored by all the network's nodes. However, proactive protocols are inefficient for large

MANETs which require a larger memory capacity of the nodes and result in huge exchange of control packets and hence consume a greater amount of energy every time users move and the MANET topology changes. On the other hand, reactive or on-demand protocols attempt to find the routing path between a source node and a destination node when they need to communicate which saves MANET resources in terms of memory and power of the nodes but could result in delay before discovering the requested path. Hybrid protocols benefit from both categories capabilities by dividing the MANET into overlapping regions or clusters where nodes could communicate proactively inside and reactively from cluster to another on-demand.

In order to consider energy consumption while routing in MANETs, a group of routing protocols classified as energy efficient routing protocols has emerged as presented by [2], [3] and [4]. Then, another group of routing protocols classified as power-aware routing protocols for MANETs as discussed by [5] and [6] consider predicting the power amount that would be consumed before choosing the routing path. The previous research work of different energy conservation techniques has been surveyed by [7]. It assumed localized power aware routing algorithms which are devised on the assumption that each network node has accurate information about the location of its neighbors and the destination node.

Swarm Intelligence (SI) is a computational intelligence

* Corresponding author:

iman_fahmy@cu.edu.eg (Imane M. A. Fahmy)

Published online at <http://journal.sapub.org/jwnc>

Copyright © 2014 Scientific & Academic Publishing. All Rights Reserved

approach, as described by [8] that is based on the study of collective behavior of social insects in decentralized, self-organized systems. SI involves a collective behavior of autonomous agents that locally interact with each other in a distributed environment to solve a given problem in the hope of finding a global solution to the problem as defined by [9]. Ant Colony Optimization (ACO) introduced by [10] and Artificial Bee Colony (ABC) Optimization by [11] are among SI optimization techniques that are relatively more robust, reliable, and scalable than other conventional routing algorithms. Since they do not involve extra message exchanges to maintain paths when network topology changes, ACO and ABC optimization techniques are suitable for optimizing MANETs routing protocols where nodes move dynamically and topology changes frequently.

The proposed Predictive Energy-Efficient Bee-inspired Routing algorithm (PEEBR) [1] combines the energy awareness and efficiency with the ABC SI optimization to determine on-demand the optimal path between a certain source-destination pair. PEEBR requires mainly two types of agents for routing optimization: Scouts, who discover on-demand new routes to the destinations and foragers, who transport data packets and simultaneously evaluate the quality of the discovered routes based on the energy amount expected to be consumed along the path and the nodes battery residual power together with the hop count. The foragers sense the state of the network, utilize measured metrics to rate different routes in MANET, and then select the optimal path for data packets routing in order to maximize network lifetime.

The paper is organized as follows: The second section introduces the Bee Colony Optimization (BCO) model. Then, the previous bee-inspired routing protocols are briefly discussed by the third section. The proposed Predictive Energy Efficient Bee Routing (PEEBR) algorithm is described in the fourth section. Then, PEEBR experiments and results are discussed by the fifth section. Finally, the sixth section concludes the paper's research contribution and future research work.

2. Bee Colony Optimization (BCO)

Bee Colony Optimization (BCO) model is a new general purpose SI optimization technique based on efficient labor employment and efficient energy consumption through a multi-agent distributed model. Unlike the Ant Colony Optimization (ACO) model, that has adopted mainly one natural insect behavior which is the "food search" that aims to discover the shortest path between the ant colony and the food source, BCO model has adopted mainly two natural behaviors from the social bees life: The mating process behavior and the foraging process behavior.

The mating behavior was practically used as a powerful SI optimization clustering technique that competes with other classic clustering techniques and also with other SI optimization models especially ACO model. While the

foraging behavior from which inspired this research is based on the natural bee food source search behavior that tries to find the food source with the highest quality.

Inside the bees hive, bees are divided into five groups plus the queen bee (or queen bees). The bee swarms staying in the hive are the "food packers" group and the "nurses" group responsible for feeding the queen and the babies. The three other groups are those involved in the food search process: The "scouts", the "foragers" and the "workers". By distributing the food search process among three troops of bees, the energy consumed by each bee to find a certain food source will be reduced and hence the search trip time will be proportionally minimized through three phases of search: First, discovering all potential sources by the scouts. Then, the foragers assign each discovered source a certain probability according to its quality (nectar amount) to allocate an equivalent number of workers. Finally, collecting nectar by worker bees from the food source according to the qualification probability assigned by the foragers to each food source. Particularly, the two main groups of bees involved:

- The scouts: responsible of discovering all possible food sources (flowers). Then, they when they return to the hive, they guide by performing the "waggle dance" or "round dance" (if the food source is near the hive) to the food source direction. The angle from the hive between the sun and food source is illustrated by figure 1.
- The foragers (onlookers): responsible of qualifying the discovered food sources (in terms of nectar amount and quality) then recruitment and guidance of the swarm of worker bees to their direction from the hive.

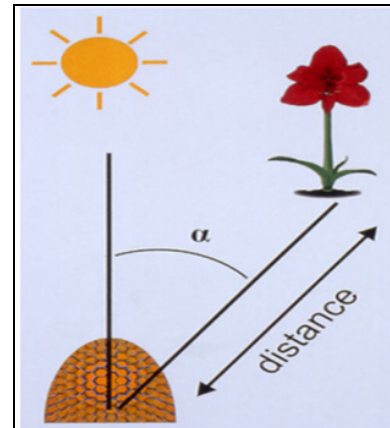


Figure 1. The direction of the food source relative to the sun from the hive and the waggle dance

Karaboga and B. Basturk in [11] and [13] inspired by the "foraging behavior" of honey bees, have proposed a numerical function optimization algorithm based on the Artificial Bee Colony (ABC) then tested its performance to prove efficiency and outperformance when compared to other SI optimization algorithms. Then, Karaboga and Ozturk in [12] have defined clustering as the process of recognizing natural groupings or clusters in multidimensional data based on some similarity measures.

3. Bee-inspired Routing Protocols

There are few recent research works solving the MANET routing problem inspired from the natural bee's behavior as discussed in the following sub-sections.

3.1. BeeHive Routing Protocol

Wedde, Farooq, and Zhang (2004) in [14] introduced a novel routing algorithm called "BeeHive" inspired by the communicative and evaluative methods and procedures of honey bees.

In this algorithm, bee agents travel through network regions called foraging zones. On their way, their information on the network state is delivered for updating the local routing tables. BeeHive was fault tolerant, scalable, and relies completely on local, or regional, information, respectively. They have also demonstrated through extensive simulations that the reactive BeeHive routing protocol achieves a similar or better performance compared to state-of-the-art Mobile Ad-hoc Networks routing algorithms such as: AODV, DSDV and DSR.

In the BeeHive algorithm, the bee's colony architecture consists of three main exploitation floors as described below:

1. The entrance floor: At this floor the scouts come back to the hive (from their exploration phase). This is the interface to lower level (MAC layer).
2. The dance floor: This is the floor where the dance takes place. The foragers update the routing information of hive's bees (node).
3. The packing floor: This floor is where the worker bees come back with honey to be packed (path control information to update tables). It is responsible of interacting with higher level layer (transport layer).

3.2. BeeAdHoc Routing Protocol

H. F. Wedde et al. (2005) in [15], then presented a new routing algorithm for MANET which is also inspired by the honey bee behavior called BeeAdHoc. The algorithm is simple and mainly needs two types of messages for routing: the scouts: They discover on-demand new routes to the destinations. Then, the forgers: which transport data packets and simultaneously evaluate the quality of the discovered routes.

The BeeAdHoc routing as shown by figure 2 [15] considers each node in the network as a hive. Each node periodically sends out bee agents: Scouts to explore the network and collect information about any available food sources regardless of their quality. The exploration process achieved by the scout bees could be described and mapped onto the following steps in MANET: Scouts are broadcasted. A TTL (Time To Live packet) is set for each Scout. Then, Scouts take a backward journey to the source (hive) on the same route. At last, Scouts recruit foragers when they are back to the hive by dancing to guide them to the food direction (angle) from the hive.

BeeAdHoc considers the dance floor as the routing table where the bee agents provide the information about the

quality of the path they have traversed. Then the exploitation process will be achieved by the foragers and the main workers. Foragers receive data packets from the transport layer (provided by the scouts) and after determining the path quality, they deliver it also by dance to the main workers.

Finally, the main workers who receive packets from the transport layer (foragers) are recruited by the foragers such that every worker has a food source.

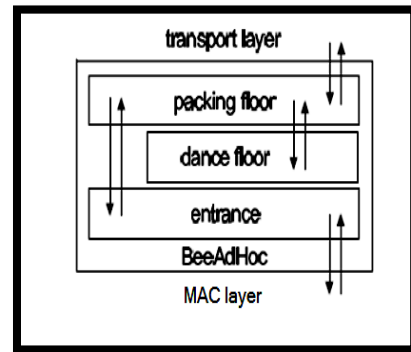


Figure 2. BeeAdHoc algorithm architecture

3.3. Bee-IP Routing Protocol

Giagkos and Wilson in [16] have introduced a new bee-inspired routing protocol Bee-IP for MANETs. They have highlighted the ability of bees to evaluate routing paths by considering several quality factors. BeeIP has used cross-layering by obtaining parameters from the lower physical and MAC layers to feed to the core of the protocol. Then, the artificial bees could be able to make predictions about the link's future performance.

In BeeIP, every time there is a need for a link to be established, the source node will behave as being the hive, the destination node will behave as being the source of food, and all the intermediate nodes will constitute the path that a forager bee needs to traverse while flying from one endpoint to the other.

The BeeIP model uses three types of agents in the form of data packets: The scouts, the Ack scout, and the forager.

- Scout: Sent when a scouting process is initialized in order to discover new paths towards a given destination every time there is a new request from the upper layer and previous routing knowledge is unsatisfactory. A scout is transmitted using broadcast to all neighboring nodes. This technique benefits not only from the propagation of the initial request, but also the introduction of the transmitting node to its neighborhood.

- Ack scout: Once the scout reaches its destination the scouting is considered successful and an Ack scout packet is created. Ack scouts use a source routing fashion to travel back to the source, using unicast transmission. Therefore, the route that was followed towards the destination is used in reverse. On their way back, Ack scouts acknowledge the success of the scouting to both the intermediate nodes and the source node.

- Forager: When BeeIP is unable to transmit a data packet, it stores it into a local queue and starts a new scouting

process for its destination. This decreases the packet loss due to incomplete routing information. Once an Ack scout returns back and acknowledges the existence of a path, all packets for the corresponding destination in the queue are being transmitted. Foragers are specially crafted packets that have three important roles: Firstly, they carry (in form of payload) the data packets from the source to the destination. Secondly, they are used to update neighboring nodes' states and links' information like scouts. Thirdly, foragers are constantly monitoring the path they traverse for any improvements. They collect the differences between the local reliability levels, calculated by using the current forager, and the local reliability levels calculated by the previous forager's visit, and report the summation back to the hive. The summation represents the total reliability level of the path, hence, the global reliability level.

4. PEEBR Optimization Algorithm

In bees search process, there are three main phases: First, the scouts seek out all potential food sources that are equivalent to the potential MANET routing paths. Then, the foragers assign each discovered food source (routing path) a certain probability according to its quality (nectar amount) interpreted as the link cost for MANET. Finally, the worker bees collect the nectar from the food source with the highest quality according to the qualification probability assigned by the foragers which is equivalent to choosing the optimal path according to its quality to communicate the data stream of packets on it in MANET.

The proposed Predictive Energy Efficient Bee-inspired Routing protocol PEEBR by [1] was inspired from the bees foraging process. Particularly, the two main groups of bees involved: The scouts and the foragers. PEEBR inspired by the BCO SI model is an algorithm for routing path selection optimization based on energy prediction, consumption efficiency and nodes battery residual power maximization for the MANET in order to increase the network lifetime. In PEEBR, the optimal path discovery process from source n_s to the destination node n_d could be described as follows:

• The Scout bee

1. Source node n_s , in order to route efficiently its packets to a destination node n_d , floods a "Scout packet" associated with a TTL (Time-To-Live) to all j neighboring nodes.
2. For each "Scout cycle", each "Scout" flies over one of the j potential routes R_j until it reaches destination node n_d .
3. If the TTL packet expires, the "Scout" bee agent packet will die indicating failure to reach destination to the source and the corresponding routing path will be avoided.
4. When a bee agent reaches the destination node n_d , it is sent back to its source n_s through the same traveled route.
5. The backward packet from destination node n_d to source node n_s , "Scout packet", collects the potential route's routing information.
6. It counts number of hops $h(R_j)$.

7. It collects each route nodes residual battery power $B_{(n_{ij})}$ where $i=1$ to N_j nodes and $j=1$ to M paths.

8. It also memorizes the amount of receiving power consumed.

• The Forager bee

1. At the source node n_s , the "forager" evaluation process starts by calculating the predicted amount of energy to be consumed for each "Backward Scout" discovered route.

2. Each potential route cost $f(R_j)$ is calculated for each route R_j dependent on its hop count $h(R_j)$, its nodes residual battery power $B_{(n_{ij})}$ and its expected amount of receiving power consumed using expression (6).

3. The "Forager" associates a fitness value $fit(R_j)$ and a goodness ratio of each route $G(R_j)$ as deduced from expressions (7) and (8).

4. At the end of a foraging iteration, each potential path nodes battery residual power $B_{(n_{ij})}$ should be decayed exponentially as in (9) to reflect the real world's energy consumption.

5. The optimal route R_o between n_s and n_d is the route with the maximum goodness ratio as given by expression (10).

6. The other potential routing paths are memorized by source node n_s (for a time interval in communication) in order to be used if any failure occurred during transmission on the optimal route R_o but with respect to their goodness ratio.

7. A new "Scout cycle" is launched until the maximum number of iterations is reached or a minimal fitness value.

In this paper, an efficient, multi-objectives optimization routing technique for MANETs is proposed inspired from the bees food search process that aims to discover all potential paths and select the optimal path among them between a source node and a destination node based on two main parameters: The amount of energy consumed by each node along the path and the nodes battery residual power together with hop count. These parameters reflect the potential path goodness assigned by forager bee agents. The generic expression used to calculate $E(p)$ the energy required to transmit a packet p from (1) to (5) are as in [17]. $E(p)$ in joules (or milli-joules) is given by Eq.(1):

$$E(p) = i * v * t_p \quad (1)$$

Where i represents the current consumption, v is the voltage used and t_p is the required time in seconds to transmit a packet given by Eq.(2):

$$t_p = \left(\frac{p_h}{6 * 10^6} + \frac{p_d}{54 * 10^6} \right) \quad (2)$$

Where p_h is the packet header size and p_d is the packet data size (both in bits). Then, the energy consumed by the node in transmit mode $E_t(p)$ is given by Eq. (3), while the energy consumed in reception mode $E_r(p)$ or in overhearing mode when the node overhears the packets exchanged within its range are given by Eq.(4):

$$E_t(p) = 280mA * v * t_p \quad (3)$$

$$E_r(p) = E_o(p) = 240mA * v * t_p \quad (4)$$

Therefore, the total amount of energy consumed at a node n_i is calculated by Eq. (5):

$$E(n_i) = E_t(p_{n_i}) + E_r(p_{n_i}) + E_o(p_{n_i}) \quad (5)$$

On the other hand, all nodes residual power $B(n_{ij})$ was initiated using a random value generation in a range from 1000 to 3000 joules. PEEBR's cost function combining the hop count $h(R_j)$ between a given source and destination nodes pair and the average predicted energy consumption $E(n_{ij})$ as path minimizing parameters while the average path nodes battery residual power $B(n_{ij})$ as maximizing parameter are given by expression (6).

$$f(R_j) = h(R_j) \frac{\sum_{i=1}^{N_j} E(n_{ij})}{B(n_{ij})} \quad (6)$$

Where N_j is the number of nodes on a potential path R_j among M potential paths between the source and destination and the path index $j=1, \dots, M$. Then, the path fitness fit (R_j) could be computed using expression (7).

$$fit(R_j) = \frac{1}{1+f(R_j)} \quad (7)$$

Therefore, the path goodness $G(R_j)$ could be computed using expression (8)

$$G(R_j) = \frac{fit(R_j)}{\sum_{j=1}^M fit(R_j)} \quad (8)$$

In order to test PEEBR's performance, it was run on $T_{max}=100$ iterations. The nodes battery residual power $B(n_{ij})$ was decayed to reflect the real world's as given by expression (9):

$$B(n_{ij}) = B(n_{ij})^0 * e^{-\frac{t}{\tau}} \quad (9)$$

Where $B(n_{ij})^0$ the initial node battery residual power, t is the iteration number and τ is a time constant. Finally, PEEBR termination conditions were: reaching the maximum number of iterations T_{max} or a minimal predefined fitness value. The resulting optimal path R_o that is the path with the highest goodness ratio that is:

$$R_o = \arg \max_j \{G(R_j)\} \quad (10)$$

5. PEEBR Experiments and Results

The proposed routing algorithm PEEBR in [1] was modified according to expressions (6) to (10) as discussed in the fourth section in order to consider energy consumption and battery residual power parameters. Then, the modified PEEBR was simulated using a self-made simulator to test its performance while optimal path selection based of the energetic parameters mainly for different MANET sizes between a certain source and destination pair as described below.

Table 1 shows the mapping of Bee Colony Optimization BCO parameters on the proposed routing algorithm PEEBR for the three experiments. These parameters include the colony size that is MANET size (number of nodes), the number of food sources mapped into the number of solutions (potential paths) and the number of trials interpreted by the number of iterations before program termination by discovering the optimal path among all potential paths. The same source-destination pair of nodes is used for different MANET sizes to ensure PEEBR's results stability.

Table 1. Mapping BCO controlling parameters on PEEBR experiments

Parameters	7-Nodes MANET	15-Nodes MANET	25-Nodes MANET
Colony size	7	15	25
Number of food sources	5	7	7
Number of trials	50	50	50

The nodes battery residual power was randomly generated for the first iteration within a range from 1000 to 3000 joules then decreased according to expression (9) for the rest of iterations.

5.1. Experiment 1: 7-Nodes MANET

According to the MANET graph in figure 3, there are six potential paths between the source node 3 and the destination node 5 as shown and discussed by table 2.

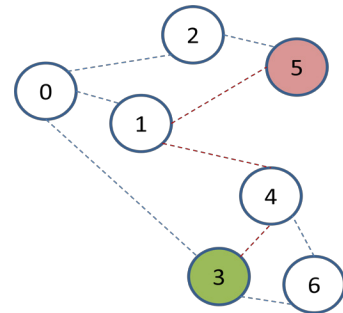


Figure 3. 7-Nodes MANET PEEBR path selection between node 3 and node 5

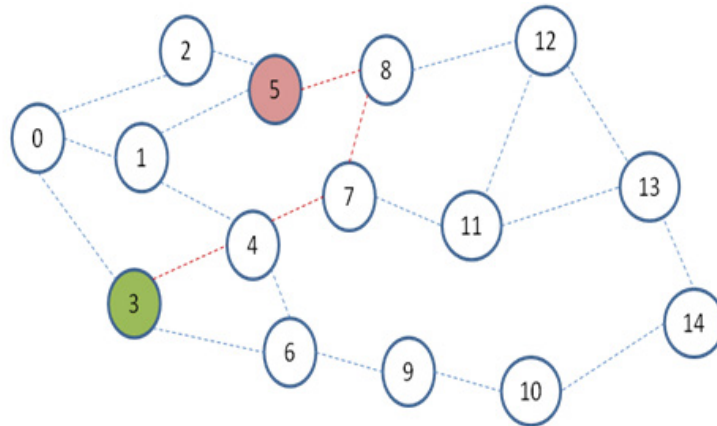
In table 2, the six potential routing paths were detected by PEEBR scouts from source node 3 to destination node 5 in a 7-nodes MANET. Then, PEEBR foragers have assigned each potential path a certain goodness ratio $G(R_j)$ based on number of hops, energy consumption amount and battery residual power that reflect the path's fitness. Although there are three paths with 3-hops, only the path $R_1:(3,0,2,5)$ was selected by PEEBR as the optimal fittest path between node 3 and node 5 since it achieves the least energy consumption and the maximum nodes battery residual power for 50 iterations.

Table 2. PEEBR Path Selection Optimization for 7-Nodes MANET between node 3 and node 5

Number of Iteration	Potential Paths $R_j = 1, \dots, M$	Number of hops $h(R_j)$	Path j nodes Residual Battery power $\sum_{i=1}^{N_j} B_{ij}(R_j)$	Path fitness $fit(R_j)$	Path Goodness $G(R_j)$	Optimal path $O(R_j)$
50	$R_1: (3,0,2,5)$	3	9197.61 J	9.91562e-006	0.99999	$R_1: (3,0,2,5)$
	$R_2: (3,4,1,0,2,5)$	5	13710.9 J	1.10861e-005	0.999989	
	$R_3: (3,6,4,1,5)$	4	12210.8 J	1.19971e-005	0.99999	
	$R_4: (3,4,1,5)$	3	8692.4 J	1.04919e-005	0.99999	
	$R_5: (3,0,1,5)$	3	8645.34 J	1.0549e-005	0.999989	
	$R_6: (3,6,4,1,0,2,5)$	6	17229.3 J	1.05866e-005	0.999989	

5.2. Experiment 2: 15-Nodes MANET

According to the MANET graph in figure 4, there are seven potential paths between the source node 3 and the destination node 5 as shown and discussed by table 3.

**Figure 4.** 15-Nodes MANET PEEBR path selection between node 3 and node 5

In table 3, as the number of MANET nodes increased to 15-nodes, seven possible routing paths were detected by PEEBR scouts from source node 3 to destination node 5. Then, PEEBR foragers have assigned each potential path a certain goodness ratio $G(R_j)$ based on number of hops, energy consumption amount and battery residual power that reflect the path's fitness. Although the 4-hops path $R_7: (3,4,7,8,5)$ is not the least hop count path, it was chosen by PEEBR as the optimal fittest path between node 3 and node 5 since it achieves the least energy consumption and the maximum nodes battery residual power for 50 iterations.

Table 3. PEEBR Path Selection Optimization for 15-Nodes MANET between node 3 and node 5

Number of Iteration	Potential Paths $R_j = 1, \dots, M$	Number of hops $h(R_j)$	Path j nodes Residual Battery power $\sum_{i=1}^{N_j} B_{ij}(R_j)$	Path fitness $fit(R_j)$	Path Goodness $G(R_j)$	Optimal path $O(R_j)$
50	$R_1: (3,0,2,5)$	3	10589.8 J	8.61206e-006	0.999991	$R_7: (3,4,7,8,5)$
	$R_2: (3,4,1,0,2,5)$	5	15526.7 J	9.78958e-006	0.99999	
	$R_3: (3,6,4,1,5)$	4	11998 J	1.01351e-005	0.99999	
	$R_4: (3,4,1,5)$	3	10218.7 J	8.92479e-006	0.999991	
	$R_5: (3,0,1,5)$	3	10376.4 J	8.78919e-006	0.999991	
	$R_6: (3,6,4,7,8,5)$	5	17319.3 J	8.77634e-006	0.999991	
	$R_7: (3,4,7,8,5)$	4	15540.1 J	7.82494e-006	0.999992	

5.3. Experiment 3: 25-Nodes MANET

According to the MANET graph in figure 5, there are seven potential paths between the source node 3 and the destination node 5 as shown and discussed by table 4.

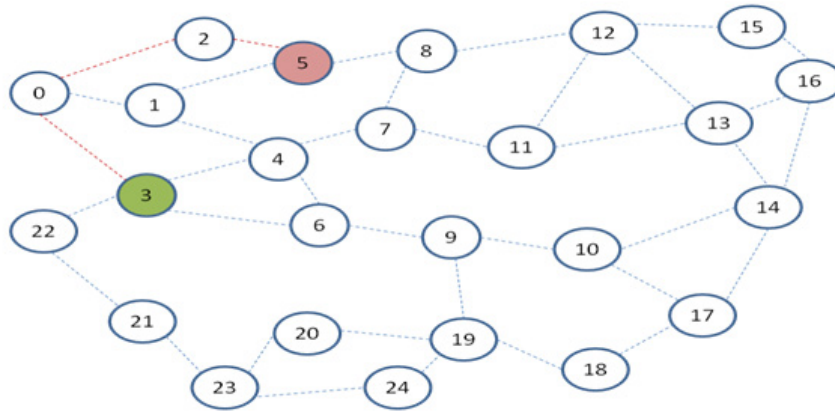


Figure 5. 25-Nodes MANET PEEBR path selection between node 3 and node 5

In table 4, for a 25-nodes MANET, seven possible routing paths were detected by PEEBR scouts from source node 3 to destination node 5. Although there are three 3-hops paths, only the path $R_1:(3,0,2,5)$ was selected by as the optimal fittest path between node 3 and node 5 since it achieves the least energy consumption and the maximum nodes battery residual power for 50 iterations.

Table 4. PEEBR Path Selection Optimization for 25-Nodes MAET between node 3 and node 5

Number of Iteration	Potential Paths $R_j = 1, \dots, M$	Number of hops $h(R_j)$	Path j nodes Residual Battery power $\sum_{i=1}^{N_j} B_{ij}(R_j)$	Path fitness $fit(R_j)$	Path Goodness $G(R_j)$	Optimal path $O(R_j)$
50	$R_1:(3,0,2,5)$	3	10880.4 J	8.38205e-006	0.999992	$R_1:(3,0,2,5)$
	$R_2:(3,4,1,0,2,5)$	5	14968.9 J	1.01544e-005	0.999999	
	$R_3:(3,6,4,1,5)$	4	11088.1 J	1.09667e-005	0.999989	
	$R_4:(3,4,1,5)$	3	8359.42 J	1.09099e-005	0.999989	
	$R_5:(3,0,1,5)$	3	9997.07 J	9.12267e-006	0.999991	
	$R_6:(3,6,4,7,8,5)$	5	11468.4 J	1.32538e-005	0.999987	
	$R_7:(3,4,7,8,5)$	4	8739.77 J	1.39134e-005	0.999986	

Figure 6 shows the impact of MANET number of nodes on the potential paths fitness values from the above experiments for PEEBR’s optimal path selection between the same source-destination pair: source node 3 and destination node 5.

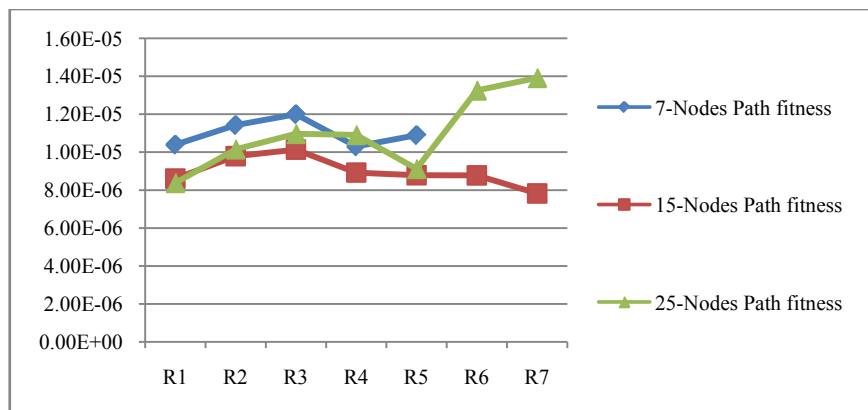


Figure 6. Impact of MANET number of nodes on paths fitness for the (3,5) source-destination pair

Figure 7 shows the impact of MANET number of nodes on the potential paths goodness from the above experiments for PEEBR's optimal path selection between the same source-destination pair: source node 3 and destination node 5.

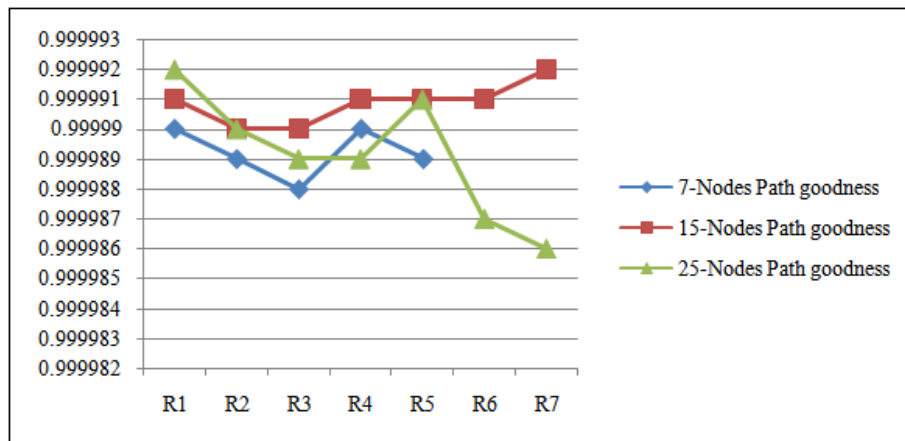


Figure 7. Impact of MANET number of nodes on paths goodness for the (3,5) source-destination pair

In [18], PEEBR's performance was evaluated in terms of energy consumption efficiency through path selection optimization compared to other MANET state-of-art routing protocols including: the Ad-hoc On-demand Distance Vector AODV, the Destination Sequenced Distance Vector DSDV and the Zone-based Routing Protocol ZRP and to another bee-inspired routing algorithm: the BeeAdHoc.

6. Conclusions

The Predictive Energy Efficient Bee-inspired Routing algorithm PEEBR for Mobile Ad-hoc Networks MANETs was modified and then simulated using a self-made event driven simulator in order to test its performance by three different experiments for three MANET sizes. PEEBR algorithm was modified in order to focus on energy reservation as it is considered a critical scarce resource for rechargeable batteries mobile nodes. PEEBR attempts to find the optimal path among a number of potential paths between a certain source-destination pair based on two main energy parameters: the predicted energy consumption and the path nodes battery residual power together with the hop count. Therefore, this energetic perspective in MANET routing problem solution aims to minimize the amount of energy consumption during both routing and communication and to maximize the MANET lifetime by selecting the path with the maximum nodes battery residual power average. The experiments results have shown PEEBR scalability and performance stability for the different MANET sizes.

ACKNOWLEDGEMENTS

I would like to acknowledge Dr. Amr Saad for his continuous help, valuable advices and technical support.

REFERENCES

- [1] Imane M. A. Fahmy, Laila Nassef and Hesham A. Hefny, "PEEBR: Predictive Energy Efficient Bee Routing Algorithm for Ad-hoc Wireless Mobile Networks", IEEE INFOmatics and Systems (INFOS2012), 2012
- [2] L. M. Feeney. "An energy consumption model for performance analysis of routing protocols for mobile ad hoc networks". *Mobile Networks and Applications*, 6(3):239–249, 2001.
- [3] L.M. Feeney and M. Nilsson. "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment". In *Proceedings of IEEE INFOCOM*, 2001.
- [4] N. Nie and C. Comaniciu, "Energy efficient aodv routing in cdma ad hoc networks using beamforming" *EURASIP J. Wireless Communication Networks.*, vol. 2006, no. 2, pp. 14–14, 2006.
- [5] Cui Y., Xue Y., Nahrstedt K., "A Utility-Based Distributed Maximum Lifetime Routing Algorithm for Wireless Networks". *Vehicular Technology, IEEE Transactions on vehicular technology*, 55(3) (2006) 797, 2006.
- [6] R. Shah and J. Rabaey, "Energy aware routing for low energy ad hoc sensor networks", *Wireless Communications and Networking Conference, WCNC2002. IEEE*, vol. 1, pp. 350–355 vol.1, 2002.
- [7] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J. -C. Chen. "A survey of energy efficient network protocols for wireless networks." *Wireless Networks*, 7(4):343– 358, 2001.
- [8] Mayur Tokekar and Radhika D. Joshi, "Enhancement of Optimized Linked state routing protocol for energy conservation", *CS & IT-CSCP*, 2011
- [9] J. Wang, E. Osagie, P. Thulasiraman, R. K. Thulasiraman, "HOPNET: A Hybrid ant colony OPTimization routing algorithm for Mobile ad hoc NETwork", *Elsevier Ad Hoc Networks*, June 2008.

- [10] M. Dorigo, M. Birattari and Thomas Stutzle. "Ant Colony Optimization: Artificial Ants as computational intelligence technique». Université libre de Bruxelles, Belgique IEEE Computational Intelligence magazine, November 2006.
- [11] D. Karaboga and B. Basturk. "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm". Springer Science+Business Media B.V., 2007.
- [12] D. Karaboga and Ozturk, "A novel clustering approach: Artificial Bee Colony (ABC) algorithm", Elsevier, Applied Soft Computing 11 (2011) 652–657, 2011.
- [13] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm", Elsevier, Applied Soft Computing 8 (2008) 687–697, 2008
- [14] H. F. Wedde, M. Farooq, and Y. Zhang. "Beehive: An efficient fault-tolerant routing algorithm inspired by honey bee behavior". In Proceedings of ANTS Workshop, LNCS 3172, pp. 83–94. Springer Verlag, 2004.
- [15] H. F. Wedde, M. Farooq, T. Pannenbaecker, B. Vogel, C. Mueller, J. Meth, and R. Jeruschkat. "BeeAdHoc: an energy efficient routing algorithm for mobile ad-hoc networks inspired by bee behavior." In Proceedings of ACM GECCO, pages 153–160, 2005.
- [16] Alexandros Giagkos and Myra S. Wilson, "BeeIP: Bee-Inspired Protocol for Routing in Mobile Ad-Hoc Networks", Dept. of Computer Science, Aberystwyth University, Penglais, SY23 3BD, Wales, UK, 2010.
- [17] Marco Fotino, Antonio Gozzi, Floriano De Rango, Salvatore Marano, Juan-Carlos Cano, Carlos Calafate, Pietro Manzoni, «Evaluating Energy-aware Behavior of Proactive and Reactive Routing Protocols for Mobile Ad Hoc Networks", SPECTS 2007, San Diego, California (USA) July 16-18, 2007.
- [18] Imane M. A. Fahmy, Laila Nassef, Hesham A. Hefny, "On the Performance of the Predicted Energy Efficient Bee-Inspired Routing (PEEBR)", IJACSA International Journal of Advanced Computer Science and Applications, Vol. 5, No. 4, 2014.