Wireless Pers Commun
DOI 10.1007/s11277-016-3692-z

IWDRP: An Intelligent Water Drops Inspired Routing Protocol for Mobile Ad Hoc Networks

Lamri Sayad[1,2] · **Louiza Bouallouche-Medjkoune**[1] · **Djamil Aissani**[1]

Abstract This paper deals with the problem of routing in Mobile Ad hoc Networks (MANET). A mobile ad hoc network is a collection of mobile devices deployed without any pre-established infrastructure or centralized administration. Routing is a very challenging issue since the appearance of this technology. The main goal of every routing protocol is to find a route between two communicating nodes while optimizing overall performances of the network. This paper introduces a novel routing protocol inspired from the nature and that should deal with the dynamic aspect of MANET. The used approach, called Intelligent Water Drops (IWD), mimics the processes that happen in the natural river systems, particularly, the actions that water drops perform in the rivers to find the shortest path to their destination (sea). In fact, it is observed that water drops of a river often find good paths among lots of possible paths in their ways from a source to a destination. We combined these ideas with a route failure prediction mechanism to develop a new routing protocol for MANETs called IWDRP. This prediction method is based on the received signal strength indicator. Further simulation results show that IWDRP is able to achieve better results in terms of packet delivery, end-to-end delay in comparison with AODV-BFABL. The achievement in this paper has certain reference value to the further study of the routing issue in MANETs.

Keywords Mobile Ad hoc Network · Routing · Intelligent Water Drops · Metaheuristic · Nature inspired algorithm

✉ Lamri Sayad
siad_lamri@yahoo.fr

[1] LAMOS Laboratory, Faculty of Exact Sciences, University of Bejaia, 06000 Bejaia, Algeria

[2] Computer Science Department, Msila University, 28000 Msila, Algeria

Published online: 12 September 2016

1 Introduction

The past several years have witnessed a growing interest to Mobile Ad hoc Networks (MANET). Therefore, an intensive studying of its challenges and issues has been performed. As a result, many novel architectures and protocols have been developed. A Mobile Ad hoc Network is a collection of independent mobile nodes communicating among themselves over wireless channels, thus forming a dynamic arbitrary graph. These networks are fully distributed, and can deployed everywhere without the help of any infrastructure. These characteristics make MANETs suitable for mission-critical applications, such as disaster recovery, battlefields, search and rescue. However, many issues and design challenges still remain open.

Routing is one of the fundamental issues with MANET, so that the network performances depend on the routing protocol efficiency. Routing in MANET is considered as the mechanism of switching data packets from a source to its destination along a certain path within the network. Many works have been conducted and proposed with the purpose of designing new techniques and protocols. This includes DSDV [25], AODV [26], DSR [16], OLSR [2], TORA [23], and ZRP [14]. In general, MANET routing protocols can be classified as being proactive (table-driven) [2, 25], reactive (on-demand) [16, 26] or hybrid [14]. Proactive routing algorithms compute all routes before they are needed. Reactive algorithms construct routes when needed. However, hybrid algorithms use a combination of proactive and reactive approaches.

Recently, a new class of routing protocols has emerged. These protocols uses techniques inspired from the nature, commonly known as Metaheuristics. Metaheuristics are computational models based on the collective behavior seen in natural, biological, physical or chemical systems such as ants, bees, rivers, termites, etc. Routing based on metaheuristics provides a promising alternative to the traditional routing approaches. Since it uses artificial agents which have the ability to adapt, cooperate and move intelligently from one node to another. Many of the Metaheuristic algorithms have been previously applied to routing in MANETs like Ant Colony Optimization [7, 5, 6], Artificial Bee Colony Optimization [18], Genetic Algorithm [38], and Particle Swarm Optimization [19]. It is in this context that this paper focuses on Intelligent Water Drops Algorithm (IWD) [33]. The purpose is to show that IWD could be a good alternative for packet switching in MANETs. A detailed description of IWD technique is given in Sect. 3.

Our goal, when designing IWDRP, is to show that IWD metaheuristic could be a good alternative to solve routing issue in MANETs. Therefore, the main contributions of this paper are to:

- Propose a new routing protocol for MANETs using IWD (Intelligent Water Drops) metaheuristic.
- Design a new packet forwarding concept based on the IWD metaheuristic ideas.
- Furthermore, this technique is coupled with a Route Failure Prediction (RFP) mechanism using Cross layer approach. The role of RFP is to estimate the route failure time based on the estimated link failure time.

The remainder of this work is organized as follows. Section 2 presents the most relevant work in the area. Section 3 briefly describes the IWD metaheuristic and then a description of the proposed routing protocol is detailed in Sect. 4. Simulation results are discussed in Sect. 5. Finally, Sect. 6 concludes the paper pointing out some future works.

2 Related Work

Several protocols have been proposed to solve the routing issue in MANETs. The earliest ones are AODV, DSDV and DSR. In this section, we focus on existing swarm intelligence based routing protocols proposed for MANETs. For brevity, we discuss only a few protocols, but an interested reader is referred to relevant survey papers [10, 11, 21, 27, 34, 36].

AntNet [3] was one of the first bio-inspired routing protocols. It is based on Ant Colony Optimization (ACO) metaheuristic proposed by Gianni Di Caro and Marco Dorigo. Therefore, special control packets (forward ants and backward ants) are used to mimic the behavior of ants. At regular fixed intervals, a forward ant is sent to collect network information on its way to a random destination node. When the forward ant reaches the destination node, it dies and a backward ant will be created and sent back to the source node. While traveling, the backward ant updates the local traffic model and the pheromone matrix.

Gunes et al. [12] have proposed another Ant Colony based Routing protocol called ARA. It imports some basic aspects of AntNet into AODV. The proposed protocol establishes multiple routes in a reactive way. As in AntNet, ARA uses two kinds of agents (packets): Forward Ant (FANT) and Backward Ant (BANT). The role of using FANT is to gather information about visited nodes and establish pheromone path to the source node. However, BANT agents are used to initialize pheromone track to the destination node. ARA is, generally, characterized by a low overhead because of the use of data packets to maintain pheromone concentration in the established routes.

Di Caro et al. [4] have introduced in 2005 a hybrid multipath ant colony based routing protocol for MANETs called AntHocNet. It consists of a reactive route setup process using reactive ants, and a proactive route maintenance mechanism using proactive ants. Unlike AntNet, AntHocNet establishes routes when they are needed before starting a data session. Every time a source node S needs to communicate with a node D and no route is available, it broadcasts a Reactive Forward Ant (RFANT). Each intermediate node i receiving RFANT forwards it to the next hop. RFANT is unicasted or broadcasted based on the availability of a route to D. On the other hand, when a session is established and at regular periods of time, a Proactive Forward Ant is launched in order to maintain the discovered routes.

Genetic Algorithms (GA) have, also, been used to solve the routing issue in MANETs. Therefore, authors in [40] have proposed to use GA with immigrants and memory schemes for dynamic shortest path routing in MANETs. Each route is considered as a potential solution and is encoded using a string of positive integers that represent the IDs of nodes. The first step consists of initializing the population of solutions. It starts looking a random path from node s to a destination r by randomly selecting a node $v1$ from $N(s)$, the neighborhood of s. Then, it randomly selects a node $v2$ from $N(v1)$. This procedure is repeated until the destination node is reached. Solutions are evaluated using a fitness function. Genetic operators are, then, applied to improve paths qualities.

Another promising metaheuristic approach, that has been successfully applied to many optimization problems, is Artificial Bee Colony Optimization (ABC). Wedde et al. [39] have introduced BeeAdHoc, an energy efficient routing algorithm for MANETs inspired from the foraging principles of honey bees. BeeAdHoc is a reactive source routing algorithm. It uses four kinds of agents, scouts, foragers, packers and swarms. The goal of packers is to find a forager for their data packet. However, scouts are launched from a source node s to a destination node d in order to discover new routes. Foragers are responsible of transporting data packets to their destinations. Simulation results obtained in [39] show that nodes in BeeAdHoc consume significantly less energy as compared to DSR, AODV, and DSDV.

Table 1 gives a summary of some metaheuristic-based routing protocols for MANET.

3 IWD Metaheuristic

3.1 Natural Water Drops

It is observed that a natural river often finds good paths among lots of possible paths in its way to the destination (lake or sea). In the ideal form, water drops of a river would be attracted by gravitational force of the earth to construct a straight path to reach its destination. However, water drops are faced to different kinds of obstacles in their way to the destination preventing the river to form a straight line, and make its real path full of twists and turns. These near optimal or optimal paths are obtained by the actions and reactions that occur among water drops and the water drops with the riverbeds. But the most important propriety about rivers is that it builds an optimal path in terms of both the distance from origin to lake/sea and the constraints (twists and turns) of the environment [32]. At every moment, each water drop flows with a velocity and gathers an amount of soil. Based on these proprieties and the amount of soil in the riverbed, three important characteristics are defined:

- Each water drop gathers an amount of soil proportional to its speed. Thus, the higher the speed of drops, the larger the amount of soil it carries (Fig 1).
- Velocity of water drops increases more on paths with low soil amount than on those with high soil amount (Fig 2).
- When a water drop has to select a path, it chooses the path containing the lowest amount of soil.

3.2 Intelligent Water Drops Algorithm

Based on these observations, Hamed Shah-Hosseini has introduced in 2007 the Intelligent Water Drops algorithm (IWD). Thus, artificial water drops have been used to mimic the

Table 1 Some Metaheuristic-based routing protocols for MANET

Protocol	Year	Class	Metaheuristic
AntNet [2]	1998	Proactive	ACO
ARA [12]	2002	Reactive	ACO
AntHocNet [4]	2004	Hybrid	ACO
BeeAdHoc [39]	2004	Reactive	ABC
GA [40]	2010	–	GA
TSRP [29]	2012	Reactive	ACO + Tabu search
ANTALG [13]	2014	Hybrid	ACO
SMART [1]	2014	Hybrid	ACO + RFD
LB-Termite [20]	2014	Hybrid	Termite
E-Ant-DSR [35]	2015	Reactive	ACO
OTLR [28]	2016	Reactive	Tabu search
Proposed	–	Hybrid	IWD

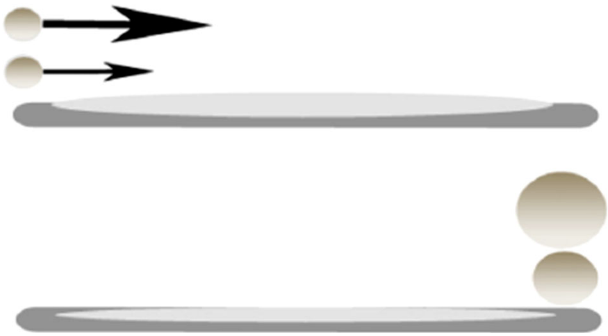


Fig. 1 The relationship between IWD velocity and the amount of soil carried. Figure adapted from [8]

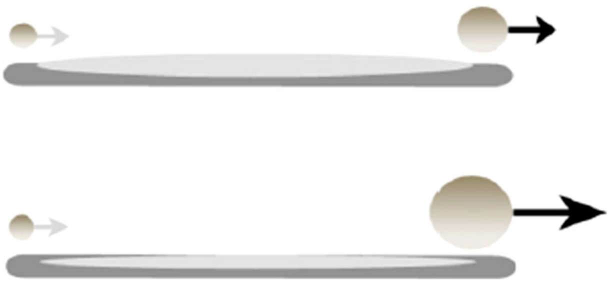


Fig. 2 IWD velocity increases when flowing on paths with less soil amount. Figure adapted from [8]

actions and reactions that occur between riverbed and the water drops that flow within. In fact, each water drop is defined with the following two proprieties:

- *Soil(IWD)* the amount of soil carried by the IWD,
- *Vel(IWD)* the velocity of the IWD.

The values of these proprieties are adjusted as IWD flows in its environment. Therefore, the amount of soil carried by a water drop IWD when moving from position i to next position j, $\Delta Soil(i,j)$, is calculated using Eq. (1).

$$\Delta Soil(i,j) = \frac{a_s}{b_s + c_s \cdot time(i,j;Vel(IWD))} \tag{1}$$

where: a_s,b_s and c_s are parameters of the algorithm.

And *time*(i, j; *VSel*(*IWD*)) is the time needed to traverse from i to j when moving with a velocity of *Vel*(*IWD*). Thus, the value of *time*(i, j; *Vel*(*IWD*)) is proportional to *Vel*(*IWD*) and inversely proportional to the distance between i and j.

This amount is removed from the path (Eq. 2) and, then, added to the total amount of soil carried by *IWD* (Eq. 3).

$$Soil(i,j) = (1-\rho) \cdot Soil(i,j) - \rho \cdot \Delta Soil(i,j) \tag{2}$$

where ρ is a parameter that should be a non-negative number less than one. The soil amount of the water drop is updated with the simple equation:

$$Soil(IWD) = Soil(IWD) + \Delta Soil(i,j) \tag{3}$$

In the other hand, IWD's velocity is adjusted every time the water drop moves from a position to another. As seen in Eq. (4), it is inversely proportional to the amount of soil between the two positions.

$$Vel(IWD, t+1) = Vel(IWD, t) + \frac{a_v}{b_v + c_v \cdot Soil(i,j)} \tag{4}$$

where a_v, b_v and c_v are positive values, thus the velocity of the water drop will increase over time.

IWD is a probabilistic algorithm, consequently an artificial water drop chooses node j as its next position according to a probability $P(i, j)$:

$$P(i,j) = \frac{f(Soil(i,j))}{\sum_{k \in Neighbors(i)} f(Soil(i,k))} \tag{5}$$

where i is the current node, $Neighbors(i)$ represents neighbors list of node i, and

$$f(soil(i,j)) = \frac{1}{\varepsilon + g(Soil(i,j))} \tag{6}$$

where

$$g(Soil(i,j)) = \begin{cases} Soil(i,j) & if \quad Min(Soil(i,\cdot)) > 0, \\ Soil(i,j) - Min(Soil(i,\cdot)) & Otherwise \end{cases} \tag{7}$$

where $Min(Soil(i,\cdot))$ is the minimum amount of soil of all neighbors of node i.

This approach has been demonstrated to be effective in solving several optimization problems like : the traveling salesman problem (TSP) in [32], the multiple knapsack problem [30], Robot Path Planning [9, 37], Vehicle Routing Problem [17], optimal data aggregation tree in wireless sensor networks [15], and multi-objective job shop scheduling [22].

4 IWDRP: The Proposed IWD Routing Protocol

4.1 Motivation

To achieve packet routing in MANETs, we propose a new bio-inspired approach called IWDRP. This protocol is inspired from action and reaction that happens between water drops and the riverbed. IWD metaheuristic could be a good alternative to solve routing issue in MANETs because of the followings:

- It can provide multipath routing,
- It is totally distributed,
- It is highly adaptive to network changes,
- It is robust to agent failures.

Due to all these reasons and the demonstrated efficiency of IWD algorithm, we think that this technique can fit well with the dynamic aspect of MANET when designing a routing protocol.

4.2 General Description

The goal, when designing IWDRP, was to adapt IWD metaheuristic in order to solve routing issue in MANETs. The similarities observed, between IWD system and MANET, were the starting point of our work. Table 2 summarizes the analogy between IWD system and MANETs.

In IWDRP, each node of the network consists of the following data structures:

- *Routing table* it stores a list of routes to all reachable destinations. Each entry of this table consists of the following fields: destination id, Next hop, Soil amount, delay, next route failure time.
- *Neighbor list* list of all neighbors of a node. Each entry of this array consists of the following fields: Neighbor ID, last received signal power, last received signal time, next link failure time.

Six types of control packets are used to support IWDRP concepts:

- Reactive Forward IWD (RFIWD),
- Reactive Backward IWD (RBIWD),
- Proactive Forward IWD (PFIWD),
- Proactive Backward IWD (PBIWD),
- Route Error (RTERR),
- HELLO.

4.3 Algorithm

When a node *s* needs to send a data packet to a destination *d* and no route is available, it selects the link with less soil amount. Algorithm 1 shows the pseudo code of selecting next hop.

Algorithm 1: Select next hop

```
If (No route is available to d)
Begin
   Reactive_route_discovery(d);
End

Min = MaxValue
Next_hop = -1
For each neighbor n of node S with a route to d do
Begin
   If(n.soil_amount < Min and n.next_route_failure_time  > current_time)
   Begin
         Min = n.Soil_amount
         Next_hop = n
   endIf
end
return Next_hop
```

4.3.1 Reactive Route Discovery Phase

However, if no route is available to destination *d*, a reactive route discovery process is launched. Figure 3 shows the algorithm executed by nodes during the reactive phase.

Table 2 Correspondence between an IWD System and Network Elements

IWD System	Network elements
Environment	Network
River	A potential route
Lake or sea	Destination node
Twists and turns of a river	Intermediate node of a route
Water drops	Control packets (Artificial IWD) used to construct and maintain routes
Obstacles and Soil on a river	Quality of a route (delay)

Therefore, a RFIWD packet is broadcasted from the source node until it reaches the destination node. While traveling, RFIWD memorizes information about all visited nodes: ID and reached time. When arrived at the destination node, a RBIWD packet is created and the RFIWD packet is destroyed. The RBIWD traverses the same nodes but in the reverse way. Its goal is to establish a route or multiple routes for every visited node i, and to initialize soil amount for all traversed links (i, j). The value of soil amount $Soil(i, j)$, is given by Eq. 8:

$$Soil(i,j) = QC \cdot delay_d(i,j) \tag{8}$$

where QC is a parameter of the protocol and $delay_d(i,j)$ is the end-to-end delay between current node i and destination node d when choosing j as the next hop.

4.3.2 Proactive Route Maintenance Phase

When a route is established between a source node s and a destination d, a proactive maintenance phase is launched (see Fig. 4). It consists of a periodic transmission of special control packet. Thus, at each interval of time, a PFIWD packet is sent from node s to d. The PFIWD contains the two proprieties of water drop: a velocity initialized with *INIT_VELOCITY* and an amount of soil initialized with zero. A node i receiving a PFIWD chooses j as the next hop for this packet according to a probability $P_d(i,j)$:

$$P_d(i,j) = \frac{f(Soil(i,j))}{\sum_{k \in Neighbors(i)} f(Soil(i,k))} \tag{9}$$

where

$$f(Soil(i,j)) = \frac{1}{\varepsilon + Soil(i,j)} \tag{10}$$

While traveling, PFIWD performs the following tasks:

- It maintains the values of soil amount of every traversed link. The dropped value is given by Eq. (11).

$$\Delta Soil(i,j) = \frac{a_s}{b_s + c_s \cdot g(i,j)} \tag{11}$$

where

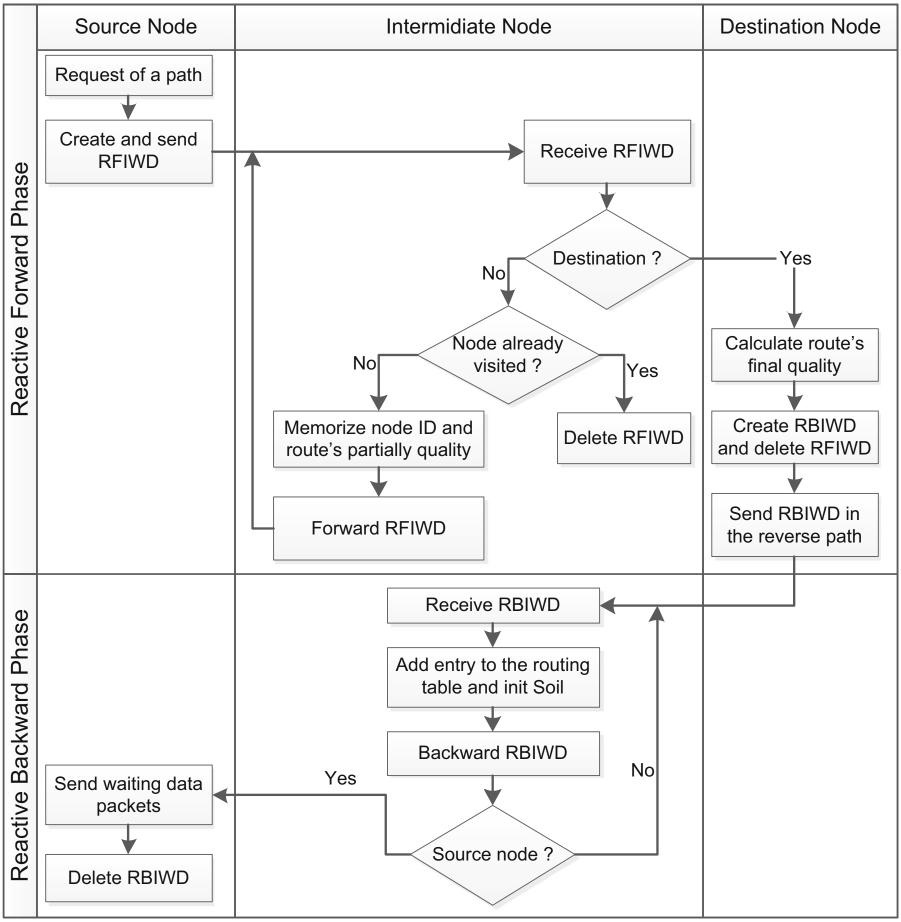


Fig. 3 reactive route discovery phase

$$g(i,j) = \left(\frac{delay_d(i,j)}{Vel(IWD)}\right)^2 \quad (12)$$

- It carries a supplementary amount of soil.

$$Soil(IWD) = Soil(IWD) + \Delta Soil(i,j) \quad (13)$$

- It adjusts its velocity

$$Vel(IWD) = Vel(IWD) + \Delta Vel(i,j) \quad (14)$$

where

$$\Delta Vel(i,j) = \frac{a_v}{b_v + c_v \cdot Soil(i,j)} \quad (15)$$

In the other hand, PBIWD is launched when a PFIWD reaches its destination. Its role is to update delay between current node and destination node, to be used next time by PFIWD.

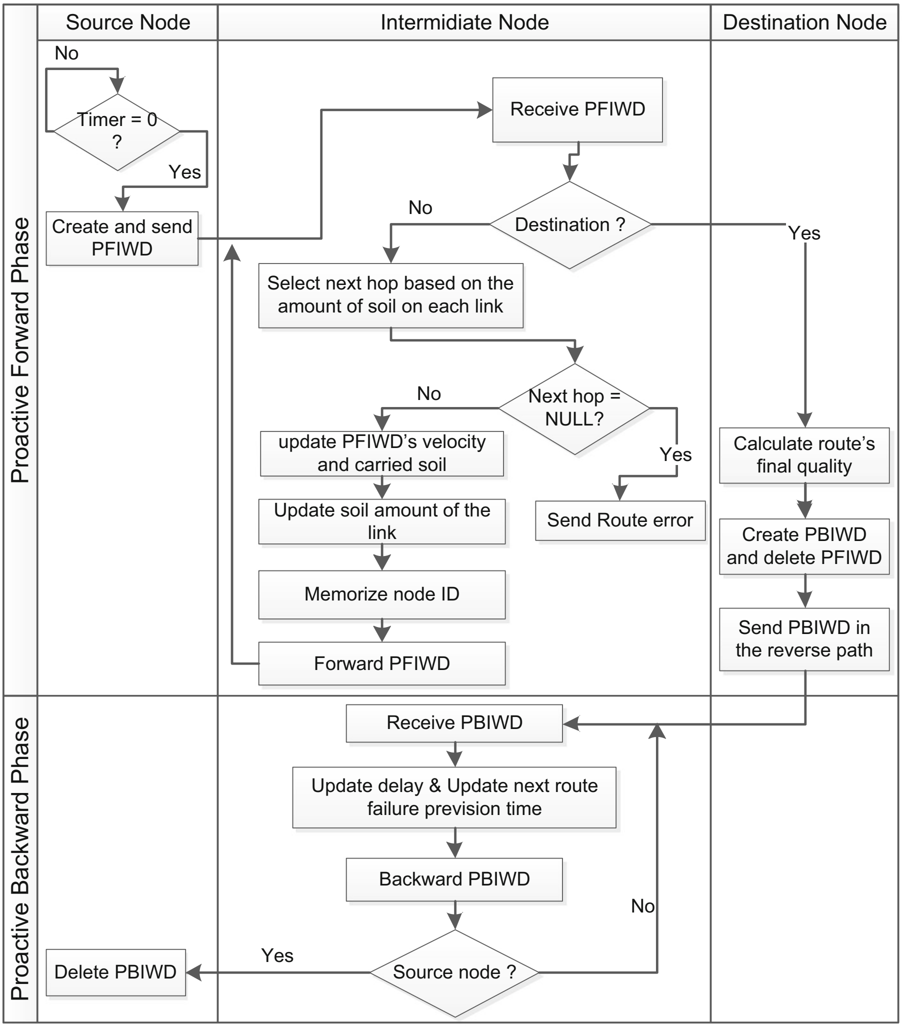


Fig. 4 Proactive route maintenance phase

4.3.3 Prediction of Route Failures

To avoid or reduce loss of data packets, a strategy should be defined to manage route failures. While most of routing algorithms are based on link failure detection by using hello messages or GPS information, in our algorithm each node implements a route failure prediction approach. It consists, first, to estimate the link failure time based on the received signal strength and then predict the route failure time.

A. Link Failure Prediction Time HELLO packets are used to maintain the estimated link failure time. Therefore, at each interval of time, a hello packet is broadcasted to all the immediate neighbors of a node. Based on the intensity of the received signal, the value of $next\_link\_failure\_time(j)$ of the corresponding neighbor j may be increased or decreased. When a node i receives a HELLO packet from another node j, it compares the received

Table 3 Link failure prediction using interpolation

T1	T2	NLFT=T3 = ?
Pr1	Pr2	PrTh

signal power with the last received one from the same node, and estimate the new value of *NLFT*(i, j) using interpolation mechanism (Table 3):

Thus, the value of *NLFT*(i, j) is obtained using the following equation:

$$NLFT(i,j) = T2 + \frac{(T2 - T1) \cdot (RxTh - Pr2)}{(Pr2 - Pr1)} \tag{16}$$

where *NLFT*(i, j) is the next link failure time of the link (i, j).

$T1$ the last received signal time,
$T2$ the current received signal time.
$Pr1$ the last received signal strength,
$Pr2$ the current received signal strength.
$RxTh$ is the signal reception threshold.

The pseudo-code of link failure estimation is given bellow (Algorithm 2).

Algorithm 2: Receive_hello_Pkt(p)
Input: *p* the received hello packet
i the current node
Pr the received signal power intensity

```
Begin
    If (p.neighbor_ID  is not a neighbor of i)
                    Add_neighbor(p.neighbor_ID);
    Pr1 = last_received_signal_power(p.neighbor_ID);
    T1 = last_received_signal_time(p.neighbor_ID);

    Update last received signal power of p.Neighbor_ID;
    Update last received signal time of p.Neighbor_ID;

    If (Pr1 ≠ MaxValue)
    Begin
          Pr2 = last_received_signal_power(p.neighbor_ID)
          T2 = last_received_signal_time(p.neighbor_ID)

          If (Pr2 < Pr1)  // neighbor node  moves  away
          Begin
               T_diff = (T2 – T1) * (RxThreashold – Pr2) / (Pr2 – Pr1)
               next_link_failure_time(p.neighbor_ID)  = T2 + T_diff
          EndIf
          Else  next_link_failure_time(p.neighbor_ID)  = MaxValue;
    EndIf
End
```

B. Route Failures Prediction Time The calculation of the Route failure prediction time is performed during the proactive route maintenance phase. In addition to soil amount maintenance performed by PBIWD packets, next route failure time is also updated. To do so, PBIWD contains an additional value, *LRFT*(*IWD*), which represents the estimated value of next route failure time of the last visited node *j*. Consequently, the next failure time of the route from current node *i* to destination *d* is obtained according to the following equation:

$$NRFT(i,d) = Min(LRFT(IWD), NLFT(i,j)) \tag{17}$$

where : *NLFT*(*i*, *j*) is the next failure time of the link (*i*, *j*).

The value of *NRFT*(*i*, *d*) is, then, encapsulated into the packet PBIWD to be used in the calculation of the next route failure time of the next node.

$$LRFT(IWD) = NRFT(i,d) \tag{18}$$

However, even though a route failure prediction mechanism has been introduced, route failures may occur. Consequently, RTERR packets are used to inform source node about the new modification.

5 Performance Evaluation

To evaluate the performance of the proposed protocol, we have used the network simulator NS2. The obtained results were, then, compared to those obtained using AODV-BFABL [24].

5.1 Parameters

Table 4 summarizes a list of all essential parameters and their values used during all simulations (unless specified otherwise, all parameters use values specified in Table 4). We consider a 50 nodes network dispersed randomly on a 1000 m × 1000 m area. Ten randomly selected nodes transmit data to respective destination nodes using CBR application agent. The nodes move according to Random Way Point mobility model.

Furthermore, Table 5 summarizes the parameter values of our protocol.

5.2 Performance Metrics

Three performance metrics, as listed below, have been used in order to evaluate how the two protocols perform.

- *End-to-end delay* the total elapsed time spent when delivering a packet from a source to a destination node.
- *Packet delivery ratio (PDR)* the ratio of data packet successfully delivery to the total of data packet generated by the traffic sources. The packet delivery ratio metric helps us in evaluating the robustness and effectiveness of a routing protocol.
- *Total number of route failures* in addition, we considered measuring the efficiency of the proposed technique for the route failure prediction time to be interesting. Thus, we introduced an additional metric: the total number of route failures occurred.

5.3 Simulation Results

In the following, we will present and discuss the results of three simulation setups. The goal is to explore how the two protocols will perform under various environmental parameters: (1) Node mobility, (2) packet size, and (3) network size.

5.3.1 Effect of Node Mobility

First, we evaluate the performance of each protocol as the node mobility increases. Node mobility is reflected with varying: (1) Pause Time (from 600 seconds until 0 second), or (2) node speed (between 5 and 25 m/s). The network is made up of 50 nodes placed randomly on an area of 1000 m $\times$ 1000 m with 10 concurrent traffics. A smaller pause time indicates a more dynamic network topology. The results are shown in Figs. 5, 6, 10.

As shown in these figures, both when varying pause time or node speed, the two protocols are sensitive to node mobility. However, IWDRP performs better than AODV-BFABL. From Figs. 5 and 6, we notice that IWDRP delivers packet faster than AODV-BFABL. It is observed that using IWDRP allows to save between 4.5% to 56% of delay. Figure 5 demonstrates that end-to-end delay is proportional to the mobility degree. However, when Pause time is less than 10 seconds, delay decreases. In some situations and scenarios, the node mobility can offer more connectivity and, thus, reduces the end-to-end delay. In the other hand, it is observed in Fig. 6 that IWDRP performs better in the case of high speeds (more than 20 m/s). The reason is that our approach is multipath, and routes are always maintained in order to consider new changes. These concepts should help reduce end-to-end delay and increase packet delivery ratio.

Figures 7 and 8 show the results of packet delivery ratio when varying pause time and speed respectively. The most important observation is the low ratio (less than 90 percent) for AODV-BFABL when speed exceeds 15 m/s (or 54 km/h). However, our proposal maintains an acceptable packet delivery ratio (more than 92 percent) even though when nodes move with high speeds. In addition, using our approach increases the PDR with a ratio of 1.3% to 6.2%.

Figure 9 plots the number of route failures under various pause time values. It is shown that for static and less mobile nodes, AODV-BFABL and IWDRP give approximately the

Table 4 Simulation parameters

Parameter	Value
Simulation time	600 s
Number of nodes	50
Simulation area	1000 m $\times$ 1000 m
Max Speed, v_max	20 m/s
Node speed's range	$[1..v_{max}]$m/s
Pause time	25 s
Transmission Radius	250 m
Number of CBR sources	10
Packet transmission rate	1 packet/s
Packet size	512
Physical layer	IEEE 802.11
Mobility model	RWP

Table 5 IWDRP parameters values (values adapted from [8, 32, 31])

Parameter	Value
a_v	100
b_v	0.01
c_v	1
a_s	0.1
b_s	0.01
c_s	100
INIT_SOIL	1000
QC	1000

same results. But, the difference between the results obtained by the two protocols increases more and more when pause time is decreasing. Our proposed protocol reduces the total number of route failures by 50% when pause time is zero. The route failure prediction mechanism, introduced in IWDRP, helps in reducing considerably route failures. The results shown in this figure confirm the remarks pointed out in Fig. 5 and 7 about high mobility (Pause time $<$ 15 seconds): when decreasing pause time value, node mobility increases, but it seems that node connectivity was increased for this scenario. This is to justify why the number of route failures decreases when pause time is smaller than 15 seconds.

In the same context, Fig. 10 confirms the obtained results shown in Fig. 9. In other words, network mobility increases when nodes speed is increasing leading to more route failures. However, our proposed protocol produces fewer failures than AODV-BFABL. IWDRP protocol anticipates route failures before they occur by estimating the next failure time for every established route. This mechanism helps in reducing the total number of route failures.

5.3.2 *Effect of Packet Size*

As expected, end-to-end delay increases when varying the size of data packets for both protocols. Figure 11 confirms this assumption. However, from Fig. 12, we observe that with the increase of packet size, ratio of successfully delivered packets is reduced. But, our proposal, always, achieves better than AODV-BFABL with about of 7.7% to 49% of delay improvement and 1% to 2.8% of PDR improvement in comparison to AODV-BFABL.

Results shown in Fig. 13 demonstrate that packet size has less effect on number of route failures. However, IWDRP allows between 30% and 42% of improvement.

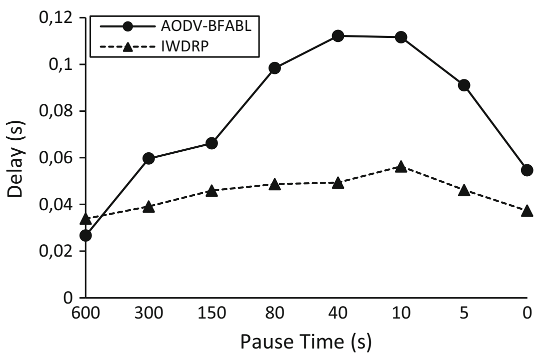


Fig. 5 Delay under various pause time

Fig. 6 Delay under various speed values

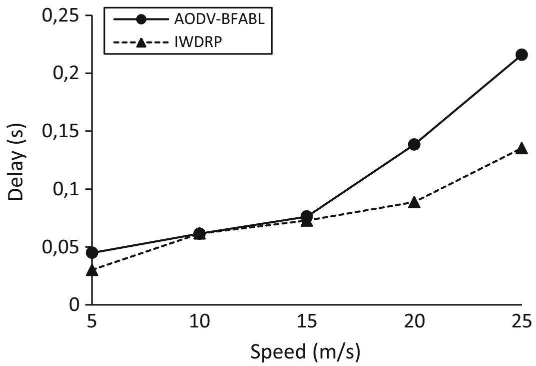


Fig. 7 PDR under various pause time

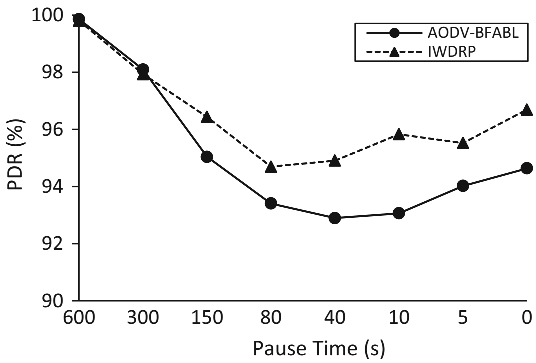
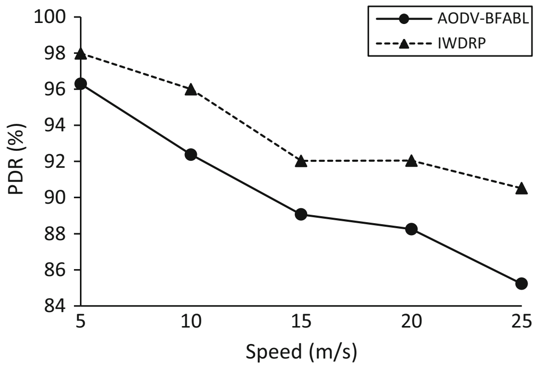


Fig. 8 PDR under various speed values



5.3.3 Effect of Network Size

We have studied in Figs. 14, 15 and 16 the impact of network size on the end-to-end delay, PDR and number of route failures respectively. To do so, we need to increase the total number of nodes when node density remains the same, as described in Table 6.

The obtained results confirm that IWDRP performs better than AODV-BFABL for the three metrics used for comparisons.

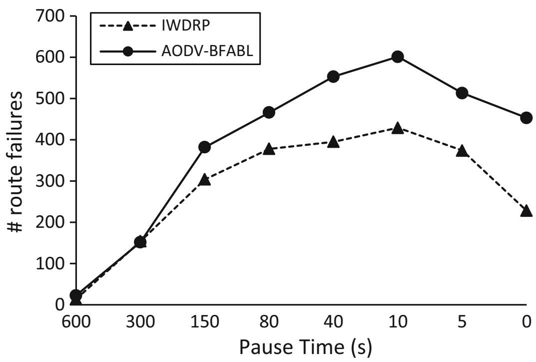


Fig. 9 Number of route failures under various pause time

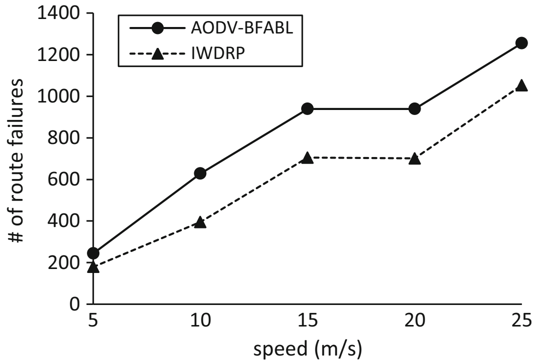


Fig. 10 Number of route failures under various speed values

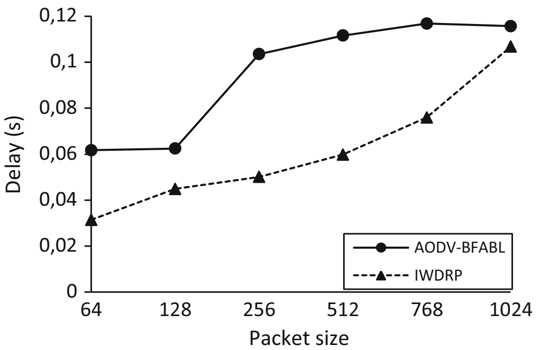


Fig. 11 Delay under various Packet sizes

6 Conclusion

This paper introduces, IWDRP, a new routing protocol for MANETs. It is based on a, relatively, new metaheuristic called Intelligent Water Drops optimization (IWD). It is inspired from the interactions observed between the water drops and the riverbed. We have observed many similarities between IWD metaheuristic and routing issue in MANETs. IWD approach has been used to construct and maintain routes between communicating nodes. However, IWDRP anticipates route failures using a route failure prediction

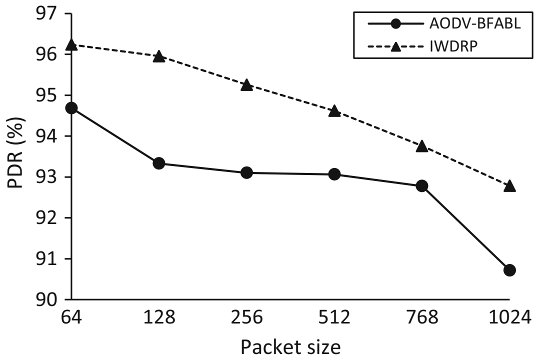


Fig. 12 PDR under various Packet sizes

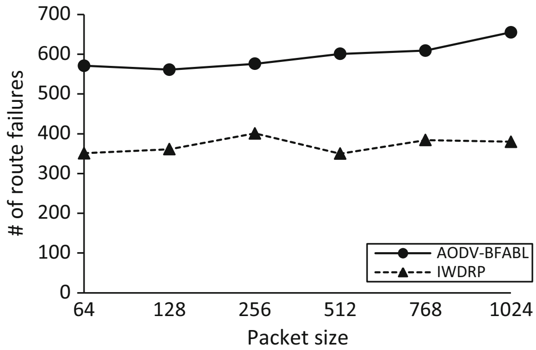


Fig. 13 Number of route failures under various Packet sizes

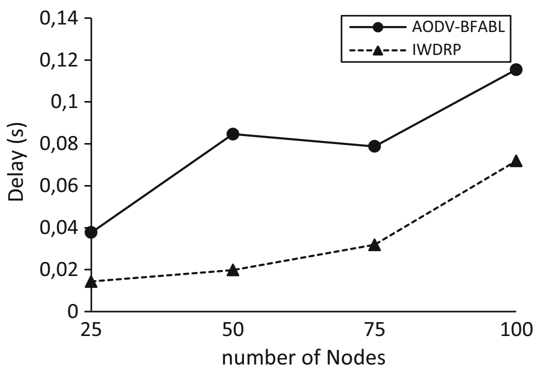


Fig. 14 Delay under various network sizes

approach. We have proposed a new scheme based on the received signal strength in order to estimate the next route failure time. Simulation results have shown the effectiveness of our algorithm in terms of packet delivery ratio, end-to-end delay and average number of route failures when compared to AODV-BFABL routing protocol.

As future work, we expect to add constraints to support quality of service (QoS) parameters like support of real time applications.

Fig. 15 PDR under various network sizes

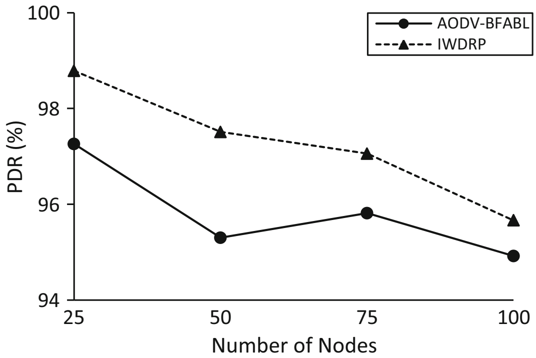


Fig. 16 Number of route failures under various network sizes

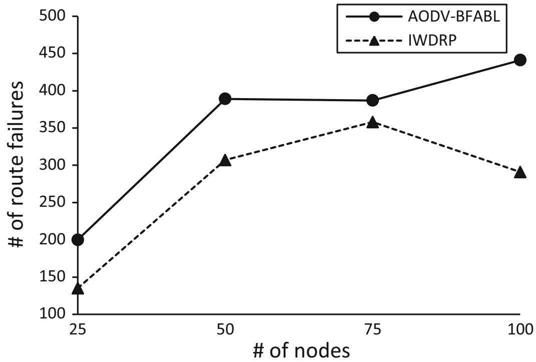


Table 6 Simulation area versus network size

Number of nodes	Simulation area
25	600 m × 600 m
50	1000 m × 1000 m
75	1200 m × 1200 m
100	1500 m × 1500 m

References

1. Amin, S., Al-Raweshidy, H., & Abbas, R. (2014). Smart data packet ad hoc routing protocol. *Computer Networks*, *62*, 162–181.
2. Clausen, T., Jacquet, P., Adjih, C., Laouiti, A., Minet, P., Muhlethaler, P., et al. (2003). Optimized link state routing protocol (OLSR). RFC 3626, IETF.
3. Di Caro, G. A. (1998). AntNet: Distributed stigmergetic control for communication networks. *Journal of Artificial Intelligence Research*, *9*, 317–365.
4. Di Caro, G. A., Ducatelle, F., & Gambardella, L. M. (2004). AntHocNet: An ant-based hybrid routing algorithm for Mobile Ad hoc Networks. In *Proceedings of parallel problem solving from nature-PPSN VIII* (pp. 461–470)
5. Dorigo, M., Di Caro, G. A., & Gambardella, L. M. (1999). Ant algorithms for discrete optimization. *Artificial life*, *5*(2), 137–172.
6. Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, *26*(1), 29–41.
7. Dorigo, M., & Birattari, M. (2010). Ant colony optimization. *Encyclopedia of Machine Learning* 36–39

8. Duan, H., Liu, S., & Wu, J. (2009). Novel Intelligent Water Drops optimization approach to single ucav smooth trajectory planning. *Aerospace Science and Technology*, *13*(8), 442–449.
9. Duan, H., Liu, S., & Lei, X. (2008). Air robot path planning based on Intelligent Water Drops optimization. In *IEEE international joint conference on neural networks, IEEE world congress on computational intelligence, IJCNN 2008* (pp. 1397–1401)
10. Ducatelle, F., Di Caro, G. A., & Gambardella, L. M. (2010). Principles and applications of swarm intelligence for adaptive routing in telecommunications networks. *Swarm Intelligence*, *4*(3), 173–198.
11. Farooq, M., & Di Caro, G.A. (2008). Routing protocols for next-generation networks inspired by collective behaviors of insect societies: An overview. In *Swarm intelligence* (pp. 101–160). Springer
12. Gunes, M., Sorges, U., & Bouazizi, I. (2002). ARA—The ant colony based routing algorithm for MANETs. In *Proceedings ICPP workshop on ad hoc networks IWAHN* (pp. 79–85)
13. Gurpreet, S., Neeraj, K., & Kumar, V. A. (2014). ANTALG: An innovative aco based routing algorithm for MANETs. *Journal of Network and Computer Applications*, *45*, 151–167.
14. Haas, Z. J., Pearlman, M. R., & Samar, P. (2002). The zone routing protocol (ZRP) for ad hoc networks. Tech. rep., Internet-Draft, draft-ietf-manet-zone-zrp-04.txt
15. Hoang, D. C., Kumar, R., & Panda, S. K. (2012). Optimal data aggregation tree in wireless sensor networks based on Intelligent Water Drops algorithm. *IET Wireless Sensor Systems*, *2*(3), 282–292.
16. Johnson, D. B., Maltz, D. A., Broch, J., et al. (2001). DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks. *Ad Hoc Networking*, *5*, 139–172.
17. Kamkar, I., Akbarzadeh-T, M.R., & Yaghoobi, M. (2010). Intelligent Water Drops a new optimization algorithm for solving the vehicle routing problem. In *IEEE international conference on systems man and cybernetics (SMC)* (pp. 4142–4146)
18. Karaboga, D. (2005). An idea based on honey bee swarm for numerical optimization. Tech. rep. Erciyes University, Engineering Faculty, Computer Engineering Department.
19. Kennedy, J., & Eberhart, R. C. (1995). Particle swarm optimization. *IEEE International Conference on Neural Networks*, *4*, 1942–1948.
20. Kiran, M., & Reddy, G. R. M. (2014). Design and evaluation of load balanced termite : A novel load aware bio inspired routing protocol for mobile ad hoc network. *Wireless Personal Communications*, *75*, 2053–2071.
21. Meisel, M., Pappas, V., & Zhang, L. (2010). A taxonomy of biologically inspired research in computer networking. *Computer Networks*, *54*(6), 901–916.
22. Niu, S., Ong, S., & Nee, A. (2012). An improved Intelligent Water Drops algorithm for achieving optimal job-shop scheduling solutions. *International Journal of Production Research*, *50*(15), 4192–4205.
23. Park, V., & Corson, S. (1997). Temporally-ordered routing algorithm (TORA) version 1 functional specification. Tech. rep., internet-draft, draft-ietf-manet-tora-spec-00.txt
24. Peng, Z., & Weihua, L. (2012). A bidirectional backup routing protocol for Mobile Ad hoc Networks. In *Second international conference on business computing and global informatization* (pp. 603–606)
25. Perkins, C. E., & Bhagwat, P. (1994). Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. *ACM SIGCOMM Computer Communication Review*, *24*, 234–244.
26. Perkins, C. E., Royer, E. M., & Das, S. R. (1999). Ad hoc on-demand distance vector (AODV) routing. In *Proceedings of IEEE workshop on mobile computing systems and applications* (pp. 90–100)
27. Rosati, L., Berioli, M., & Reali, G. (2008). On ant routing algorithms in ad hoc networks with critical connectivity. *Ad Hoc Networks*, *6*(6), 827–859.
28. Sayad, L., Aissani, D., & Bouallouche-Medjkoune, L. (2015). On-demand routing protocol with tabu search based local route repair in mobile ad hoc networks. *Wireless Personal Communications*. doi:10.1007/s11277-015-3081-z.
29. Semchedine, F., Bouallouche-Medjkoune, L., Bennacer, L., Aber, N., & Aissani, D. (2012). Routing protocol based on tabu search for wireless sensor networks. *Wireless Personal Communications, 67*(2), 105–112.
30. Shah-Hosseini, H. (2008). Intelligent Water Drops algorithm: A new optimization method for solving the multiple knapsack problem. *International Journal of Intelligent Computing and Cybernetics*, *1*(2), 193–212.
31. Shah-Hosseini, H. (2009). The Intelligent Water Drops algorithm: A nature-inspired swarm-based optimization algorithm. *International Journal of Bio-inspired Computation*, *1*(2), 71–79.
32. Shah-Hosseini, H. (2007). Problem solving by Intelligent Water Drops. In *IEEE congress on evolutionary computation CEC 2007* (pp. 3226–3231).
33. Shah-Hosseini, H. (2009). Optimization with the nature-inspired Intelligent Water Drops algorithm. In Evolutionary computation (pp. 297–320).

34. Shirkande, S. D., & Vatti, R. A. (2013). Aco based routing algorithms for ad-hoc network (wsn, manets): A survey. In *IEEE 2013 international conference on communication systems and network technologies (CSNT)* (pp 230–235).
35. Shubhajeet, C., & Swagatam, D. (2015). Ant colony optimization based enhanced dynamic source routing algorithm for mobile ad-hoc network. *Information Sciences*, *296*, 67–90.
36. Singh, G., Kumar, N., & Kumar Verma, A. (2012). Ant colony algorithms in manets: A review. *Journal of Network and Computer Applications*, *35*(6), 1964–1972.
37. Soheila, S., Hesam, O., & Homayun, M. (2013). An Intelligent Water Drops algorithm for solving robot path planning problem. In *14th IEEE international symposium on computational intelligence and informatics (CINTI 2013)*
38. Srinivas, M., & Patnaik, L. M. (1994). Genetic algorithms: A survey. *Computer*, *27*(6), 17–26.
39. Wedde, H. F., Farooq, M., et al. (2004). Bee ad hoc: An energy-aware scheduling and routing framework. Tech. rep., p 439, LSIII, School of Computer Science, University of Dortmund.
40. Yang, S., Cheng, H., & Wang, F. (2010). Genetic algorithms with immigrants and memory schemes for dynamic shortest path routing problems in Mobile Ad hoc Networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, *40*(1), 52–63.

Lamri Sayad is currently an Associate Professor at the University of Msila (Algeria) and a researcher at the LAMOS research unit of the University of Béjaïa. He received his “Magister” degree in Computer Science from Ecole Superieure d’informatique (ESI) Algeria in 2007 and, the doctorate degree in Computer Science from the University of Béjaïa in 2016. He is a member of the research group EPSIRT (Evaluation de Performances des Systèmes Informatiques et Réseaux de Télécommunication). His research interests are: Optimization of wireless networks (MANET, WSN, WMN, VANET), routing, network design, Application of computational intelligent methods and Meta-heuristics to various kinds of optimization problems such as data mining, routing, network design, vehicle routing problem.

Louiza Bouallouche-Medjkoune received the engineer degree in Computer Science from University of Sétif (Algeria), the Magister degree in Applied Mathematics from University of Béjaïa (Algeria), the doctorate’s degree in Computer Science from University of Sétif (Algeria) in 2006 and, the HDR (Habilitation à Diriger des Recherches) from the University of Constantine (Algeria) in 2009. BOUALLOUCHE-MEDJKOUNE works as a teacher at the Department of Computer Science of University of Béjaïa and as a researcher at the LAMOS Laboratory (Modeling and Optimization of Systems). She is head of the research team EPSIRT (Evaluation de Performances des Systèmes Informatiques et Réseaux de Télécommunication) since 2005 and head of department of Operation Research since January 2010. Her publications have appeared in various publishing houses: Taylor & Francis, Elsevier, Springer, AMS, BCS… Her research interests are: Performance evaluation of Computer Systems and Telecommunication Networks (Markov chains, Queuing networks, Simulation…), stability of systems, Quality of service, Routing and Protocols of systems and networks (mobiles, ad hoc, sensors…).

Djamil Aissani was born in 1956 in Biarritz (Basque Country, France). He started his career at the University of Constantine in 1978. He received his Ph.D. in 1983 from Kiev State University (Soviet Union). He is at the University of Béjaïa since it's opened in 1983/1984. Director of Research, Head of the Faculty of Science and Engineering Science (1999–2000), Director of the LAMOS Laboratory (Modelling and Optimisation of Systems), Scientific Head of the Doctoral Computer School (since 2004), he has taught in many universities (Algiers, Annaba, Rouen, Dijon, Montpellier...). He has published many papers on Markov chains, queuing systems, reliability theory, performance evaluation and their applications in such industrial areas as electrical networks and computer systems. He was the president of the national Mathematical Committee (Algerian Ministry of Higher Education and Scientific Research).