Applied Soft Computing 38 (2016) 321–328

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Dynamic fuzzy logic and reinforcement learning for adaptive energy efficient routing in mobile ad-hoc networks

Saloua Chettibi*, Salim Chikhi

MISC Laboratory, SCAL Team, University of Constantine 2, Algeria

ARTICLE INFO

Article history:
Received 28 August 2013
Received in revised form 1 September 2015
Accepted 3 September 2015
Available online 3 October 2015

Keywords:
MANETs
Routing protocol
Fuzzy logic system
Dynamic membership function
Reinforcement learning

ABSTRACT

In this paper, a dynamic fuzzy energy state based AODV (DFES-AODV) routing protocol for Mobile Ad-hoc NETworks (MANETs) is presented. In DFES-AODV route discovery phase, each node uses a Mamdani fuzzy logic system (FLS) to decide its Route REQuests (RREQs) forwarding probability. The FLS inputs are residual battery level and energy drain rate of mobile node. Unlike previous related-works, membership function of residual energy input is made dynamic. Also, a zero-order Takagi Sugeno FLS with the same inputs is used as a means of generalization for state-space in SARSA-AODV a reinforcement learning based energy-aware routing protocol. The simulation study confirms that using a dynamic fuzzy system ensures more energy efficiency in comparison to its static counterpart. Moreover, DFES-AODV exhibits similar performance to SARSA-AODV and its fuzzy extension FSARSA-AODV. Therefore, the use of dynamic fuzzy logic for adaptive routing in MANETs is recommended.

© 2015 Elsevier B.V. All rights reserved.

1. Introduction

A Mobile Ad-hoc Network (MANET) is a set of wireless devices having the ability to communicate without referring to any central communication infrastructure. Nodes in MANET are constantly moving. This leads to frequent topological changes and links breakage. Also, the wireless communication medium is errors-prone with limited bandwidth capacity. It follows that finding and maintaining routing paths in MANET are very challenging tasks.

In MANET, communicating devices are battery powered. Unfortunately, in many practical usage scenarios, batteries cannot be replaced or recharged. Moreover, battery exhaustion of nodes may not only result in low connectivity but can also lead to network partitioning. With the aim of extending MANET lifetime, energy-aware routing protocols have been proposed. Three main approaches are identified in the related literature [1]. First, maximum lifetime routing that seeks to balance energy expenditure among mobile nodes. Second, power-save approach that searches to minimize energy loss during the inactivity periods. Third, power-control approach where nodes adjust their transmission power so that a good compromise is found between goals of: maximizing network connectivity and minimizing energy dissipation.

MANETs are highly dynamic and uncertain environments. On one hand, changes in MANETs' topology, data-traffic load, bandwidth and energy resources are frequent. On the other hand, due to MANETs' dynamic and to their distributed organization, available routing information is uncertain and incomplete. To ensure good network performance, a routing protocol for MANETs must change its routing policy online to account for changes in network conditions [2] and to deal with routing information imprecision [3]. In other words, a routing protocol for MANETs should be adaptive. Different adaptivity contexts may be considered such as traffic, energy and mobility. This paper presents adaptive energy-efficient routing solutions.

Majority of proposed adaptive routing protocols use techniques derived from computational intelligence (CI) discipline. However, due to limited energy and computation resources in MANETs, some CI techniques are more suitable. This includes swarm intelligence [4,5], reinforcement learning [6–9] and fuzzy logic [10]. The main focus of this paper is on the use of fuzzy logic and reinforcement learning. Particularly, the following questions are answered. Firstly, which CI paradigm is more appropriate for adaptive energy efficient routing in MANETs: fuzzy logic or reinforcement learning? Secondly, does the combination of both paradigms ensure significant improvement? In fact, research in adaptation and hybridization in computational intelligence is receiving growing interest. Examples of recent related works include extensions of: krill herd algorithm [11–16], particle swarm optimization [17,18] and cuckoo search algorithm [19,20].

To the best of our knowledge, all fuzzy-logic based routing protocols for MANETs use static membership functions. In this paper, a dynamic membership function is defined to enhance the adaptivity

* Corresponding author. Tel.: +213 699 65 33 05; fax: +213 34 50 13 92.
E-mail address: sa.chettibi@gmail.com (S. Chettibi).

http://dx.doi.org/10.1016/j.asoc.2015.09.003
1568-4946/© 2015 Elsevier B.V. All rights reserved.

of the proposed fuzzy logic system. Moreover, a fuzzy extension of a previously proposed RL based routing protocol for MANETs [21] is presented. Finally, simulation results demonstrate that the proposed dynamic fuzzy logic system ensures similar performance to those achieved by reinforcement learning system. Thus, the use of dynamic fuzzy logic for the task of adaptive energy aware routing in MANETs is more appropriate.

The rest of this paper is organized as follows. Section 2 discusses the related work covering adaptive energy aware routing in MANETs using fuzzy logic and reinforcement learning. Section 3 provides an overview of the basics of fuzzy logic and reinforcement learning. Moreover, the AODV routing protocol is presented. Section 4 describes the proposed adaptive fuzzy logic system for Route REQuests (RREQs) probability forwarding tuning problem. In Section 5, SARSA-AODV protocol [21] is presented with its fuzzy extension. Section 6 reports simulation results. Finally, Section 7 concludes the paper.

2. Related work

This section describes major related works to adaptive energy aware routing in MANETs using fuzzy logic and reinforcement learning. Clearly, both techniques are adequate to achieve adaptivity feature. However, no prior work has compared the performance of each technique. The present work comes to fill this gap.

2.1. Adaptive energy aware routing in MANETs using fuzzy logic

Typically, an energy-aware routing protocol based on fuzzy reasoning uses FLS either for adjusting some routing parameters or for estimating the energy-cost of a routing path. For example, in [22], fuzzy path selection power-based AODV (FPSP-AODV) is proposed. To select a path for routing, number of hops, bandwidth and node remaining power are used to evaluate its cost. In [23], a fuzzy-based virtual backbone (FVB) routing scheme for large-scale mobile ad hoc networks is presented. FVB aims to maximize the network lifetime. For this purpose, the authors have developed a FLS with the following inputs: node residual energy, traffic and mobility. The FLS output indicates the node eligibility to be a cluster-head. Abirami et al. [24] suggested using a FLS to choose energy efficient paths after a route discovery procedure. The FLS inputs are battery cost and power consumption of discovered paths. Hiremath et al. [25] designed an adaptive energy efficient reactive routing protocol where mobile nodes use fuzzy residual-energy thresholds to decide RREQ forwarding. Chettibi and Chikhi [26] have extended OLSR proactive routing protocol with a FLS for energy aware routing. Remaining energy and expected residual lifetime are used by a node to adjust its willingness parameter that reflects its ability to act as a router.

2.2. Adaptive energy aware routing in MANETs using reinforcement learning

A mobile node implementing a RL-based routing protocol learns either how to adjust some routing parameters or how to make routing decisions (i.e. choosing next-hop or path for routing). The energy aware-routing problem in MANETs was tackled as a RL problem in [21,27,28]. In [27], each node learns how to choose its next-hop for routing. This is by using a stochastic gradient descent RL algorithm. A node decision depends on its neighbors' selfishness and remaining energy. Naruephiphat et al. [28] have proposed a RL-model that aims to balance objectives of maximizing nodes lifetime and minimizing energy consumption. Each source node runs the first-visit on policy Monte Carlo (ONMC) RL algorithm to learn how to choose among: the minimum-energy path, the max–min residual battery path, and the minimum-cost path. Chettibi and Chikhi [21] have formulated the problem of RREQs forwarding rate in an energy aware route discovery procedure as a reinforcement learning task. Each node decides the amount of RREQs to forward according to its expected residual lifetime. The learning goal is to balance energy consumption among nodes as a means for extending network lifetime.

3. Applied methods and routing protocol

In this section, preliminaries on fuzzy logic systems and reinforcement learning technique are described. Since all the proposed routing policies in this paper are implemented on the top of AODV protocol, Section 3.3 is dedicated to its presentation.

3.1. Fuzzy logic

Human beings are able to make decisions even in the presence of imprecise or incomplete knowledge. Fuzzy logic allows approximating human reasoning. In fuzzy set theory, initialized by Zadeh [29], an element can be a member of a set with a certain degree. A fuzzy logic system (FLS) uses fuzzy sets to make decisions. A FLS can be seen as an expert system that encompasses a set of linguistic fuzzy rules. Fuzzy rules follow this general pattern: *If premises(s) Then conclusion(s).* In a fuzzy rule, premises and conclusions correspond to fuzzy input and fuzzy output sets, respectively.

As shown in Fig. 1, a FLS contains four main modules. First, fuzzification module that transforms the crisp numbers inputs into fuzzy sets. This is by using membership functions. Second, knowledge base stores the IF-THEN rules. Third, inference engine is used to establish fuzzy conclusions. Finally, DEFuzzification module transforms the obtained fuzzy conclusion into a crisp value.

Mamdani [30] and Takagi Sugeno (TSK) [31] are the most utilized FLS models in the literature. They mainly differ in the output structure. In Mamdani systems, both inputs and outputs are fuzzy propositions using linguistic variables. In TSK models, system outputs are numerical values rather than linguistic variables. The outputs can be constants, polynomials, functions or differential equations.

3.2. Reinforcement learning

Reinforcement learning (RL) [32] is an efficient method for discovering policies in Markovian sequential decisions tasks. Whenever an RL agent takes an action, the environment responds by a reward or a punishment signal. The feedback signal gives an indication about the quality of undertaken actions. As depicted in Fig. 2, an RL agent interacts with its environment in discrete time steps. At a time step t, the environment state is s_t. The RL agent chooses an action a_t. Consequently, it receives a feedback r_t and a new state s_{t+1} is determined. This cycle is repeated until that the learning agent converges to an optimal policy maximizing the expected future reward.

When state transition and reward functions are known, dynamic programming [32] can be successfully applied to find an optimal policy. However, in practice, RL agents do not have a complete knowledge about their environments' models. In such circumstances, temporal difference (TD) and Monte-Carlo (MC) RL algorithms [32] are more suitable.

In this paper, SARSA, a well-known TD algorithm (see Fig. 3), is used. It is based on evaluation of action-value function denoted by $Q^{\pi}(s, a)$. This latter estimates the expected future reward to the agent when it performs a given action, a, in a given state, s, and follows the policy π thereafter. At every time step, SARSA updates the action-value function Q^{π} using the quintuple $(s_t, a_t, r_t, s_{t+1}, a_{t+1})$, which gives rise to the name of the algorithm. SARSA is an on-policy RL algorithm. It uses the learned policy not only to take decisions

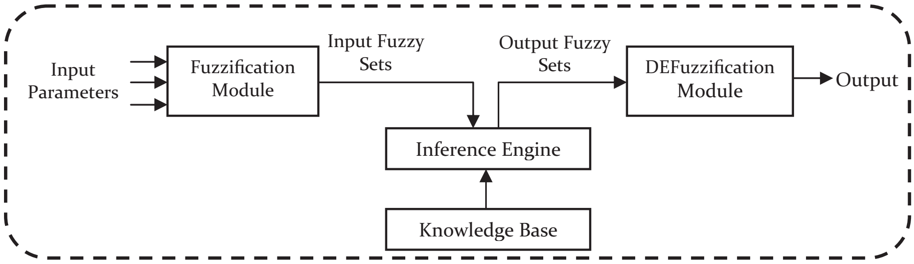

Fig. 1. Modules of a fuzzy logic system.

but to update Q-values as well. In Fig. 3, α and γ denote the learning rate and the discount factor, respectively.

3.3. Ad hoc on demand distance vector (AODV) routing protocol

AODV as specified in [33] is a reactive routing protocol for MANETs. Only a node that requires a route toward a given destination broadcasts a Route REQuest (RREQ) packet. On receiving a RREQ, a node that maintains a valid route can directly answer. Otherwise, the RREQ is rebroadcasted again. The same operation is repeated until a path is found. A node that has an answer sends a Route REPly (RREP) packet toward the source node. The RREP tracks the reverse route already taken by the corresponding RREQ.

An example of route discovery operation in AODV protocol is depicted in Fig. 4. Node A looks for a path toward node J. Hence, RREQs packets are broadcasted in the entire network. On receiving the first RREQ from node H, destination node J generates a RREP packet.

In reactive route discovery, the likelihood of a node to be chosen as a router is proportional to the amount of RREQs that it forwards. Therefore, node willingness to participate in routing can be reflected by tuning its RREQ forwarding ratio. In the perspective of maximizing network lifetime, node willingness to participate in routing is adjusted according to its energy profile. This approach is known in the literature as local maximum lifetime routing [34].

4. Adaptive fuzzy logic system for energy-aware RREQs probability forwarding tuning

To deal with the problem of energy aware RREQs probability forwarding tuning, an adaptive FLS is proposed in this section. Each node in the network uses this FLS to infer a fuzzy willingness value.

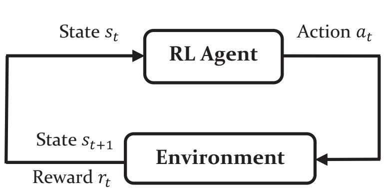

Fig. 2. Reinforcement learning agent interaction with its environment.

Initialization:
  Initialize $Q(s, a)$ and s_t;
Choose an action a_t using any action selection rule derived from Q;
Repeat for each time-step
  1. Take a_t;
  2. Observe the reward r_t and the state s_{t+1};
  3. Choose an action a_{t+1} using any action selection rule derived from Q
  4.Update $Q(s_t, a_t) : Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$
  5. $t \leftarrow t + 1$;
Until the terminal state is reached.

Fig. 3. SARSA RL algorithm.

The FLS inputs are: the node residual energy (RE) and its energy drain rate (EDR). Unlike previous related-works, the membership function of the RE input is made dynamic. Willingness (W) computation is accomplished by using fuzzy If-Then mapping rules given in Table 1.

4.1. Fuzzy inputs and fuzzification

As mentioned above, two fuzzy inputs are used: RE and EDR. The same linguistic variables for these fuzzy sets are used, namely: *Low* and *High*. Linguistic variables *Low* and *High* have Trapezoidal-shaped membership functions. Fuzzy sets describing these input variables are depicted in Figs. 5 and 6. A trapezoidal curve is a function $\mu(x)$ depending on four scalar parameters: a, b, c, and d. Parameters a and d locate the *feet* of the trapezoid, whereas parameters b and c locate the *shoulders*. Function $\mu(x)$ is defined as follows:

$$\mu(x) = \begin{cases} 0 & \text{if } x \le a \\ \dfrac{x-a}{b-a} & \text{if } a \le x \le b \\ 1 & \text{if } b \le x \le c \\ \dfrac{d-x}{d-c} & \text{if } c \le x \le d \\ 0 & \text{if } b \le x \end{cases} \tag{1}$$

For RE input, if node residual energy is less than 40% of its initial energy then it is considered to be *Low*. If it is greater than 90% of its initial energy, then it is considered to be *High*. However, what is actually considered to be a low residual energy will not remain the same later. This is because nodes constantly drain their batteries until their complete depletion. Therefore, the membership function for RE input should be time varying.

Let us denote by EMin and EMax the membership function parameters of RE input. EMin and EMax are initially set to 4 and 9 J, respectively. If the same EMin and EMax values are maintained for

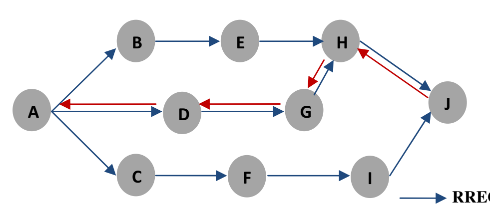

Fig. 4. Route discovery procedure in AODV protocol.

Table 1
Fuzzy If-Then mapping rules for willingness computation.

Rule 1	If RE is Low and EDR is High Then *W* is Very Low
Rule 2	If RE is Low and EDR is Low Then *W* is Low
Rule 3	If RE is High and EDR is High Then *W* is Medium
Rule 4	If RE is High and EDR is Low Then *W* is High

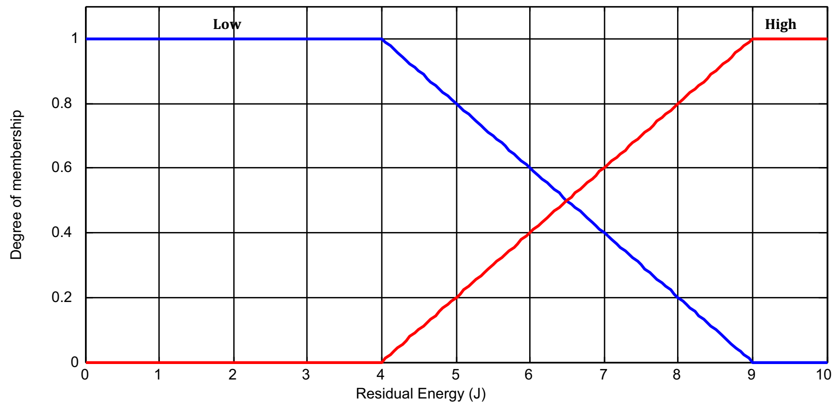

Fig. 5. Fuzzy sets of RE input with an initial energy of 10 J.

all network operation time, then nodes behavior will tend toward selfishness. Nodes with low residual energy, relatively to the prefixed EMin value, will set their willingness to be *Low* or *Very Low*. Consequently, nodes energy will be economized but at the cost of degrading data packets delivery ratio. However, to achieve energy efficiency, objectives of maximizing both network lifetime and throughput should be balanced. In other words, energy conservation should never come at the cost of reduced network throughput.

In order to avoid selfishness problem, EMax and EMin parameters are adjusted so that constant decrease in residual energy is reflected. For this purpose, Hello packets that are used by AODV protocol for maintaining connectivity are exploited as follows. Each node piggybacks its residual energy value in broadcasted Hello messages. At each T sampling time interval, each node computes the mean of its neighbors' residual energies MRE (*itself included*). MRE is used to update EMax parameter as shown in Eq. (2). Next, EMin is computed using Eq. (3).

$$\text{EMax} = \begin{cases} \text{MRE} - 1 & \text{if MRE} > 1 \\ 1 & \text{if MRE} < 1 \end{cases} \tag{2}$$

$$\text{EMin} = \frac{\text{EMax}}{2} \tag{3}$$

Whenever EMax starts to take negative values (MRE falls below 1 J), EMax and EMin are fixed at 1 and 0.5 values, respectively. To illustrate adaptive tuning of EMax and EMin parameters, an example extracted from simulation trace-file is provided (see Fig. 6). This example shows how a node updates its EMin and EMax parameters

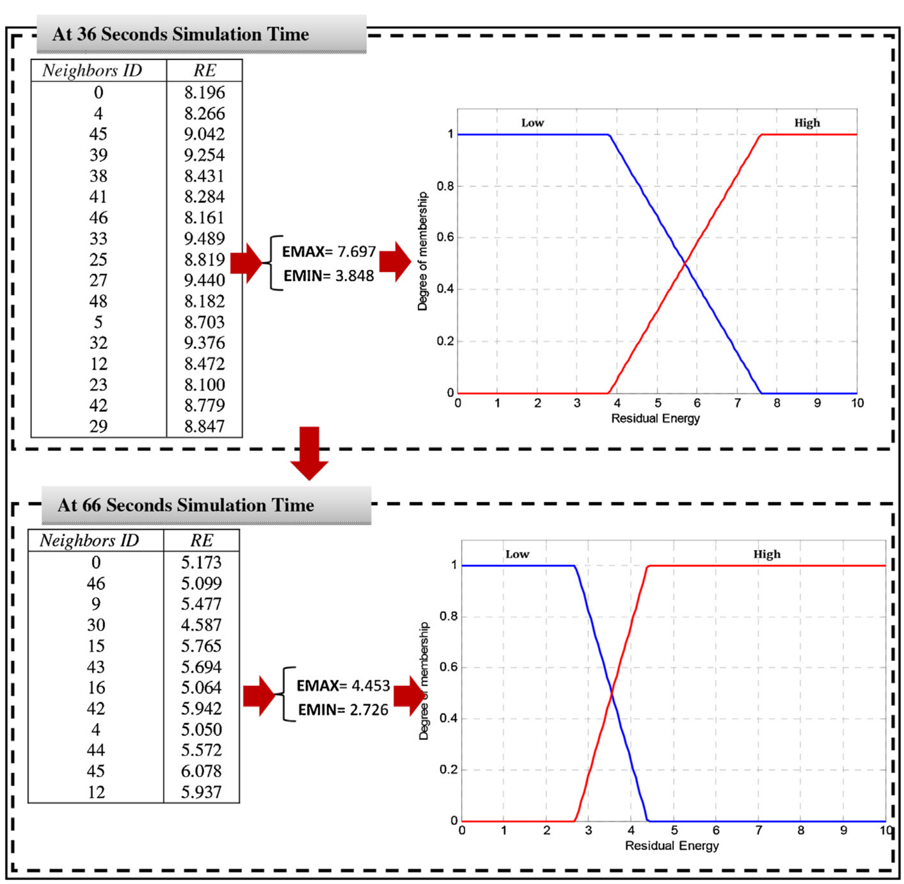

Fig. 6. Illustration of adaptive tuning of EMIN and EMAX parameters.

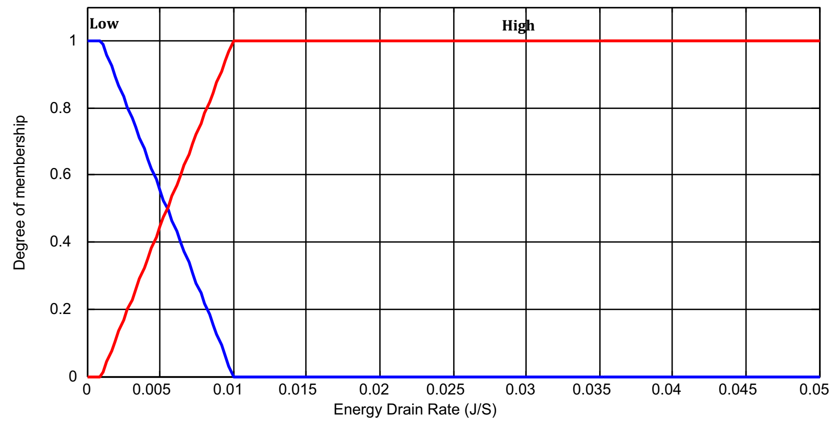

Fig. 7. Fuzzy sets for EDR input with an initial energy of 10 J.

according to its neighbors' residual energies at simulation times: 36 and 66 s. Naturally, list of neighbors changes over time due to nodes mobility.

For EDR input, consumption less than 0.001 J/s is considered to be *Low*, whereas EDR greater than 0.01 J/s is considered to be *High*. The EDR of a node at time step *t* is calculated using the exponential moving average method as follows:

$$\mathrm{EDR}_t = 0.3 \times \mathrm{EDR}_{t-1} + 0.7 \times \mathrm{EDR\_Sample}_t \quad (4)$$

EDR_{t-1} and $\mathrm{EDR\_Sample}_t$ correspond, respectively, to the old and the newly calculated energy drain rate values. To compute the energy drain rate per second, each node monitors its energy consumption during a *T* seconds sampling interval [35] (Fig. 7).

4.2. Fuzzy output and DEFuzzification

In the proposed FLS, the fuzzy output is node willingness. Node willingness is described by four linguistic variables, namely: *Very Low*, *Low*, *Medium* and *High*. The output has triangular shaped membership functions as illustrated in Fig. 8. The obtained crisp value by DEFuzzification procedure corresponds to node forwarding probability of RREQs. Note that the smallest of maximum (SOM) DEFuzzification method is applied.

5. Energy aware fuzzy-SARSA based AODV

For comparison purpose, the energy aware SARSA-AODV protocol [21] is implemented. In SARSA-AODV, each node learns to tune its RREQs forwarding rate during route discovery process using SARSA RL algorithm. The proposed RL model is as follows. Mobile node state at time step t, s_t, corresponds to its expected residual lifetime (RT) in seconds. The state space is experimentally quantized into discrete intervals. Actions correspond to the ratio of RREQs to be forwarded by a node. Finally, the reward signal is defined so that energy consumption fairness among mobile nodes is enhanced. It is calculated at each node as follows:

$$R_t = \begin{cases} +\dfrac{\mathrm{RE}_t}{\mathrm{IE}} \times \dfrac{dv_t}{dv_t+1} & \mathrm{IF}\, dv_t > 0 \\ -\dfrac{\mathrm{CE}_t}{\mathrm{IE}} \times \dfrac{|dv_t|}{|dv_t|+1} & \mathrm{IF}\, dv_t < 0 \\ 0 & \mathrm{IF}\, dv_t = 0 \,\mathrm{OR}\, N = 0 \end{cases} \quad (5)$$

In Eq. (5), RE_t, CE_t and IE denote, respectively, residual energy, consumed energy and initial energy at time-step *t*. dv_t is the energy drain rate deviation of current node in comparison to its neighbors. In SARSA-AODV implementation, the set of actions A is discretized into the set $\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$. Also, $s_t \in \{[0, 100), [100, 200), [200, 300), [300, \infty)\}$.

As mentioned above, the state space in SARSA-AODV is experimentally discretized. Thus, the obtained discretization is not general enough. If used under different experimental settings, it may leads to poor performance. Therefore, fuzzy logic is used as means to introduce generalization in the state space. This gives rise to a fuzzy extension of SARSA-AODV baptized FSARSA-AODV.

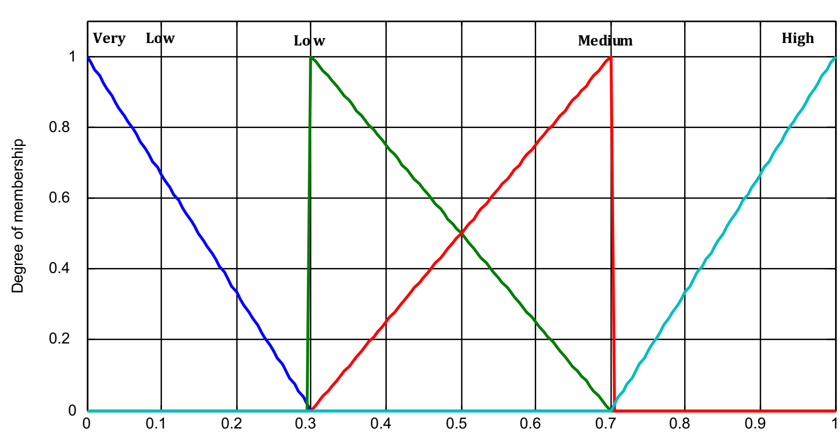

Fig. 8. Fuzzy sets for willingness output.

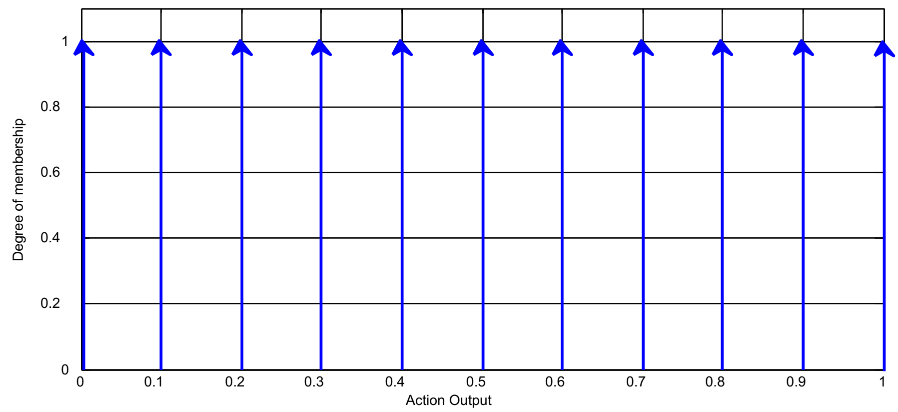

Fig. 9. Membership function for action output.

5.1. Fuzzy SARSA algorithm

FSARSA-AODV implements fuzzy SARSA algorithm to learn RREQs forwarding probability. The proposed fuzzy-SARSA algorithm has a critic only architecture [32]. In a critic only architecture, a FLS is used for approximating the action value function $Q(s, a)$. Action selection probability of the output depends on this approximation.

The proposed FLS is a two-input/one-output zero-order TSK fuzzy system with four rules. Note that the same inputs RE and EDR described in Section 4.1 are maintained. In SARSA-AODV, node state corresponds to its expected residual lifetime (RT). RT is computed as the ratio of node residual energy to its energy drain rate. This explains how RE and EDR can substitute RT in FSARSA-AODV.

Each inference rule in the proposed FLS, R_i, has the general form: If RE is L_{i1} and EDR is L_{i2} Then (a_{i1} with value $Q(i, 1)$ or ... or a_{im} with value $Q(i, m)$). Where $L_i = L_{i1} \times L_{i2}$ is the two-dimensional fuzzy set of the ith rule. The number of possible discrete actions for each rule is m; a_{ij} denotes the jth candidate action and $Q(i,j)$ is the Q-value of the jth action in the ith rule. The membership function of the FLS output, Action, has 11 singleton functions as shown in Fig. 9. The fuzzy inference rules that we used are described in Table 2.

In fuzzy-SARSA algorithm, actions are chosen using a softmax action selection rule. In this latter, the best action is given the highest selection probability; whereas, all remaining actions are ordered in function of their estimated Q values. In particular, a modified Boltzmann softmax action selection rule proposed in [36] is applied. In rule i, action a_{ij} is chosen with probability:

$$P(a_{ij}) = \frac{\exp(\mu_i Q(i,j)/\mathrm{Temp})}{\sum_{m}^{k=1} \exp(\mu_i Q(i,k)/\mathrm{Temp})} \tag{6}$$

Table 2
Fuzzy If-Then mapping rules for choosing RREQs forwarding probability in FSARSA-AODV.

Rule 1	If RE is Low and EDR is High Then (0 with value Q(0, 0) or 0.1 with Q(0, 1) or 0.2 with Q(0, 2) or 0.3 with Q(0, 3))
Rule 2	If RE is Low and EDR is Low Then (0.3 with value Q(1, 0) or 0.4 with Q(1, 1) or 0.5 with Q(1, 2))
Rule 3	If RE is High and EDR is High Then (0.5 with value Q(2, 0) or 0.6 with Q(2, 1) or 0.7 with Q(2, 2))
Rule 4	If RE is High and EDR is Low Then (0.7 with value Q(3, 0) or 0.8 with Q(3, 1) or 0.9 with Q(3, 2) or 1 with Q(3, 3))

Temp is a positive parameter called temperature. The firing strength of each rule i, μ_i, is computed as follows:

$$\mu_i = \mathrm{MIN}(\mu_{\mathrm{RE\,is}\,L_{i1}}(\mathrm{RE}), \mu_{\mathrm{EDR\,is}\,L_{i2}}(\mathrm{EDR})) \tag{7}$$

The rule with the highest firing strength will be used to compute the system output. Whenever the reward and the new state are observed, the Q-value function is updated using the standard SARSA formula. To summarize, fuzzy-SARSA algorithm is depicted in Fig. 10.

6. Simulation results

In this section, the performance of four routing protocols, namely: DFES-AODV, SFES-AODV, SARSA-AODV and FSARSA-AODV are evaluated. Note that DFES-AODV is the implementation of the FLS described in Section 4 on the top of AODV routing protocol. SFES-AODV protocol uses the same FLS but with static membership functions. All mentioned protocols are implemented with NS-2 simulator [37]. Simulation parameters are reviewed in Table 3.

In experiments, the following metrics are measured: data packets delivery ratio (PDR), time to half nodes energy die (THNED) and end to end delay (EED). Each result, in Table 4, corresponds to the mean of 40 simulation runs with random mobility scenarios.

As previously explained, extending network lifetime should never come at the cost of network throughput degradation. Therefore, energy efficiency is the most appropriate metric to evaluate the performance of any maximum-lifetime routing policy. In fact, energy-efficient routing can be seen as multi-objective optimization problem. Hence, to find the protocol p that presents the best compromise between maximizing network lifetime and throughput goals, the weighted TchebyCheff method is applied [38].

Initialization:
$Q(s_t, a_t) = 0$ for all s_t and all a_t
Repeat For each time step t :
1. RE Residual energy of Current ~~n~~node.
EDR ← Energy Drain Rate of current node.
2. s_t ← Fuzzification (RE, EDR)
3. Calculate the firing strength of each rule using the MIN operator.
4. Choose the rule R_i to run:
$\mu(R_i) \leftarrow \max_{j=1,4} \mu(R_j)$
5. Choose action a_{ij} using the modified Boltzman action selection rule.
6. Observe the reward r_t and the state s_{t+1}.
7. Update $Q(s_t, a_t)$:
$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$
8. $t \leftarrow t + 1$;
Until the terminal state is reached.

Fig. 10. Fuzzy SARSA algorithm.

Table 3
Simulation setting.

Simulation parameter	Value
Network scale	800 m × 800 m
Simulation time	900 s
Number of nodes	50
Mobility model	Random way point
Maximum velocity	5 m/s
Pause time	0 s
Traffic type	CBR
Connections number	10
Packets transmission rate	4 packets/s
Initial energy	10 J
Transmission power	0.6 W
Reception power	0.3 W
T sampling interval	6 s
Learning rate, α	0.6
Discount factor, γ	0.4
Temperature, temp	0.1

Table 4
PDR, THNED and EED results.

Protocol	PDR	THNED	EED
SFES-AODV	38.8590	219.6413	0.1646
DFES-AODV	54.5859	131.6042	0.0926
SARSA-AODV	55.9527	129.7382	0.0779
FSARSA-AODV	56.3261	129.5932	0.0819

According to TchebyCheff method definition, the best compromise protocol is one that minimizes the maximal differences between ideal and actual values of PDR and THNED. Each protocol, p, is characterized by a cost vector $C = (\text{PDR}_p, \text{THNED}_p)$. Formally, the best protocol is one that minimizes the function $f(C, R, W)$ defined as follows:

$$f(C, R, W) = \max\left\{\frac{w_1\left|\text{PDR}_p - R_1\right|}{R_1}, \frac{w_2\left|\text{THNED}_p - R_2\right|}{R_2}\right\} \tag{8}$$

where R is the reference vector with ideal values and W is the weighting vector. In this study, maximizing network PDR is given the highest importance by setting $W = (0.75, 0.25)$. Reference vector is fixed at $(100, 900)$. The first component, 100, corresponds to the best PDR with no data loss. The second component, 900, represents the best THNED that can be achieved with a simulation time of 900 s. Obtained results by TchebyCheff method are reviewed in Table 5.

As can be seen from Table 4, SFES-AODV has marked higher THNED than DFES-AODV protocol but at the cost of its PDR degradation. Moreover, SFES-AODV has shown the worst EED result which reflects unbalanced load distribution among nodes. Measurement of the TchebyCheff norm in Table 5 reveals that DFES-AODV is more energy efficient than SFES-AODV. This confirms the usefulness of the proposed dynamic membership function.

FSARSA-AODV solves the problem of experimental discretization of state space in SARSAAODV without degrading its performance. If the experimental setting is changed, no modification in FSARSA-AODV will be needed. This is in contrast to SARSA-AODV that will require a novel discretization for its state space.

Table 5
TchebyCheff norm computation for energy efficiency evaluation.

Protocol	$\frac{w_1\lvert\text{PDR}_p - R_1\rvert}{R_1}$	$\frac{w_2\lvert\text{THNED}_p - R_2\rvert}{R_2}$	$f(C, R, W)$
SFES-AODV	0.4585	0.1889	0.4585
DFES-AODV	0.3406	0.2134	0.3406
SARSA-AODV	0.3303	0.2139	0.3303
FSARSA-AODV	0.3275	0.2140	0.3275

Finally, DFES-AODV and FSARSA-AODV achieve very close results in terms of PDR, THNED and EED metrics. In addition, DFES-AODV and FSARSA-AODV do not differ much in terms of energy efficiency. It can be concluded that the dynamic fuzzy model ensures similar performance to the reinforcement leaning model. However, in comparison to DFES-AODV, there are many parameters to be adjusted in FSARSA-AODV, namely: learning rate, discount factor and temperature parameter. Therefore, the use of DFES-AODV for adaptive energy aware routing in MANETs is recommended.

7. Conclusions

MANETs are highly dynamic and uncertain environments with limited computing, storage and energy resources. Hence, a routing protocol for MANETs should be adaptive and energy efficient. To meet the adaptivity feature, computational intelligence techniques are generally applied. In particular, fuzzy reasoning and reinforcement learning are widely accepted due to their low computational and memory requirements.

This paper deals with the problem of adaptive energy-aware RREQ probability tuning in AODV reactive routing protocol. Two extensions of AODV are proposed and compared, namely: DFES-AODV and FSARSA-AODV. First, DFES-AODV implements a fuzzy logic system with a time varying membership function for residual energy input. In comparison to SFES-AODV that is built on the top of a traditional fuzzy logic system, DFES-AODV is more energy efficient. Second, FSARSA-AODV uses a fuzzy extension with a critic only architecture of SARSA RL algorithm. This hybrid algorithm overcomes the problem of empirical discretization of state space in SARSA-AODV protocol.

It has been shown through simulation that DFES-AODV and FSARSA-AODV exhibit similar performance. However, FSARSA-AODV still requires empirical adjustment of several functional parameters. In contrast to FSARSA-AODV, DFES-AODV is self-adaptive. In conclusion, dynamic fuzzy logic is more appropriate than reinforcement learning for adaptive energy aware routing in MANETs.

References

[1] J. Li, D. Cordes, J. Zhang, Power-aware routing protocols in ad hoc wireless networks, IEEE Wirel. Commun. 12 (2005) 69–81.
[2] F. Ducatelle, G. Di Caro, L. Gambardella, Principles and applications of swarm intelligence for adaptive routing in telecommunications networks, Swarm Intell. 4 (2010) 173–198.
[3] E. Natsheh, A survey on fuzzy reasoning applications for routing protocols in wireless ad hoc networks, Int. J. Bus. Data Commun. Netw. 4 (2008) 22–37.
[4] R. Kulkarni, A. Forster, G. Venayagamoorthy, Computational intelligence in wireless sensor networks: a survey, IEEE Commun. Surv. Tutor. 13 (2011) 68–96.
[5] G. Singh, N. Kumar, A. Verma, Ant colony algorithms in MANETs: a review, J. Netw. Comput. Appl. 35 (2012) 1964–1972.
[6] H. Al-Rawi, M. Ng, K. Yau, Application of reinforcement learning to routing in distributed wireless networks: a review, Artif. Intell. Rev. 43 (2013) 381–416.
[7] S. Chettibi, S. Chikhi, Adaptivity condition as the extended reinforcement learning for MANETs, J. Netw. Technol. 2 (2011) 122–129.
[8] B. Russell, M. Littman, W. Trappe, Integrating machine learning in ad hoc routing: a wireless adaptive routing protocol, Int. J. Commun. Syst. 24 (2011) 950–966.
[9] K. Yau, P. Komisarczuk, P. Teal, Reinforcement learning for context awareness and intelligence in wireless networks: review, new features and open issues, J. Netw. Comput. Appl. 35 (2012) 253–267.
[10] M. Marimuthu, A. Kannammal, A survey on fuzzy based QoS routing in mobile ad hoc networks, in: Proceedings of the IEEE International Conference on Intelligent Systems and Control (ISCO), 2013, pp. 376–379.
[11] G. Wang, L. Guo, H. Wang, H. Duan, L. Liu, J. Li, Incorporating mutation scheme into krill herd algorithm for global numerical optimization, Neural Comput. Appl. 24 (2012) 853–871.
[12] G. Wang, A. Gandomi, A. Alavi, G. Hao, Hybrid krill herd algorithm with differential evolution for global numerical optimization, Neural Comput. Appl. 25 (2013) 297–308.

[13] G. Wang, A. Hossein Gandomi, A. Hossein Alavi, A chaotic particle-swarm krill herd algorithm for global numerical optimization, Kybernetes 42 (2013) 962–978.
[14] G. Wang, L. Guo, A. Gandomi, G. Hao, H. Wang, Chaotic krill herd algorithm, Inf. Sci. 274 (2014) 17–34.
[15] G. Wang, A. Gandomi, A. Alavi, Stud krill herd algorithm, Neurocomputing 128 (2014) 363–370.
[16] G. Wang, A. Gandomi, A. Alavi, An effective krill herd algorithm with migration operator in biogeography-based optimization, Appl. Math. Model. 38 (2014) 2454–2462.
[17] G. Wang, A. Hossein Gandomi, X. Yang, A. Hossein Alavi, A novel improved accelerated particle swarm optimization algorithm for global numerical optimization, Eng. Comput. 31 (2014) 1198–1220.
[18] G. Wang, A. Gandomi, A. Alavi, S. Deb, A hybrid method based on krill herd and quantum-behaved particle swarm optimization, Neural Comput. Appl. (2015), http://dx.doi.org/10.1007/s00521-015-1914-z.
[19] G. Wang, A. Gandomi, X. Zhao, H. Chu, Hybridizing harmony search algorithm with cuckoo search for global numerical optimization, Soft Comput. (2014), http://dx.doi.org/10.1007/s00500-014-1502-7.
[20] G. Wang, S. Deb, A. Gandomi, Z. Zhang, A. Alavi, Chaotic cuckoo search, Soft Comput. (2015), http://dx.doi.org/10.1007/s00500-015-1726-1.
[21] S. Chettibi, S. Chikhi, Adaptive maximum-lifetime routing in mobile ad-hoc networks using temporal difference reinforcement learning, Evol. Syst. 5 (2013) 89–108.
[22] N. Chen, Q. Zhang, S. Jin, A fuzzy path selection power-based for MANET, in: B. Cao, T. Li, C. Zhang (Eds.), Fuzzy Information and Engineering, Springer, Berlin Heidelberg, 2009, pp. 1283–1291.
[23] W. El-Hajj, D. Kountanis, A. Al-Fuqaha, M. Guizani, A fuzzy based hierarchical energy efficient routing protocol for large scale mobile ad hoc networks, in: Proceedings of the IEEE International Conference on Communications, 2006, pp. 3585–3590.
[24] S. Abirami, R. Dhanasekaran, V. Bhanumathi, A balanced approach for power aware routing in MANET using fuzzy logic, in: Proceedings of the International Conference on Recent Trends in Computational Methods, Communication and Controls, 2012, pp. 334–345.
[25] P. Hiremath, S. Joshi, Energy efficient routing protocol with adaptive fuzzy threshold energy for MANETs, J. Comput. Netw. Wirel. Commun. 2 (2012) 402–407.
[26] S. Chettibi, S. Chikhi, FEA-OLSR: an adaptive energy aware routing protocol for MANETs using zero-order Sugeno fuzzy system, Int. J. Comput. Sci. Issues 10 (2013) 136–141.
[27] P. Nurmi, Reinforcement learning for routing in ad-hoc networks, in: Proceedings of the Fifth International Symposium on Modeling and Optimization in Mobile, Ad-Hoc, and Wireless Networks (WiOpt), 2007.
[28] W. Naruephiphat, W. Usaha, Balancing tradeoffs for energy efficient routing in MANETs based on reinforcement learning, in: Proceedings of the 67th IEEE Vehicular Technology Conference, 2008, pp. 2361–2365.
[29] L. Zadeh, Fuzzy sets, Inf. Control 8 (1965) 338–353.
[30] E. Mamdani, Application of fuzzy logic to approximate reasoning using linguistic synthesis, IEEE Trans. Comput. C-26 (1977) 1182–1191.
[31] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE Trans. Syst. Man Cybern. 5 (1985) 116–132.
[32] R. Sutton, A. Barto, Reinforcement Learning, MIT Press, Cambridge, MA, 1998.
[33] C. Perkins, E. Royer, Ad hoc on-demand distance vector routing, in: Proceedings of the Second IEEE Workshop on Mobile Computing Systems and Applications, 1999, pp. 90–100.
[34] N. Vassileva, F. Arroyo, A survey of routing protocols for maximizing the lifetime of ad hoc wireless networks, J. Softw. Eng. Appl. 2 (2008) 77–90.
[35] D. Kim, J. Garcia-Luna-Aceves, K. Obraczka, J. Cano, P. Manzoni, Power aware routing based on the energy drain rate for mobile ad hoc networks, in: Proceedings of the 11th International Conference on Computer Communications and Networks (ICCCN), 2002, pp. 565–569.
[36] V. Derhami, V. Majd, M. Ahmadabadi, Fuzzy Sarsa learning and the proof of existence of its stationary points, Asian J. Control 10 (2008) 535–549.
[37] K. Fall, K. Varadhan, The NS Manual (Formerly NS Notes and Documentation), 2010, Available at: http://www.isi.edu/nsnam/ns/ns-documentation.html.
[38] R. Steuer, Multiple Criteria Optimization: Theory, Computation and Applications, Wiley, Chichester, 1986.