Computer Networks 54 (2010) 1692–1706

Contents lists available at ScienceDirect

Computer Networks

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

Intelligent QoS management for multimedia services support in wireless mobile ad hoc networks

Lyes Khoukhi [a,*], Soumaya Cherkaoui [b]

[a] Autonomic Networking Environment, ICD/ERA, CNRS UMR-STMR 6279, University of Technology of Troyes, 10000, France
[b] Department of Electrical and Computer Engineering, University of Sherbrooke, Sherbrooke, J1K 2R1, QC, Canada

ARTICLE INFO

Article history:
Received 15 May 2009
Received in revised form 3 November 2009
Accepted 22 January 2010
Available online 4 February 2010
Responsible Editor: Qian Zhang

Keywords:
Wireless mobile ad hoc networks
Fuzzy logic system
Quality of service
Fuzzy Petri nets
Congestion control

ABSTRACT

In this paper, we propose a new intelligent cross-layer QoS support for wireless mobile ad hoc networks. The solution, named FuzzyQoS, exploits fuzzy logic for improving traffic regulation and the control of congestion to support both real-time multimedia (audio/video) services and non-real-time traffic services. FuzzyQoS includes three contributions: (1) a fuzzy logic approach for best-effort traffic regulation (FuzzyQoS-1), (2) a new fuzzy Petri nets technique (FuzzyQoS-2) for modeling and analyzing the QoS decision making for traffic regulation control, and (3) a fuzzy logic approach for threshold buffer management (FuzzyQoS-3). In FuzzyQoS-1, the feedback delay information received from the network is used to perform a fuzzy regulation for best-effort traffic. Using fuzzy logic, FuzzyQoS-3 uses fuzzy thresholds to adapt to the dynamic conditions. The evaluation of FuzzyQoS performances was studied under different mobility, channel, and traffic conditions. The results of simulations confirm that a cross layer design using fuzzy logic at different levels can achieve low and stable end-to-end delay, and high throughput under different network conditions. These results will benefit delay- and jitter-sensitive real-time services.

1. Introduction

Wireless ad hoc networks are a new technology in the evolution of wireless communications. In this technology, wireless devices can communicate with each other in the absence of a fixed infrastructure. Ad hoc networks usually consist of a set of nodes that communicate over wireless links in a multihop manner without a need for a central control, which creates a high level of flexibility to users.

With the widespread use of wireless technology, the ability of mobile wireless ad hoc networks to support real-time services with Quality of Service (QoS) has become a challenging research subject. The notion of QoS satisfaction is defined as the guarantee by the network to satisfy some predetermined service constraints for users in terms of end-to-end delay, available bandwidth, etc. However, the QoS issue is yet a challenging task in ad hoc networks because there is no fixed infrastructure and the topology is frequently changing due to nodes mobility. Furthermore, links are constantly established and broken. The availability and quality of a link fluctuates due to channel fading and interference from other transmitting devices. Various approaches and protocols have been proposed to address QoS ad hoc networking problem [1–12]. Multiple efforts are also still under way within academic and industrial research projects.

SWAN [1], INSINGIA [2], and FQMM [3] are some noteworthy QoS models attempting to establish comprehensive QoS solutions for MANETs. SWAN proposes service differentiation in stateless wireless ad hoc networks using distributed control algorithms and a rate control system at each node. However, one of the drawbacks of SWAN is how to calculate the threshold rate limiting any excessive delay that might be experienced [4]. It also uses merely two levels of services: real-time and best-effort traffic. SWAN and INSIGNIA are intranet QoS models providing services that

* Corresponding author.
E-mail addresses: Lyes.Khoukhi@USherbrooke.ca (L. Khoukhi), Soumaya.Cherkaoui@USherbrooke.ca (S. Cherkaoui).

1389-1286/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.comnet.2010.01.014

have to be mapped to either per-flow or per-class services, but SWAN remains the best example of stateless distributed QoS framework developed for wireless ad hoc networks. INSIGNIA is on the other hand, a QoS framework with per-flow granularity and reasonable treatment for mobility. The main goal of INSIGNIA is to provide adaptive QoS guarantees for real-time traffic. It employs an in-band signaling system that supports fast reservation, restoration, and adaptation algorithms. Three levels of services are implemented: best-effort, minimum, and maximum. The bandwidth is the only QoS parameter used in INSIGNIA. FQMM is another approach combining the advantages of per-class granularity of DiffServ with the per-flow granularity of IntServ. It tries to preserve the per-flow granularity for a small portion of traffic in MANETs, given that a large amount of the traffic belongs to per aggregate of flows, that is, per-class granularity. FQMM offers a good solution for small and medium-size ad hoc network, but it is not suitable for large networks.

Recently, some intelligent methods have been applied in the area of ad hoc networks, aiming to obtain more adaptive and flexible models over the existing models. In [13], the authors developed a SHROT model, which is a dynamic source routing protocol using a self healing and optimized routing techniques based on fuzzy logic concepts. The basic idea of this model is the modification of the entries of the neighbour table and the time-stamp of each entry based on fuzzy system. The model AntNet proposed in [14] is an adaptive algorithm based on mobile-agents. This algorithm is inspired by work on the ant colony metaphor. In AntNet, each node periodically launches network exploration agents called forward ants to every destination. At each node, the ants will choose their next hop using that nodes routing table. As the ants visit a node, they record their arrival time and the node identity in a stack [14]. In [15], the authors describe a policy-based management system for improving the flexibility of wireless mobile ad hoc network. This system provides the capability to express networking requirements in the form of policies at a high level and have them automatically applied in the network by intelligent agents [15]. The model proposed in [16] investigates the use of fuzzy logic theory for assisting the TCP error detection mechanism in ad hoc networks. An elementary fuzzy logic engine was presented as an intelligent technique for discriminating packet loss due to congestion from packet loss by wireless induced errors.

In [6], we have proposed an intelligent QoS model with service differentiation based on neural networks for mobile ad hoc networks named GQOS. GQOS is composed of a kernel plan which ensures basic functions of routing and QoS support control, and an intelligent learning plan which ensures the training of GQOS kernel operations by using a multilayered feedforward neural network (MFNN). The objective of using neural networks was the fast learning of the different operations performed by the kernel and the reduction of the processing time in the network. However, GQOS is not scalable in terms of end-to-end delay under higher network mobility and traffic load.

We have explored in [7] the usage of a fuzzy logic semi-stateless approach for service differentiation in wireless ad hoc networks. The proposed model named FuzzyMARS includes a set of mechanisms: admission control for real-time traffic, a fuzzy logic system for best-effort traffic regulation, and three schemes for real-time traffic regulation. FuzzyMARS architecture support real-time UDP traffic as well as TCP traffic. The resulted simulations have shown the benefits of using a fuzzy logic semi-stateless model; the average delay obtained is quite stable and low under different network conditions. Nevertheless, in FuzzyMARS we considered neither buffer management nor QoS decision making for traffic regulation.

In this paper, we explore an integrated new intelligent cross-layer QoS solution based on fuzzy logic for wireless mobile ad hoc networks. This choice is justified by the fact that fuzzy logic is well adapted to systems characterized by imprecise states, as in the case of ad hoc networks. The proposed approach, FuzzyQoS, aims to improve the control of traffic regulation rate and congestion control of multimedia applications. FuzzyQoS integrates three mechanisms at different layers: a fuzzy logic approach for best-effort traffic regulation (FuzzyQoS-1), QoS decision making for traffic regulation (FuzzyQoS-2), and a fuzzy logic approach for threshold buffer management (FuzzyQoS-3). The delay feedback information received from the network is the key parameter used in FuzzyQoS-1 and FuzzyQoS-2, to ensure that best-effort traffic coexists well with real-time traffic in the multimedia applications. The feedback measurement represents the packet delay measured by the IEEE 802.11 MAC which is integrated as a part of the FuzzyQoS architecture. The objective of the FuzzyQoS-1 is to dynamically adjust the transmission of traffic according to the network conditions.

FuzzyQoS-2 is a fuzzy Petri nets technique for modeling and analyzing the QoS decision making for traffic regulation in wireless ad hoc networks. FuzzyQoS-2 exploits fuzzy concepts to model the QoS decision making by the source nodes. The fuzzy Petri nets tool is used for its efficiency and flexibility over other modeling tools (such as Petri nets) with the objective of better modeling and representing the process of traffic regulation.

Finally, FuzzyQoS-3 uses fuzzy thresholds to adapt to the dynamic conditions of the network. The notion of threshold is practical for discarding data packets and adapting the traffic service depending on the occupancy of buffers. The threshold function has a significant influence on the performance of networks in terms of both packets average delay and throughput. Therefore, the selection of a particular threshold may be decisive to the control of congestion. This selection in the proposed FuzzyQoS model is based on fuzzy logic.

We studied FuzzyQoS performances under different network conditions in terms of mobility and scalability. The results of simulations, shown in Section 3, confirm that FuzzyQoS promises to be an efficient QoS solution in terms of the average delay and throughput to support both real-time and non-real-time multimedia services.

The objective of the integration of these three mechanisms is to find a good balance between network performances (by improving the end-to-end delay parameter using FuzzyQoS-1 and FuzzyQoS-2 mechanisms) and reliability (by improving the throughout parameter using FuzzyQoS-3).

This paper is organized as follows: in Section 2, we describe in detail the proposed FuzzyQoS architecture. Section 3 shows the performance evaluation and simulation results of FuzzyQoS under different network mobility, traffic, and channel conditions. Finally, we conclude the paper in Section 4.

2. FuzzyQoS architecture

2.1. Overview of the FuzzyQoS architecture

Fig. 1 illustrates the FuzzyQoS architecture. As presented in the schematic diagram, the architecture aims to support QoS and to adapt to the dynamic changes of the environment. This is achieved by the cooperation between a set of functionalities and mechanisms integrating fuzzy logic at different layers that work jointly. FuzzyQoS was built over our previous work FuzzyMARS [7] which already contained some functions such as a routing scheme, admission controller, and classifier.

The routing scheme and the temporary resource reservation process perform the discovery of routes and bandwidth reservation. The admission controller efficiently estimates the local available bandwidth at each node. The decision to admit a new flow is done by the admission control mechanism. The classifier is able to differentiate between flows in terms of QoS requirements best-effort flows and real-time flows, in order to delay the best-effort packets. Note that even if the admission control is performed to guarantee enough available bandwidth before accepting a new flow, a congestion may occur in the network because of the nodes mobility. Therefore, it is of outmost importance to assure traffic regulation. The classified best-effort packets are regulated using a fuzzy logic system according to the application requirements and the network state. The fuzzy logic system we developed will be explained throughout the rest of the paper. The fuzzy best-effort regulation uses the feedback delay received from the network. The fuzzy regulation process is performed in three steps: fuzzification, inference rules evaluation, and defuzzification. In FuzzyQoS model, like in SWAN, it is not necessary to have a QoS-capable MAC to deliver service differentiation. Rather, real-time services are built using existing best-effort IEEE 802.11 MAC technology. Later, in the paper, after giving an overview regarding fuzzy logic concepts, we will give more details about the proposed fuzzy logic cross-layer solution.

2.2. Overview of fuzzy logic

Fuzzy logic theory [17–21] was first introduced by Zadeh in the 1960s as a tool for modeling the uncertainty of natural language. It has been commonly employed for supporting intelligent systems. This technology has proven efficiency in various applications such as decision support and intelligent control, especially where a system is difficult to be characterized. A fuzzy logic system basically consists of considers basically three steps: fuzzification, rules

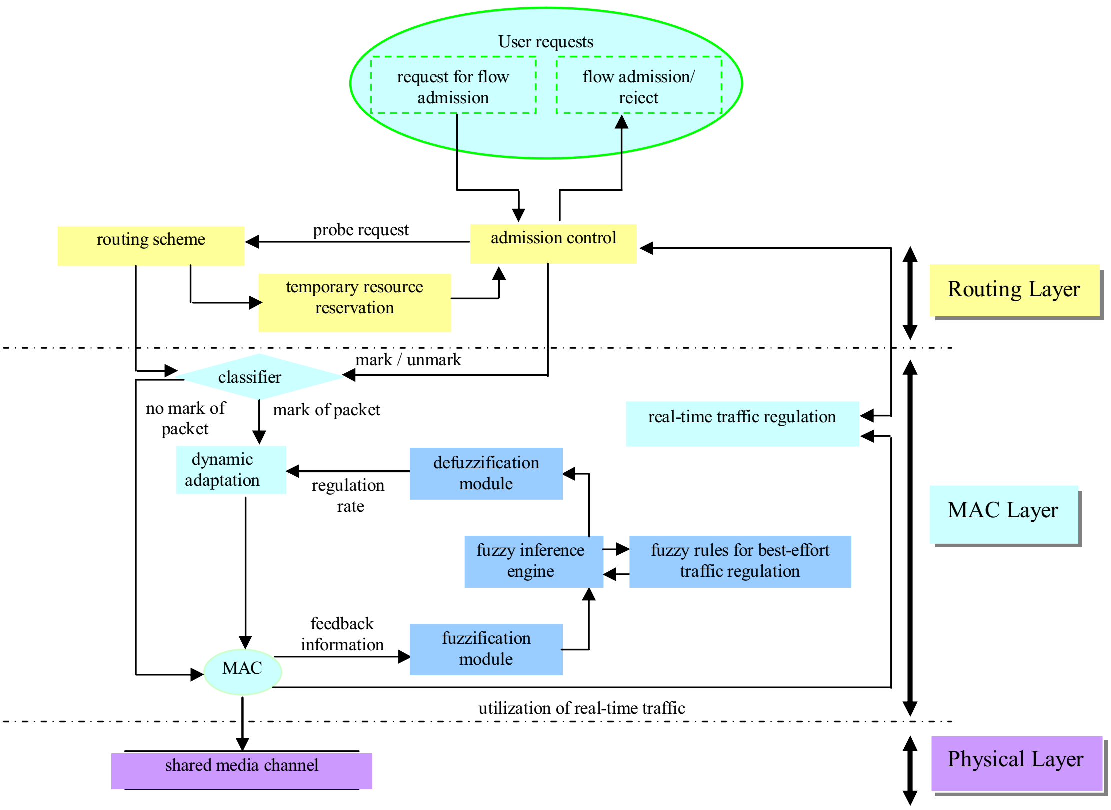


Fig. 1. A schematic diagram of FuzzyQoS model.

evaluation, and defuzzification. The first step is responsible for mapping discrete (also called crisp) input data into proper values in the fuzzy logic space. For that end, membership functions (fuzzy sets) are used to provide smooth transitions from false to true (0 to 1). The second step performs reasoning on the input data by following predefined fuzzy rules. Once the input data are processed by fuzzy reasoning, the defuzzification takes the task of converting back these input data into crisp values.

There are two main characteristics of fuzzy systems that give them better performance for specific applications [18,19]:

- Fuzzy systems are suitable for uncertain or approximate reasoning, especially for the system with a mathematical model that is difficult to derive.
- Fuzzy logic allows decision making with estimated values under incomplete or uncertain information.

The importance of fuzzy logic derives from the fact that most manners of human reasoning and especially common sense reasoning are approximate in nature. Bellman and Zadeh write: "Much of the decision making in the real world takes place in an environment in which the goals, the constraints and the consequences of possible actions are not known precisely". This "imprecision" represents the core of fuzzy logic and fuzzy sets theory. This later was proposed as a generalization of classical set theory [19,30,31]. Readers are referred to ([18,26,32]) for more detail about fuzzy logic reasoning.

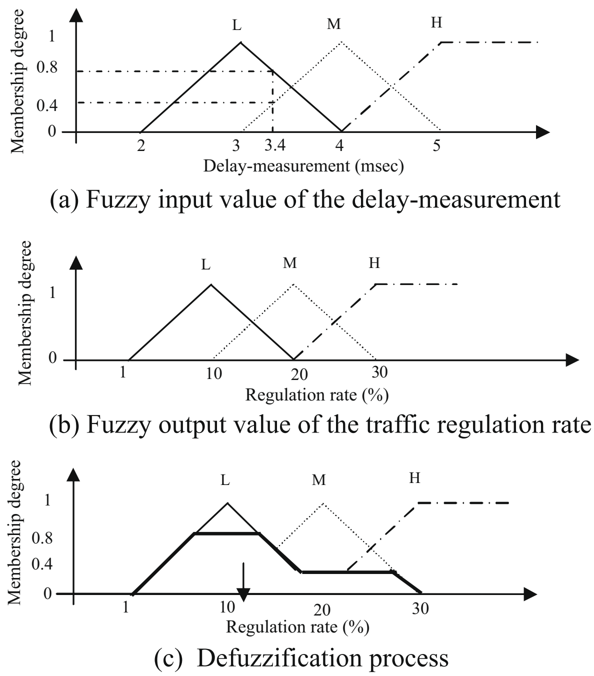
2.3. Fuzzy logic approach for best-effort traffic regulation (FuzzyQoS-1)

FuzzyQoS-1 uses fuzzy logic to perform the regulation of best-effort traffic. The use of fuzzy logic can add more flexibility and capability for operating with imprecise information due to nodes mobility in wireless ad hoc network. The feedback delay received from the network is the key input parameter of FuzzyQoS-1. FuzzyQoS-1 ensures that best-effort traffic will coexist well with real-time traffic.

In the proposed fuzzy regulation approach, the feedback measurement represents the packet delay measured by the IEEE 802.11 MAC. The measure of the packet delay is performed as follows: at the reception of a packet by the MAC layer, the later listens to the channel and differs the access to the channel according to the CSMA/CA algorithm. When the MAC gets access to the channel, then RTS-CTS-DATA-ACK packets are exchanged. The reception of ACK packet by the transmitter means that the packet was successfully received by the receiver. The time taken to send the packet between transmitter and receiver including the total differed time represent the packet delay. This delay represents the difference between the time that a packet is passed to the MAC layer (from the upper layer), and the time of reception of ACK packet from the receiver. The received packet delay can reflect the network state: a high delay means that a possible situation of congestion has occurred in the network. Thus, when one or more packets have a bigger delay than a certain value, the traffic regulation process is triggered in order to reduce the traffic.

As mentioned earlier, the regulation control of the best-effort traffic is performed as a response to the delay feedback by means of a fuzzy logic system. Three steps are considered by FuzzyQoS-1: in the first step (i.e. fuzzification), the delay-measurements are transformed into fuzzy sets; then in the second step (i.e. rules evaluation), fuzzy rules are applied into the fuzzy input in order to compute the fuzzy outputs. The third step (i.e. defuzzification) translates the fuzzy outputs into crisp values. In the following, we give more details about these steps:

(1) *Fuzzification*: the delay-measurement obtained as feedback from MAC layer represents the fuzzy input parameter of FuzzyQoS-1. The fuzzy output parameter is represented by the traffic regulation rate. These two parameters have to be converted into fuzzy sets. Note that a fuzzy set may contain elements that have different degrees of membership in various sets, whereas, an ordinary set a element should have full membership in the set in order to be considered a set member. If the delay-measurement parameter is considered in an ordinary set, then it can only be either low or high, not both simultaneously. However, the delay-measurement in a fuzzy set can be classified as: not high, medium, or quite low. Thus, the membership of an element may be not the same over various fuzzy sets. The membership function represented as a line or curve indicates how to map each input (i.e. delay-measurement) or output (i.e. regulation rate) parameters in order to obtain their membership values. The threshold values of the fuzzy sets are presented in



(a) Fuzzy input value of the delay-measurement

(b) Fuzzy output value of the traffic regulation rate

(c) Defuzzification process

Fig. 2. Fuzzification and defuzzification process.

Fig. 2. We represented the following sets: low (L), medium (M), and high (H). Let us consider that the threshold for low delay-measurement is 3 ms, for medium delay-measurement it is 4 ms, and for high delay-measurement it is 5 ms. Then, by mapping the current delay-measurement onto the graph of the membership function, the delay will be allocated with a membership value in each set between 0 and 1. For example, in Fig. 2.a if the current delay-measurement is 3.4 ms, then this value can be fuzzified into low delay with the degree of 0.8, medium delay with the degree of 0.4 and high delay with the degree of 0. Fig. 2.a and b illustrate respectively, the process of fuzzification of input parameters delay-measurement and traffic regulation rate.

(2) *Rules evaluation:* the fuzzy rules are presented as a set of rules "*if* (...) *then* (...)". Throughout rules evaluation process, these fuzzy rules are applied over fuzzy sets using an inference engine. The inference control of the traffic regulation rate is illustrated by the following rules:

If the delay-measurement is increased, then reduce the traffic rate

If the delay-measurement is decreased, then raise the traffic rate

As the delay-measurement becomes high, this means that data packets take more time to be received by the destination node, which means that a possible congestion has occurred. Consequently, a decrease in the traffic rate should be performed to reduce the congestion level in the network. Thus, the decision making logic of the traffic regulation rate follows the delay-measurement parameter as will be explained in the next section.

(3) *Defuzzification*: in this phase, the resulted fuzzy decision sets will be converted into crisp values. We have chosen the method of mean of maxima (MoM) [13] to perform the defuzzification because of its light computational complexity. The evaluation result is obtained as the average of the elements that reach the maximum grade in a fuzzy set.

Now that we presented the fuzzy logic for traffic regulation mechanism, let us detail how to model and analyse the QoS making decision for traffic regulation using a fuzzy Petri nets tool.

2.4. Fuzzy Petri nets model for traffic regulation (FuzzyQoS-2)

FuzzyQoS-2 describes the regulation process performed by the source node; it represents the different fuzzy steps for decision making by formulating the different rules of this process taking into account both the delay measurement and nodes mobility parameters. The proposed model studies the fuzzy regulation traffic rules in order to deal with the imprecise information caused by the dynamic topology of ad hoc networks. The representation of different fuzzy processes for decision making can be performed by formulating the production rules of these processes. Each fuzzy production rule is a set of antecedent input conditions and consequent output propositions. We proceed to construct the previous aspects (the input and output parameters) of the production rules in order to better represent and understand the process of traffic regulation in wireless ad hoc networks. The traffic regulation used to avoid the congestion depends on the traffic state and the dynamic topology of the network. These constraints represent the input parameters of FuzzyQoS-2. The traffic state is represented by the delay-measurement. The delay measurement parameter can give information about the status of a network in terms of congestion. A big value of this parameter means that a congestion case has may appeared in the network. Therefore, the process of traffic regulation should be started. The second input parameter in FuzzyQoS-2 is node mobility. The output parameter is the regulation rate. The choice of using a fuzzy Petri nets tool has been motivated by its efficiency and flexibility over other modeling tools for better representing the process of traffic regulation.

2.4.1. Overview of fuzzy Petri nets

It is observed that classical Petri Nets [16] do not have sufficient capacity to model the uncertainty in systems [20]. This limitation of Petri nets has encouraged researchers to extend the exiting models by using the fuzzy reasoning theory [21,22]. The combination of Petri nets models and fuzzy theory has given rise to a new modeling tool called Fuzzy Petri Nets (FPN). FPN formalism has been widely applied in several applications such as, robotics systems [23], real-time control system [20], fuzzy reasoning systems [25], etc.

In the following, we give a brief description about the FPN modeling tool [22,24].

Let consider FPN = (PN, CND, MF, FSR, FM).

(a) The tuple PN = $(P,T,A,\mathrm{FW},\mathrm{FH})$ is called Petri nets if: (P,T,A) is a finite net, where [16]: $P = \{P_1, P_2, \ldots, P_n\}$ is a finite non-empty set of places, $T = \{T_1, T_2, \ldots, T_n\}$ is a finite non-empty set of transitions,
$A \subseteq (P \times T) \cup (T \times P)$ is a finite set of arcs between the places and transitions or vice versa.
FW: $A \to N^+$ represents a weighting function that associates with each arc a non-negative integer of N^+.
FH $\subset (P \times T)$: represents an inhibition function that associates a place $P_i \in P$ contained in FH (T_j) to a transition T_j itself.

(b) $\mathrm{CND} = \{\mathrm{cd}_1, \mathrm{cd}_2, \ldots, \mathrm{cd}n\}$ represents a set of conditions that will be mapped into the set P; each $\mathrm{cd}_i \in \mathrm{CND}$ is considered as one input to the place $P_i \in P$. A condition cd_i takes the form of "X is Z", which means a combination between the fuzzy set Z and the attribute X of the condition. For instance, in the condition "the delay measurement is small", the attribute "X = delay measurement" is associated to the fuzzy set "Z = small", but other fuzzy sets can also be considered (e.g. "Z = medium", "Z = large", etc.).

(c) Consider MF: $u_Z(x) \to T$, a membership function which maps the elements of X (as defined in b.) into the values of the range [0,1]. These values represent the membership degree in the fuzzy set Z. The element x belonging to X represents the input parameter of the condition "X is Z", and $u_z(x)$ measures the degree of truth of this condition. Note that the composition of membership function degrees of the required conditions is performed by fuzzy operators such as MIN/MAX.

(d) Let consider the following rule R_i:
R_i: ifx_1 isz_1 and /or x_2 is z_2 then A is B
The firing strength function of rule R_i (FSR$_i$) represents the strength of belief in R_i. The conclusion of R_i (modeled by CSR$_i$) can take one of the following forms:

$$\mathrm{CSR}_i = \mathrm{MIN}(u_{z1}(x_1), u_{z2}(x_2)) = u_{z1}(x_1) \wedge u_{z2}(x_2),$$
$$\mathrm{CSR}_i = \mathrm{MAX}(u_{z1}(x_1), u_{z2}(x_2)) = u_{z1}(x_1) \vee u_{z2}(x_2).$$

(e) SWR is the selected wining rule R_L among the n-rules R_1, $R_2,\ldots,$ R_n. SWR is the rule which has the highest degree of truth. Let FSR$_L$ be the corresponding firing strength of R_L, then the selected rule SWR is given as follows: SWR=MAX$(\mathrm{FSR}_1, \mathrm{FSR}_2, \ldots, \mathrm{FSR}_n)$

(f) The marking task in FPN illustrates the satisfaction of events occurred during the performance of fuzzy rules. This marking function called "fuzzy marking" (FM) distributes the tokens over the places of the nets.

The sequence $\delta = \langle T_1, T_2, \ldots, T_n\rangle$ is said to be reachable from a fuzzy marking FM$_1$, if $T_i \in T$is a firable from FM$_{i-1} \in$ FM and leads to FM$_{i+1} \in$ FM, for all transitions $T_i \in \delta$. The firing of transition $T_i \in T$ (Fig. 3) is performed in two steps: (a) T_i removes tokens and then, (b) T_i places tokens.

2.4.2. Fuzzy regulation traffic rules usage

According to [26], most of fuzzy systems use the following form for modeling:

Rule R: *if* Ip$_1$ *is* A AND Ip$_2$ *is*B then Op is C
Where:

- Ip$_1$ and Ip$_2$ are the input parameters,
- Op is an output parameter,
- A, B, and C are fuzzy sets,
- AND represent fuzzy operator,
- The fuzzy conditions of rule R are "Ip$_1$ is A", and "Ip$_2$ is B".

The construction of the above aspects (inputs, outputs, and fuzzy sets) for performing the traffic regulation depends on the traffic state and the dynamic topology of wireless ad hoc networks. Thus, the previous fuzzy aspects can take various values:

- The first input parameter is represented by the Delay-Measurement (DM) at a mobile node. DM can be either "small" or "large".
- The second input parameter is represented by the Node Mobility (NM). NM can either be "slow" or "medium" (note that "fast node mobility" is included in the case of "medium node mobility").
- The output parameter is represented by the Traffic regulation rate (TR). TR can either be "decreased" (slowly or largely) or "increased" (slowly or largely).

Let consider the following fuzzy rule R_L as example:
R_L: *if* DM *is small and* NM *is slow, then* TR *is increased largely*

R_L takes into consideration the input parameter of the feedback delay-measurement DM and the node mobility NM in wireless ad hoc networks. The traffic regulation rate TR represents the output parameter.

FPN that models the dynamic aspect of the fuzzy rule R_L is illustrated in Fig. 4.

- P_{acd1}: models the antecedent condition 1 (acd$_1$) of R_L; acd$_1$ = "DM is small".
- P_{acd2}: models the antecedent condition 2 (acd$_2$) of R_L; acd$_2$ = "NM is slow".
- T_{aMf1}: models the membership function of the antecedent condition 1; $T_{\mathrm{aMf1}=}u_{small}$(DM).
- T_{aMf2}: models the membership function of the antecedent condition 2; $T_{\mathrm{aMf2}=}u_{slow}$(NM).
- P_{aMd1}: models the membership degree value of the condition 1 of R_L. This value determines the satisfaction degree of the DM input parameter to the fuzzy set "small".
- P_{aMD2}: models the membership degree value of the condition 2 of R_L. This value determines the satisfaction degree of the NM input parameter to the fuzzy set "slow".
- T_{FSCL}: models the operation of minimum composition "MIN" between the antecedent conditions (e.g. condition 1 and condition 2) of R_L. The firing strength of R_L is represented by the MIN operation: $\mathrm{MIN}(u_{small}(\mathrm{DM}), u_{slow}(\mathrm{NM}))$.
- P_{FSCL}: models the value of the firing strength of R_L. This value defines the degree of truth of the output proposition "TR is increased largely".

2.4.3. Fuzzy Petri nets model for traffic regulation

FuzzyQoS-2 considers the following rules:

R_1: *if* DM *is small and* NM*is slow then*TR *is increased largely*,

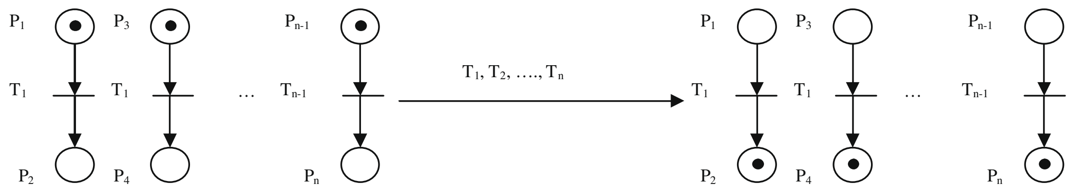


Fig. 3. The transitions firing in FPN.

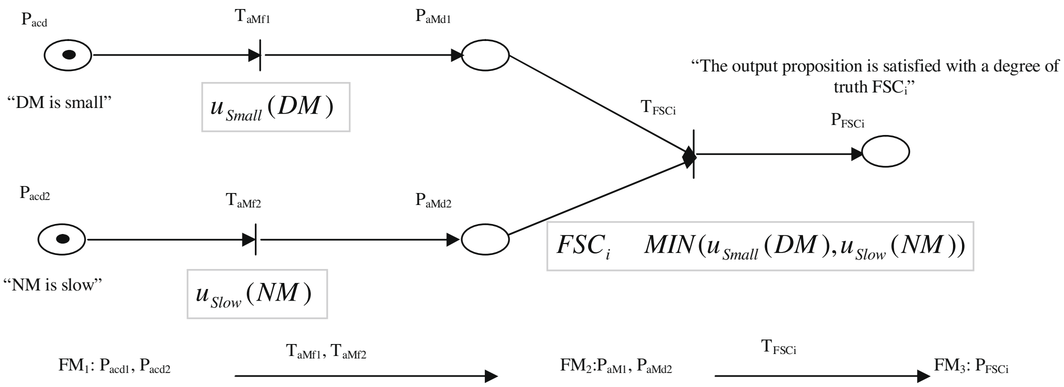
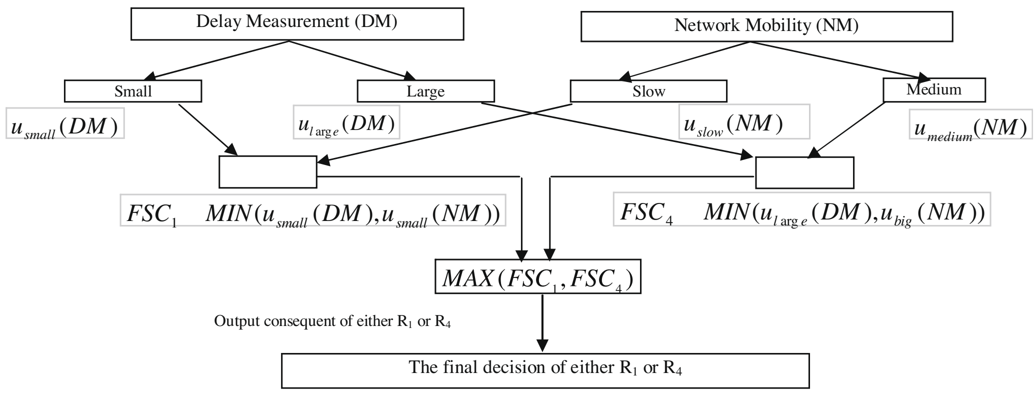


Fig. 4. The modeling of fuzzy rules structure and its dynamic behaviour.

R_2: *if* DM *is small and* NM*is medium then* TR*is increased*,
R_3: *if* DM *is large and* NM *is slow then* TR *is decreased*,
R_4: *if* DM*is large and* NM *is medium then* TR *is decreased largely*.

- Input parameters:
  - The input parameter of the first antecedent condition of R1, R2, R3, and R4 is the delay measurement "DM".
  - The input parameter of the second antecedent condition of R_1,R_2,R_3, and R_4 is the node mobility "NM".
- Fuzzy sets: The fuzzy set of the antecedent conditions of R_1, R_2, R_3, and R_4 are: small, large, slow, and medium.
- Antecedent conditions (acd$_i$):
  - The first antecedent conditions in R_1, R_2, R_3, and R_4 are: acd$_1$: "DM is small", acd$_2$: "DM is large".
  - The second antecedent conditions in R_1, R_2, R_3, and R_4 are: acd$_3$: "NM is slow", acd$_4$: "NM is medium".
- Output parameters: The output parameter of R_1, R_2, R_3, and R_4 is the traffic regulation rate "TR".
- The rules R_1, R_2, R_3, and R_4 use the following decisions making: "increased largely", "increased", "decreased", and "decreased largely"
- The fuzzy logic operator used by the rules R_1, R_2, R_3, and R_4 is t AND

The fuzzy operator "AND" is used in order to combine the two antecedent conditions of each rule using the MIN function. This provides the firing strength value for each rule. After that, MAX composition function is used to combine all firing strength values of the defined rules R_1, R_2, R_3, and R_4 in the aim of determining the highest one that will



Fuzzy Decision making algorithm:

- Phase 1: enter the input parameters of the rules R_1, R_2, R_3, and R_4.
- Phase 2: calculate the degree of truth of the antecedent conditions.
- Phase 3: apply the operation of minimum composition (MIN) with the fuzzy operator AND/OR in order to generate the firing strength value for each rule R_1, R_2, R_3, and R_4.
- Phase 4: apply the operation of maximum composition to select the wining rule among the rules R_1, R_2, R_3, and R_4.
- Phase 5: generate the output consequent of the selected wining rule.

Fig. 5. Fuzzy decision making algorithm.

be the selected wining rule. Fig. 5 shows the fuzzy logic scheme for decision making of R_1, R_2, R_3, and R_4.

In what follows, we illustrate the steps of the proposed FPN model.

(a) Enter the input parameters into the places and transitions:
  – $P_{IP} = \{P_{IP1}, P_{IP2}, \ldots, P_{IPn}\}$ is a set of places that represent the input parameters. In the Fig. 6, the places used are P_1 and P_2 which represent respectively, the first (e.g. delay measurement DM) and second (e.g. node mobility NM) antecedent condition of R_1, R_2, R_3, and R_4.
  – $T_{IP} = \{T_{IP1}, T_{IP2}, \ldots, T_{IPn}\}$ represents a set of input parameter transitions. The transitions T_{IP1} and T_{IP2} illustrated in Fig. 6 are used to distribute, the input parameters "DM" and "NM" respectively for making the first and second antecedent conditions of the defined rules R_1, R_2, R_3, and R_4.

(b) Represent the antecedent conditions and compute the membership function for each condition.
  – $P_{acd} = \{P_{acd1}, P_{acd2}, \ldots, P_{acd}\}$ is a set of places that represent the antecedent conditions. P_{acd1} and P_{acd2} in the model presented in Fig. 6 describe the antecedent conditions "acd$_1$" and "acd$_2$", respectively.
  – $T_{aMf} = \{T_{aMf1}, T_{aMf2}, \ldots, T_{aMfn}\}$ a set of transitions that represent the antecedent membership functions. T_{aMf1}, T_{aMf2}, T_{aMf3}, T_{aMf4} observed in Fig. 6 represent the membership functions of , $u_{small}(\mathrm{DM})$, $u_{large}(\mathrm{DM})$, $u_{slow}(\mathrm{NM})$, $u_{medium}(\mathrm{NM})$respectively.
  – $P_{aMd} = \{P_{aMd1}, P_{aMd2}, \ldots, P_{aMdn}\}$ is a set of places that represent the antecedent membership degrees. The values of the place P_{aMd1} indicates the degree of satisfaction of the input parameter DM to the fuzzy set "small".

(c) Compute the firing strength of conditions
  – $T_{FSC} = \{T_{FSC1}, T_{FSC2}, \ldots, T_{FSCn}\}$ represent a set of transitions that model firing strength conditions. For instance, the transition T_{FSC1} shown in Fig. 6 performs the operation of minimum composition (MIN) on the antecedent conditions of the rule R_1:MIN$(u_{small}(\mathrm{MD}), u_{slow}(\mathrm{NM}))$. Note that the fuzzy operator AND is integrated with the MIN operation to combine the first and second conditions of R_1.
  – $P_{FSC} = \{P_{FSC1}, P_{FSC2}, \ldots, P_{FSCn}\}$ is a set of places that represent the firing strength. P_{FSCi} tokens are proportional to the number of antecedent conditions of a rule R_i. This number is shown by the label illustrated between the transitions T_{aMfi} and the

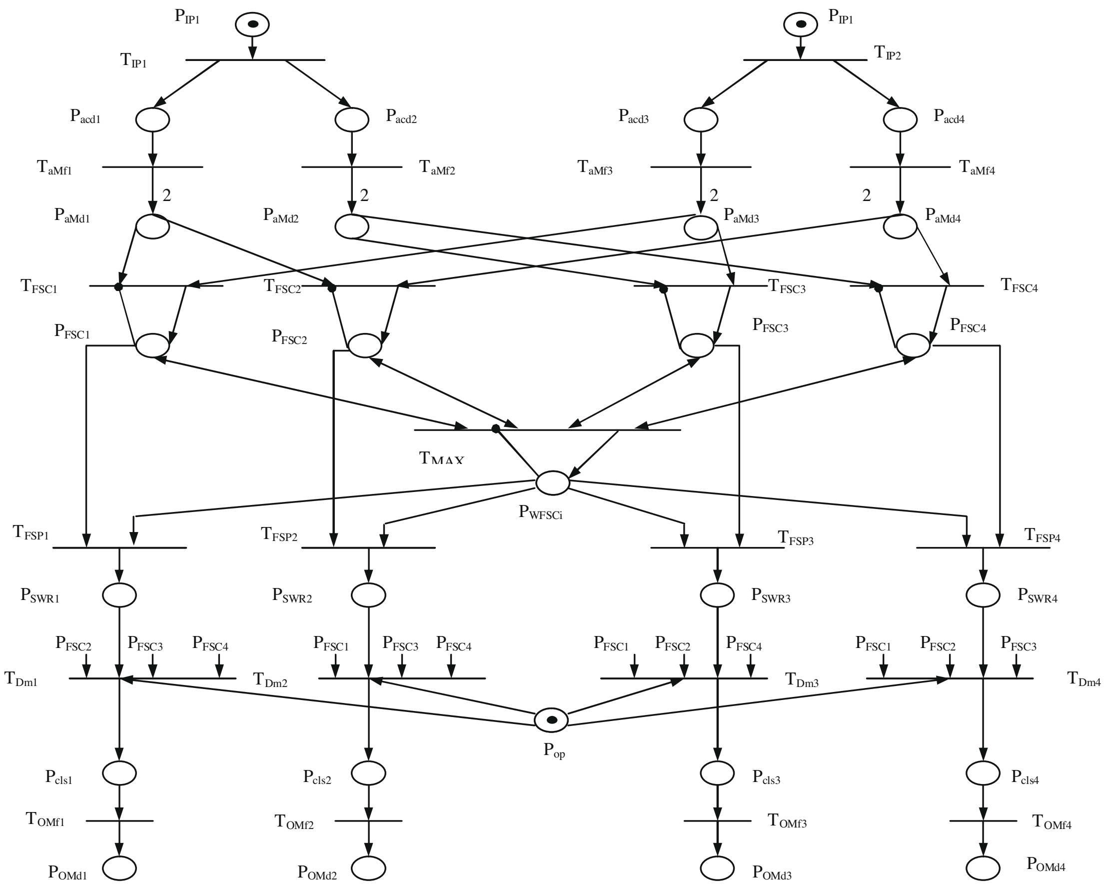


Fig. 6. Fuzzy Petri nets for traffic regulation process.

place P_{aMdi}. The construction of the antecedent conditions of a rule R_i is performed by firing a transition T_{FSCi}. The inhibitor arc designed between a place P_{FSCi} and T_{FSCi} is useful to note that T_{FSCi} should fire one time.

(d) Determine the selected winning rule among the activated rules to make the final decision making on the traffic regulation
  - $T_{\text{FMAX}} = \text{MAX}\{P_{\text{FSC1}}, P_{\text{FSC2}}, \ldots, P_{\text{FSC}n}\}$ is a transition that models the maximum composition operation (MAX) for the defined rules. The firing strength value of a rule R_i is stored in the place P_{FSCi}.
  - P_{WFSCi} represents the firing strength condition FSC_i of the selected wining rule R_i. The later rule is determined as in the following step.
  - $T_{\text{FSP}} = \{T_{\text{FSP1}}, T_{\text{FSP2}}, \ldots, T_{\text{FSP}n}\}$ is a set of transitions that model the firing strength comparison. For instance, the transition T_{FSP3} is useful to make a comparison between FSC_3 of the rule R_3 and the selected wining firing strength WFSC_{i}.
  - $P_{\text{SWR}} = \{P_{\text{SWR1}}, P_{\text{ SWR2}}, \ldots, P_{\text{SWR}n}\}$ is a set of places that models the selected wining rules. The rule R_i is selected to be fired if the place P_{SWRi} contains a token.

(e) The conclusion of the selected rules:
  - $T_{\text{Dm}} = \{T_{\text{Dm1}}, T_{\text{Dm2}}, \ldots, T_{\text{Dm}n}\}$ is a set of transitions that represent the decision of the selected rule. T_{Dmi} deletes the firing strength values of other rules in order to fire only the selected rule R_{i}.
  - P_{op} is a place that models the output parameter. As shown in Fig. 6, the place P_{op} represents the traffic regulation rate TR.
  - $P_{\text{cls}} = \{P_{\text{cls1}}, P_{\text{cls2}}, \ldots, P_{\text{cls}n}\}$ models a set of places that describe the different decisions of the defined rules. The places P_{cls1}, P_{cls2}, P_{cls3}, and P_{cls4} illustrate the following conclusions respectively, "increased largely", "increased", "decreased", and "decreased largely". Only one place among all places will contain a token which represent the conclusion of the selected wining rule. For instance, the conclusion of the selected rule R_1 is "increased largely" if T_{Dm1} transfers a token from the place P_{SWR1} to the place P_{cls1}.
  - $T_{\text{OMf}} = \{T_{\text{OMf1}}, T_{\text{OMf2}}, \ldots, T_{\text{OMf}n}\}$ is a set of transitions that represent the output membership functions. T_{OMf1}, T_{OMf2}, T_{OMf3}, and T_{OMf4} represent the calculation performed by the used fuzzy method to compute the membership degree of respectively, $u_{large\_increase}(\text{TR}), u_{increase}(\text{TR}, u_{decrease}(\text{TR}), u_{large\_decrease}(\text{TR})$,
  - $P_{\text{OMd}} = \{P_{\text{OMd1}}, P_{\text{OMd2}}, \ldots, P_{\text{OMd}n}\}$ is a set of places that represent output membership degree. The places P_{OMd1}, P_{OMd2}, P_{OMd3}, and P_{OMd4} indicate that the output parameters of "TR is increased", "TR is increased largely", "TR is decreased", and "TR is decreased largely" are satisfied with the following membership degree, $u_{large\_increase}(\text{TR})$, $u_{increase}(\text{TR}), u_{decrease}(\text{TR})$, $u_{large\_decrease}(\text{TR})$, respectively.

Now that we explained how traffic regulation is performed in FuzzyQoS-2, let us focus hereafter on another vital aspect for ensuring the control of congestion which is the buffer management. In order to deal with the dynamic buffer occupancy and the uncertain and imprecise nature of ad hoc network information, a fuzzy logic approach for threshold selection is also used in FuzzyQoS.

2.5. Fuzzy logic approach for threshold buffer management (FuzzyQoS-3)

In the aim of improving the reliability of the proposed model, fuzzy buffer management mechanism (FuzzyQoS-3) has been integrated. Given the study performed in FuzzyMARS [7], fuzzy logic promises to offer an efficient tool for buffer management by using adequate thresholds that deal with the imprecise information in a wireless ad hoc network. Also, fuzzy logic has been successfully applied to the queue management in the cell-switching networks [27]. FuzzyQoS-3 model applies a fuzzy technique for buffer management based on fuzzy sets theory. The later extends the classical logic set $\{0, 1\}$ to use linguistic variables (e.g. full buffer, merely full buffer, empty buffer). Using fuzzy logic, FuzzyQoS-3 investigated the fuzzy thresholds ability to adapt to the dynamic conditions over the classical inflexible thresholds.

The classical thresholds are characterized by their limitation and restriction, because the selection of threshold is based on a single value. Thus, the utilization of a buffer may be either "poor" or "surcharged". When the selected value is small (e.g. 30% of capacity), then the admission of new packets is possible only when the buffer occupancy is low. This indicates a poor utilization of the buffer; since most of incoming packets are rejected even if the buffer is almost unfilled. On the other hand, when the selected value is big (e.g. 90% of capacity), problems may occur when the bursty traffic is used. The transmission of packets generated by a bursty traffic is very changing. It can vary from small to "near-peack" rate in a short period of time.

On the other hand, it is observed that most of events occurring in an ad hoc network are dynamic and random. Manually predefining a value for threshold is not suitable. In addition, it is important to note that the rate of packets arriving at a particular node is not static. The classical threshold mechanism divides the buffer into an "admitted" part and a "no-admitted" part. Let consider that the threshold of the buffer shown in Fig. 7a is equal to 60% (the values are based on the simulations). In this scheme, the occupancy level may range from 0 to 60%. When the buffer occupancy is superior to 60%, no incoming packets are accepted in the buffer. Therefore, the change in decision making from "admit state" to "no-admit state" is performed from 60%–61%. This means that a small variation in the buffer occupancy may influence the decision making of incoming packets.

In the proposed FuzzyQoS-3, we attempt to extend the two-discrete states "admit" and "no-admit" of the buffer occupancy by using fuzzy logic. The aim of introducing fuzzy logic is to develop a more realistic representation of buf-

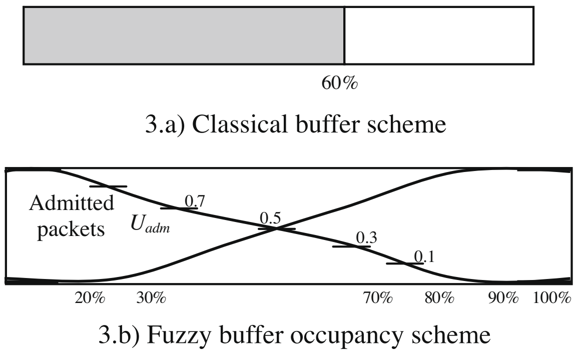


Fig. 7. Classical and fuzzy buffer schemes.

fer occupancy that helps to offer an efficient decision making. Hence, the definition of "buffer occupancy" will consider the two fuzzy cases of "getting full" and "not getting full", rather than "admit" and "no-admit", in the existing approaches. This fuzzy representation replaces the two-discrete sets by a continuous set membership, and performs small gradual transitions between different states of buffer occupancy.

Based on the fullness of the buffer, the fuzzy membership function aims to determine the fuzzy threshold. Several membership functions may be used for that purpose: "triangular", trapezoidal", or "sigmoid" function. These functions can give a representation about the buffer fullness level. In FuzzyQoS-3, we used the sigmoid membership function. This choice is based on the fact that this function would reflect well the dynamic occupancy of the buffers that we want to model.

Fig. 7.b shows that the admit membership function is inversely proportional to the occupancy fullness level of buffer. Thus, when the occupancy fullness is small, the value of the admit membership function is big. At higher fullness occupancy levels, the admit membership function value becomes small. When the value of the "no-admit" membership function is getting big, then only a small quantity of packets will be permitted to enter the buffer. In Fig. 7.b, the value of the membership function is represented by the symbolu_{adm}. The fuzzy rules associated are as follows:

If the value of the admit membership function is big, then increase the accepted incoming packets into buffer

If the value of the admit membership function is small, then reduce the accepted incoming packets into buffer

The previous fuzzy rules are illustrated by Fig. 7.b. The rejection of packets is controlled based on the degree of fullness of the buffer. For instance, when the buffer is occupied at 40%, this means that the value of u_{adm}) is about 0.7 (i.e. the amount of packets admitted is about 70%). Then, about "30%" of incoming packets will be not admitted. Note that the fuzzy threshold approach covers the continuous set of values representing possible buffer occupancy (i.e. from 0 to u_{adm}). This is opposite to the classical threshold approaches that hold only one predefined single value. Therefore, fuzzy logic adds more flexibility to the threshold selection.

3. Performance evaluation

The simulation of the proposed solution is studied with the scalable ns-2 simulator. We compared the performance of FuzzyQoS with the 'original model', FuzzyMARS described in our previous work [7] and the SWAN model described in [1]. We use the word 'original model' to refer to IEEE 802.11 wireless networks without FuzzyQoS mechanisms. Each mobile host has a transmission range of 250 m and shares an 11 Mbps radio channel with its neighboring nodes. The simulation is performed in two steps: the first simulation investigates the performance of the proposed model in an environment characterized by a single shared channel. The second simulation considers a multi-hop environment with different mobility scenarios.

3.1. Performance of a single shared channel

In this section, we consider a single hop environment that consists of a square shape of 150 m $\times$ 150 m. The simulation includes a variety of traffic types; FTP macro-flows, WEB micro-flows, and real-time flows. The video and voice flows representing real-time traffic are active and monitored for the duration of 100 s. Video traffic is modeled as 200 Kbps constant rate traffic with a packet size of 512 bytes. Voice traffic is modeled as 32 Kbps constant rate traffic with a packet size of 80 bytes. We have considered different scenarios of node's scalability (30 nodes, 50 nodes, 70 nodes, etc.).

In order to better understand the properties of the FuzzyQoS, the simulation considers multiple scenarios of TCP best-effort traffic, voice and video flows. The TCP traffic is modeled as a mixture of FTP and Web traffic. Web traffic represents micro-flows, whereas FTP traffic corresponds to macro-flows. TCP flows are greedy FTP type of traffic with packet size of 512 bytes. Web traffic is modeled as short TCP file transfers with random file size and random silent period between transfers.

The length of the silent period between two transfers is Pareto in distribution with the shape parameter of 1.2. The file size is also driven from a Pareto distribution with a mean file size of 10 Kbytes and shape parameter of 1.2. For the reasons of simplification and clarification, we have chosen to put the QoS performance results of "FuzzyQoS", "FuzzyMARS", "SWAN" and "Original" into different figures, and to make the comparison between "FuzzyQoS" and other solutions separately. This shows precisely the difference in terms of performances between FuzzyQoS and other models.

3.1.1. Impact of scalability of video traffic

Figs. 8–10 show the impact of scalability of number of UDP video flows on the average end-to-end delay. The simulation uses a mixture of real-time traffic and TCP best-effort traffic which consists of 16 Web and FTP flows. It is observed in Fig. 8 that the original model (IEEE 802.11) shows an average delay larger than 12 ms with only 5 video flows and over 20 ms with 15 or more video flows. FuzzyQoS shows delay inferior to 2 ms with 5 video flows and less than 2.4 ms with 20 video flows. Hence, the reduction

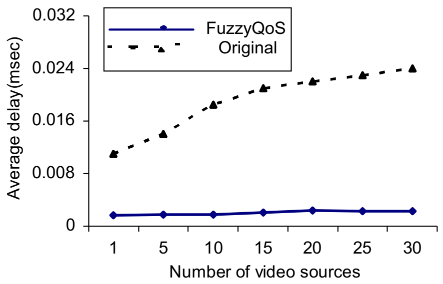


Fig. 8. Average delay in IEEE 802.11 and FuzzyQoS models vs number of video flows.

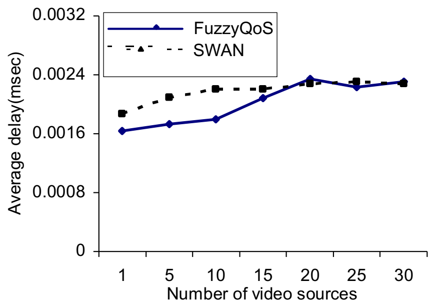


Fig. 9. Average delay in FuzzyQoS and SWAN models vs number of video flows.

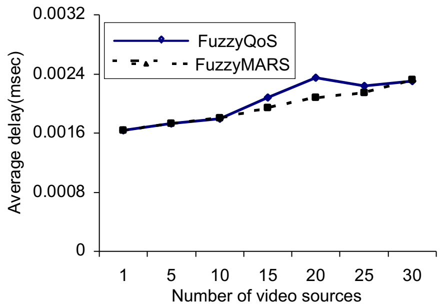


Fig. 10. Average delay in FuzzyQoS and FuzzyMARS models vs number of video flows.

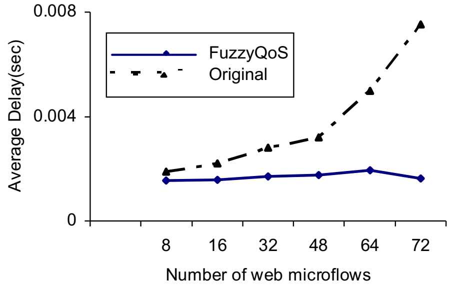


Fig. 11. Average delay in IEEE 802.11and FuzzyQoS models vs number of web micro-flows.

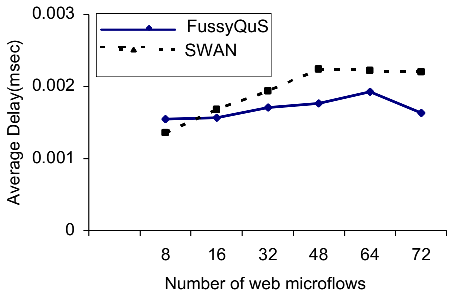


Fig. 12. Average delay in FuzzyQoS and SWAN models vs number of web micro-flows.

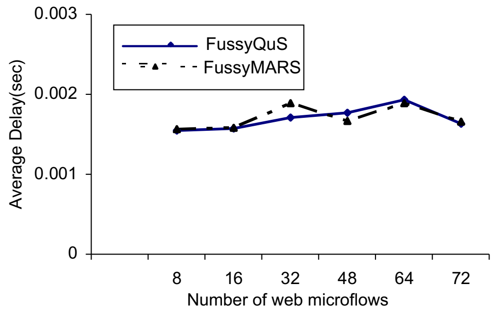


Fig. 13. Average delay in FuzzyQoS and FuzzyMARS models vs number of web micro-flows.

achieved by FuzzyQoS in terms of the average delay is about 90% in comparison to the original model. On the other hand, Fig. 9 illustrates that for up to 20 video flows, FuzzyQoS outperforms SWAN by about 13%. We observe in Fig. 10, that the average delay in both models FuzzyQoS and FuzzyMARS is almost the same. The average delay in both models grows slowly as the video flows number increases.

When the number of video flows becomes high, the average delay of traffic in FuzzyMARS becomes slightly smaller than in FuzzyQoS. The impact of high number of video flows on the delay is due essentially to the congestion in the nodes.

3.1.2. Impact of scalability of web traffic

Figs. 11–13 show the impact of the scalability of a growing number of web micro-flows on the average end-to-end delay. It is observed in Fig. 11 that the increasing number of web micro-flows has much more impact on the average delay in IEEE 802.11 than in FuzzyQoS.

The average delay in FuzzyQoS remains around 1.8 ms, whereas, in the original model when the number of web micro-flows increases from 8 to 72 web micro flow the average end-to-end delay grows from 1.8 to 7 ms. On the other hand, it is observed in Fig. 12 that the average delay in SWAN and FuzzyQoS models is similar for up to 16 web micro-flows. For the highest number web micro-flows, the average delay of traffic in FuzzyQoS becomes smaller than in SWAN by about 18%. Fig. 13 observes almost a similar performance between FuzzyQoS and FuzzyMARS models, the gain achieved by the proposed model is about 9%.

3.2. Performance in multihop environment

In this section, the simulation considers a multihop network of 50 mobile nodes. The network area has a rectangular shape of 1500 m × 300 m that minimizes the effect of network partitioning. The AODV protocol [28] is chosen as a routing protocol. In this multihop network, we consider a mixture of real-time and TCP best-effort traffic. The real-time traffic is modeled as 4 voice and 4 video flows. The TCP traffic is modeled as a mixture of web micro-flows and FTP macro-flows traffic.

3.2.1. Impact of scalability of TCP flows

Figs. 14–19 show the scalability impact of the increasing number of TCP flows on the average end-to-end delay and throughput of traffic. Fig. 14 illustrates a significant difference in terms of the average delay between FuzzyQoS and the original model. The average delay in FuzzyQoS grows slowly with the increasing number of TCP flows, and it remains between 2 and 3 ms. In contrast, the average delay in the original model grows from 7 to 31 ms as the number of TCP flows increases from 2 to 12 flows. Hence, the gain achieved by FuzzyQoS in terms of the average end-to-end delay is about 74–92%. In Fig. 16, we observe that the average delay in FuzzyMARS grows slowly with the increasing number of TCP flows, and it remains almost less than 3 ms. We observe that the average delay of TCP traffic in FuzzyQoS is almost the same as in FuzzyMARS; the latter outperforms FuzzyQoS by about 3%. Fig. 18 shows the average end-to-end delay in both FuzzyQoS and SWAN models. It is shown that the average delay is almost inferior to 3 ms in the proposed model, whereas, in

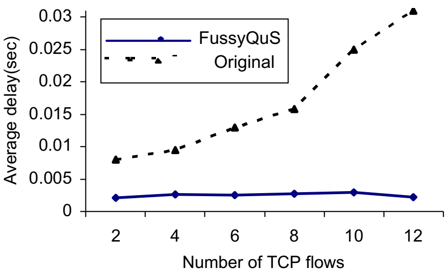


Fig. 14. Average delay in IEEE 802.11 and FuzzyQoS models vs number of TCP flows.

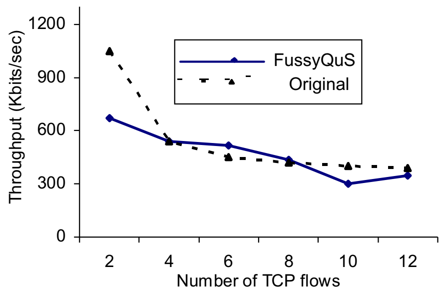


Fig. 15. Average throughput in IEEE 802.11 and FuzzyQoS models vs number of TCP flows.

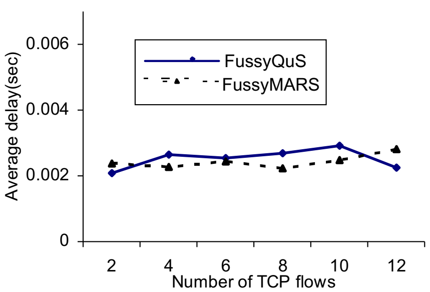


Fig. 16. Average delay in FuzzyQoS and FuzzyMARS models vs number of TCP flows.

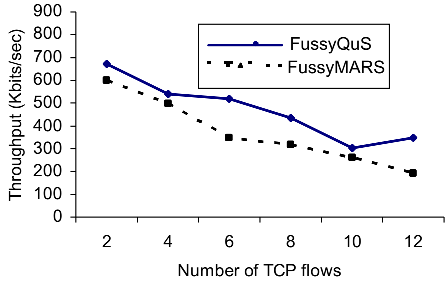


Fig. 17. Average throughput in FuzzyQoS and FuzzyMARS models vs number of TCP flows.

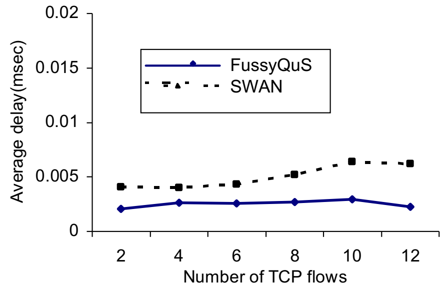


Fig. 18. Average delay in FuzzyQoS and SWAN models vs number of TCP flows.

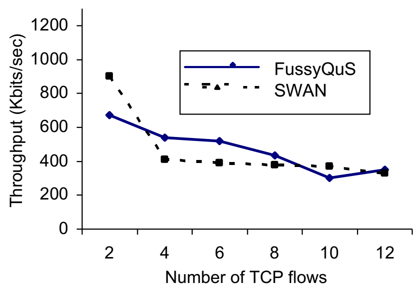


Fig. 19. Average throughput in FuzzyQoS and SWAN models vs number of TCP flows.

SWAN model the average delay is around 5 ms. This means that the achieved gain offered by FuzzyQoS is about 49% in terms of delay.

Figs. 15, 17 and 19 illustrate the impact of growing number of TCP flows on the average throughput of TCP traffic over the different models of simulation. The average throughput of the TCP traffic in FuzzyQoS is almost the same as in IEEE 802.11, as shown in Fig. 15. At a lower number of TCP flows, the average throughput in the original model is superior to that in FuzzyQoS. A similar result is observed in Fig. 19 between FuzzyQoS and SWAN. On the other hand, we observe in Fig. 17 that the average throughput of the TCP traffic in the proposed model is about 21% better that than the FuzzyMARS.

These results confirm that by adopting the FuzzyQoS mechanisms, we can achieve a reduction in terms of average end-to-end delay an estimated, 74%–92% and 49% in comparison to the original and SWAN models respectively, with almost the same throughput. Moreover, FuzzyQoS outperforms FuzzyMARS in terms of average throughput around 21% at cost of 3% decrease in the average delay.

3.2.2. Impact of mobility

The impact of mobility on the performances of FuzzyQoS is explored in Figs. 20–25. The real-time traffic is modeled in the same manner as discussed previously. The best-effort TCP flows consist of both web and FTP flows. The random waypoint mobility model [29] is implemented at each node in the network. In the beginning, the nodes are randomly placed in the area. Then, each mobile node selects a random destination and moves with a random speed up to a maximum speed of 20 m/s. After reaching the destination, the node will stay there for a given "pause time" and then begin starts to move toward another destination. This process is repeated during all simulation time.

Fig. 20 shows that the average end-to-end delay in FuzzyQoS increases slowly. The average delay in the proposed model remains almost less than 5.4 ms, while the average delay in the original model grows from 25 to 38 ms. This means that the proposed FuzzyQoS achieves a reduction of about 79%–87%. On the other hand, it is observed in the Fig. 21 that the throughput of TCP best-effort traffic decreases slowly in the original model as the mobility increases. The average throughput in FuzzyQoS is superior to that of the original model by about 33% for different mobility scenarios.

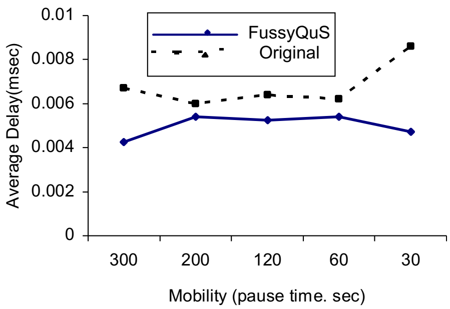


Fig. 20. Average delay in IEEE 802.11 and FuzzyQoS models vs mobility.

We observe in Fig. 22 that the average end-to-end delay in FuzzyMARS increases slowly and grows only for the highest mobility scenarios. Although the average delay of-

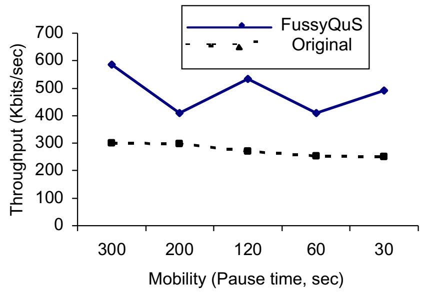


Fig. 21. Average throughput in IEEE 802.11 and FuzzyQoS models vs mobility.

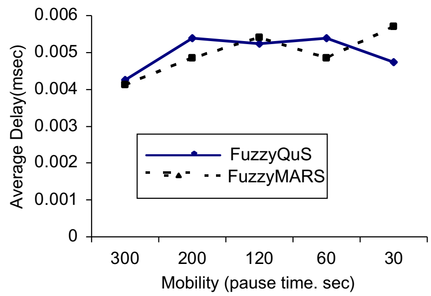


Fig. 22. Average delay in FuzzyQoS and FuzzyMARS models vs mobility.

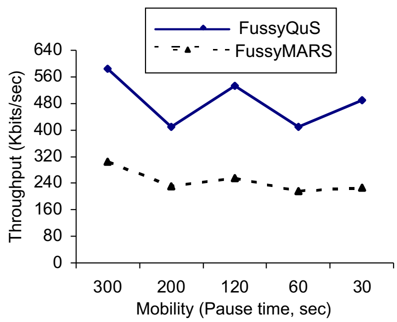


Fig. 23. Average throughput in FuzzyQoS and FuzzyMARS models vs mobility.

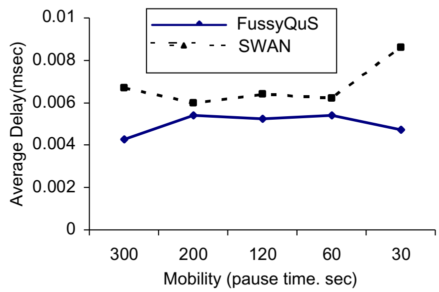


Fig. 24. Average delay in FuzzyQoS and SWAN models vs mobility.

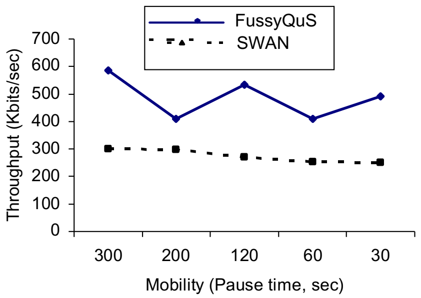


Fig. 25. Average throughput in FuzzyQoS and SWAN models vs mobility.

fered by FuzzyMARS is about 2% better than in FuzzyQoS the FuzzyQoS throughput is 49% better in different mobility scenarios as shown in Fig. 23. During the mobility of nodes, some flows are dropped in both FuzzyMARS and FuzzyQoS models because of the difficulty in capturing the dynamics of the environment in the ad hoc network.

Fig. 24 illustrates the average end-to-end delay with different mobility scenarios in both FuzzyQoS and SWAN models. For different mobility scenarios, the average delay offered by FuzzyQoS is about 10%–36% better than that offered by SWAN. It is shown in Fig. 25 that for different mobility scenarios, the throughput in FuzzyQoS is better than in SWAN model by about 43%.

The previous investigation of impact of TCP traffic and mobility illustrates clearly that the proposed FuzzyQoS model provides an average end-to-end delay with low and almost stable values, which is a promising result for jitter-sensitive multimedia services. This advantage is illustrated by the trace graphs as shown in the following.

Figs. 26–29 show some trace graphs that observe the impact of TCP flows and nodes scalability on the average delay in FuzzyQoS in both single hop and multihop wireless ad hoc environment. The simulation consists of a mixture of 8 real-time flows and TCP flows (i.e. a mixture of FTP and web flows), which are modeled as in the previous simulations. Figs. 26 and 27 trace the packet delay of a single hop environment using, 10 and 20 TCP flows with 20 and 30 nodes respectively. Figs. 28 and 29 trace the packet delay of a multiple hop environment using, 10 and 20 flows with 20 and 30 nodes respectively. It is observed from the trace graphs that the average end-to-end delay in the proposed model increases slowly as the number of nodes becomes high. For different scenarios of TCP and nodes

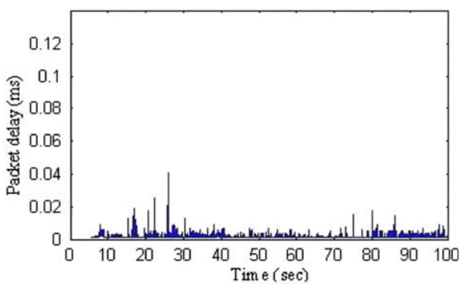


Fig. 26. Packets delay in FuzzyQoS with 10 TCP vs simulation time (Single-hop setting).

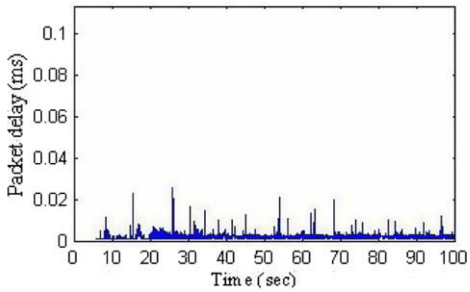


Fig. 27. Packets delay in FuzzyQoS with 20 TCP vs simulation time (Single-hop setting).

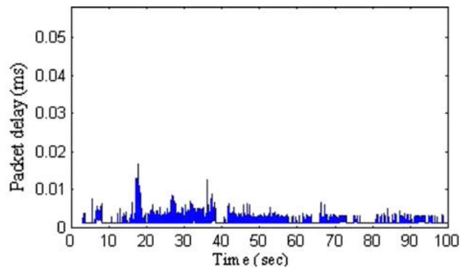


Fig. 28. Packets delay in FuzzyQoS with 10 TCP vs simulation time (Multihop setting).

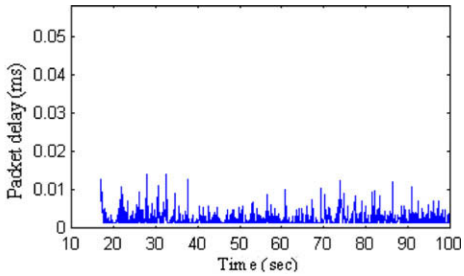


Fig. 29. Packets delay in FuzzyQoS with 20 TCP vs simulation time (Multihop setting).

scalability, FuzzyQoS experiences a low and almost stable average delay around 2 ms. This result indicates that the proposed model can support real-time traffic with low and stable delay for different network conditions, which is a remarkable benefit for jitter-sensitive multimedia applications.

4. Conclusion

In this paper we explored the usage of an intelligent cross-layer QoS approach for QoS support in wireless ad hoc networks. FuzzyQoS aims to improve the traffic regulation and the control of congestion in multimedia applications. Moreover, FuzzyQoS explores a new Fuzzy Petri nets technique for modeling and analyzing the QoS decision making for the traffic regulation in wireless ad hoc networks.

FuzzyQoS makes use of the feedback delay information from the network in order to perform a fuzzy logic traffic regulation which ensures that best-effort traffic coexists well with real-time traffic in the multimedia applications. Additionally, FuzzyQoS explores the use of fuzzy threshold because of the importance of threshold notion for the discarding of data packets and adapting the traffic service to the occupancy of buffers. We have evaluated the performance of FuzzyQoS using the ns-2 simulator under diverse mobility, traffic, and channel conditions. In terms of traffic scalability, the simulations have shown that we achieved a gain in terms of average end-to-end delay by about 74%–92% and 49% in comparison to, the original (i.e. IEEE 802.11 wireless networks) and SWAN models respectively. The simulation outcomes indicate that fuzzy logic system can be hopeful to deal with the dynamics of ad hoc networks when managing QoS delivery especially for supporting multimedia services.

References

[1] G.H. Ahn, A. T. Campbell, A. Veres, L. H. Sun, "SWAN: Service Differentiation in Stateless Wireless Ad Hoc Networks, in: IEEE INFOCOM, June, 2002.

[2] S.-B. Lee, G.-S. Ahn, X. Zhang, A.T. Campbell, INSIGNIA: An ip-based quality of service framework for mobile ad hoc networks, Journal of Parallel and Distributed Computing, Special Issue on Wireless and Mobile Computing and Communication 60 (4) (2000) 374–406.

[3] H. Xiao, W.K.G. Seah, A. Lo, K. Chaing, Flexible QoS model for mobile ad hoc networks, in: The Proceedings of IEEE Vehicular Technology Conference, Tokyo, May, vol. 1, 2000, pp. 445–449.

[4] Y.L. Morgan, T. Kunz, PYLON: An architectural framework for ad hoc QoS interconnectivity with access domains, HICSS'03, Hawaii, USA, Jan 2003.

[5] J.L. Sobrinho, A.S. Krishnakumar, Quality of service in ad hoc carrier sense multiple access networks, IEEE Journal on Selected Areas in Communication 17 (8) (1999) 1353–1368.

[6] L. Khoukhi, S. Cherkaoui, Toward neural networks solution for multimedia support in wireless mobile ad hoc networks, Journal of Networks 4 (2) (2009) 148–161. ISSN: 1796–2056.

[7] L. Khoukhi, S. Cherkaoui, FuzzyMARS: A fuzzy logic approach with service differentiation for wireless ad hoc networks, in: IEEE WirelessCom2005, June 2005.

[8] C.R. Lin, J.-S. Liu, QoS routing in ad hoc wireless networks, IEEE Journal on Selected Areas in Communication 17 (8) (1999) 1426–1438.

[9] L. Khoukhi, S. Cherkaoui, Flexible QoS routing protocol for mobile ad hoc networks, in: IEEE International Conference on Telecommunication (IEEE ICT2004), Brazil, Augt 2004.

[10] S. Chen, K. Nahrstedt, Distributed quality-of service in ad hoc networks, IEEE Journal on Selected Areas in Communications 17 (8) (1999) 1426–1438.

[11] C.-R. Lin. On-demand QoS routing in multihop mobile networks, IEEE INFOCOM 2001, April 2001, pp. 1735–1744.

[12] S. Chen, K. Nahrstedt, On finding multi-constrained paths, in: IEEE International Conference on Communication, 1998, pp. 874–879.

[13] C. Venkatesh, N. Yadaiah, A.M. Natarajan, Dynamic source routing protocol using fuzzy logic concepts for ad hoc networks, Academic Open International Journal 15 (2000).

[14] D. Gianni, D. Marco, AntNet: Distributed stigmergetic control for communications networks, Journal of Artificial Intelligence Research 9 (1998) 317–365.

[15] R. Chadha, Y. Cheng, J. Chiang, G. Levin, S. Li, A. Poylisher, Policy-based mobile ad hoc network management for DRAMA, in: IEEE MILCOM2004, Military Communications Conference, Monterey, CA, 2004.

[16] R. Oliveira, T. Braun, A delay-based approach using fuzzy logic to improve TCP error detection in ad hoc networks, IEEE Wireless Communications and Networking Conference 5 (1) (2004) 1660–1665.

[17] C.C. Lee, Fuzzy logic in control systems: fuzzy logic controller-Part I and II, IEEE Transactions on Systems, Man, and Cybernetics 20 (2) (1990) 404–418.

[18] L.A. Zadeh, Fuzzy logic = computing with words, IEEE Transactions on Fuzzy Systems 4 (2) (1996) 104–111.

[19] M.B. Dwyer, L.A. Clarke, A compact Petri net representation and its implication for analysis, IEEE Transaction Software Engineering 22 (1996) 794–811.

[20] T. Murata, T. Suzuki, S.M. Shatz, Fuzzy-timing high level Petri net model of a real-time network protocol, in: The proceeding of ITC-CSCC 96, Seoul, Korea, 1996, pp. 1170–1173.

[21] S.-M. Chen, J.-S. Ke, J.-F. Chang, knowledge representation using fuzzy Petri nets, IEEE Transaction on Knowledge Data Engineering 2 (3) (1990) 311–319.

[22] G. Looney, Fuzzy Petri nets for rule-based decision making, IEEE Transaction on System, Man, and Cybernetics 18 (1) (1998) 178–183.

[23] T. Cao, A.C. Sanderson, Variable reasoning and analysis about uncertainty with fuzzy Petri nets, in: International Conference on Application and Theory of Petri Nets, Troy, NY, 1993, pp. 126–175.

[24] S.I. Ashon, Petri net models of fuzzy neural networks, IEEE Transaction on System, Man, and Cybernetics 25 (6) (1995) 926–932.

[25] A. Chaudhury, D.C. Marinescu, A. Whinston, Net-based computational models of knowledge processing systems, IEEE Expert 8 (2) (1993) 79–86.

[26] L.A. Zadeh, Knowledge representation in fuzzy logic, IEEE Transaction knowledge Data Engineering 1 (1989) 89–100.

[27] A.R. Bonde, S. Ghosh, A comparative study of fuzzy versus fixed thresholds for robust queue management in cell-switching networks, IEEE Transaction on Networking 2 (4) (1994) 337–344.

[28] C.E. Perkins, E.M. Royer, ad hoc on-demand distance vector routing, in: IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, LA, 1999, pp.90–100.

[29] J. Broch, D. Maltz, D. Johnson, Y.-C. Hu, J. Jetcheva, A performance comparison of multihop wireless ad hoc network routing protocols, in: ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom), October 1998.

[30] R.E. Bellman, L.A. Zadeh, Decision-making in a fuzzy environment, Management Science Journal 17 (1970) 141–164.

[31] D. Dubois, H. Prade, An introduction to fuzzy systems, International Journal of Application and Mathematics and Computation Sciences 6 (3) (1996) 485–503.

[32] H.J. Zimmermann, Fuzzy Set Theory, third ed., Kluwer Academic Publishers, Boston, 1996. ISBN 0-7923-9624-3.

Lyes Khoukhi is an associate professor at the University of Technology of Troyes (France), since 2009. In 2008, he was researcher at the computer sciences department of the University of Montreal (Canada). He received Ph.D. degree in Electrical and Computer Engineering from the University of Sherbrooke (Canada) in 2007, and M.Sc. degree in Computer Engineering from University of Versailles (France) in 2002. During 2003–2007, he stayed in INTERLAB Communications Research Laboratory, Sherbrooke University. His research interests include wireless communications, mobile ad hoc networking, Quality of Service, and intelligent systems.

Soumaya Cherkaoui is an Associate Professor at Sherbrooke University, Canada which she joined in 1999. Since 2000, she is also the director of INTERLAB, a research group comprising more than 12 faculty and research assistants which conducts research funded both by government and industry. Before joining Sherbrooke University as a faculty member, she worked for industry as a project leader. She has over 50 publications in the areas of network protocols and distributed systems.