Towards Reasoning Vehicles: A Survey of Fuzzy Logic-Based Solutions in Vehicular Networks

IRINA TAL and GABRIEL-MIRO MUNTEAN, Dublin City University

Vehicular networks and their associated technologies enable an extremely varied plethora of applications and therefore attract increasing attention from a wide audience. However, vehicular networks also have many challenges that arise mainly due to their dynamic and complex environment. Fuzzy Logic, known for its ability to deal with complexity, imprecision, and model non-deterministic problems, is a very promising technology for use in such a dynamic and complex context. This article presents the first comprehensive survey of research on Fuzzy Logic approaches in the context of vehicular networks, and provides fundamental information which enables readers to design their own Fuzzy Logic systems in this context. As such, this article describes the Fuzzy Logic concepts with emphasis on their implementation in vehicular networks, includes classification and thorough analysis of the Fuzzy Logic-based solutions in vehicular networks, and discusses how Fuzzy Logic could be employed in the context of some of the key research directions in the 5G-enabled vehicular networks.

CCS Concepts: • **Networks** → **Wireless access networks**; **Ad hoc networks**;

Additional Key Words and Pharses: Fuzzy logic, computational intelligence, smart vehicles, vehicular networks

ACM Reference format:
Irina Tal and Gabriel-Miro Muntean. 2017. Towards Reasoning Vehicles: A Survey of Fuzzy Logic-Based Solutions in Vehicular Networks. *ACM Comput. Surv.* 50, 6, Article 80 (December 2017), 37 pages.
https://doi.org/10.1145/3125640

1 INTRODUCTION

Vehicular networks (VN) are acknowledged for their great potential for supporting an extremely large range of communication-based applications in three major areas of interest: active road safety, traffic efficiency, and management and infotainment [1, 2]. The highly positive impact of these applications on both the society as a whole and individual lives is measured in terms of the number of crashes avoided and number of saved lives, reduced damage to property, reduced traffic congestion, decreased fuel consumption, improvement in environment conditions, and increased satisfaction of the drivers in traffic. It is therefore a natural consequence that vehicular networking will have its distinct place in the context of 5G, the next generation wireless

This work is supported in part by Dublin City University under the Daniel O'Hare Research Scholarship scheme, Science Foundation Ireland grant 10/CE/I1855 to Lero (http://www.lero.ie) and European Union's Horizon 2020 Research and Innovation programme under Grant Agreement no. 688503 for the NEWTON project (http://www.newtonproject.eu).
Authors' address: I. Tal, Room 3.19, School of Computing, National College of Ireland IFSC, Mayor Street, North Dock, Dublin 1; email: irina.tal@ncirl.ie; G.-M. Muntean, Room S326, Research and Engineering Building, School of Electronic Engineering, Dublin City University, Glasnevin, Dublin 9, Dublin, Ireland; email: gabriel.muntean@dcu.ie.

https://doi.org/10.1145/3125640

communications [3]. VNs inter-link “smart” vehicles (i.e., they are the nodes of the network) which communicate with each other and with the infrastructure via V2X communications. In the context of the relatively high mobility of the vehicle-nodes, these networks are very dynamic in terms of both their topology and their delivery-related characteristics. Modeling these networks has a high degree of imprecision and is difficult to define accurate analytical models. Fuzzy Logic (FL), known for its ability to deal with imprecision and non-deterministic problems, is therefore often used in this VN context. FL provides human-like reasoning based on linguistic information and has already been widely used in self-control systems (automatic) in automotive and electronics industries.

In the literature, there are many surveys and tutorials dedicated to the emergent field of VNs. These articles include both more general studies [2, 4–6], and particular works that review a certain problem in the context of VNs such as VN-dedicated standards [7], routing [8, 9], MAC protocols [10], clustering algorithms [11, 12], vehicular social networking [13], vehicular cloud networking [14], and vehicular content delivery networks [15]. On the other hand, there are also reviews dedicated to computational intelligence and mathematical intelligent frameworks applicable in networking [16–21]. However, none of the later surveys focuses on FL applied to VNs. In Reference [16], some of the first FL-based solutions in telecommunications are presented. Game theory application in network selection solutions in heterogeneous wireless networks has been surveyed in Reference [17] while its application in network security and privacy issues has been surveyed in Reference [22]. Swarm intelligence applicability in the context of wireless heterogeneous networks is surveyed in Reference [23]. In Reference [18], a survey and tutorial on the mathematical modeling of network selection in heterogeneous networks is presented, where FL is included among the mathematical frameworks used in modeling network selection. In Reference [19], the authors mention FL among the computational intelligence techniques applied in the context of wireless sensor networking. The surveys presented in References [20] and [21] present the employment of intelligent techniques used for reasoning in cognitive radio networks, and FL appears among these techniques.

This article is the first comprehensive survey of research on FL approaches in VN context and aims to provide fundamental information to enable readers to understand existing FL systems and their particularities and design their own FL systems in the VN context. This work includes the following main contributions:

— A detailed presentation of FL concepts with the emphasis on their implementation in VNs;
— A step-by-step tutorial on how to design a FL-based system in VN context;
— A classification and thorough analysis of the FL-based solutions in VNs and their applications along with a discussion on the VN-related challenges that FL address and why FL was considered to be suitable for addressing them. Design decisions that relate to the FL system are also highlighted for each presented solution;
— A summary of the most important lessons learnt as well as a discussion on future research directions in this context.

The structure of this article is as follows: Section 2 familiarizes the readers with VNs. Section 3 introduces FL concepts with the emphasis on their implementation in VN, presents a step-by-step tutorial on how to design a FL-based system in VN context, and introduces a three-class classification of FL-based solutions designed in VN landscape. Sections 4, 5, and 6 present a comprehensive analysis of the FL-based solutions in VNs that subscribe to the three main classes identified in the classification. The last section draws conclusions in form of main lessons learnt and identifies future research directions.

2 INTRODUCTION TO VEHICULAR NETWORKS

This section presents an overview of VNs: main enabling VN communication technologies are presented and the VN's specific characteristics that also impose the main challenges in VNs. Appropriate resources to the reader for further study are also indicated.

2.1 VN-Enabling Technologies Supporting Standards

This section presents an overview of VNs: main enabling VN communication technologies are presented and the VN's specific characteristics that also impose the main challenges in VNs. Appropriate resources for the reader for further study are also indicated. The nodes in VNs communicate with each other and to the infrastructure via V2X communications. There are various proposed standards enabling V2X communications and more enabling technologies proposed to be developed in the context of 5G. The main existing standards are briefly mentioned below. More details about these standards can be found in References [1], [2], and [7].

Wireless Access for Vehicular Environment (WAVE) standards [7] include the following standards dedicated to vehicular communications: IEEE 802.11p and IEEE P1609.x. IEEE 802.11p,[1] developed to provide wireless access to vehicles, is a new amendment of IEEE 802.11 that was ratified in July 2010. Its aim was to make IEEE 802.11 suitable to the ever-changing transportation environment and able to deal with very short latencies. The IEEE P1609.x suite of standards covers the entire VN scope of services, from application down to the MAC network layer, as IEEE 802.11p already covers the Physical and MAC layers.

ITS-G5[2] is the profile standard of IEEE 802.11p, defined by the European Telecommunication Standards Institute (ETSI) in order to adapt IEEE 802.11p to the European spectrum. ETSI also defined in a standard the ITS station and communication reference architecture that covers the whole network stack.[3]

Other types of technologies, not dedicated exclusively to VNs, are also used in supporting vehicular communications (e.g., WiFi, cellular technologies). Among the cellular technologies, LTE is currently the most promising in enabling vehicular applications [24]. However, LTE was originally designed for mobile broadband traffic and the requirements for V2X traffic are very different. For instance, direct V2V communication should be supported which means that the infrastructure should be bypassed. Steps were done in this direction as from Release 12 (Rel.12) a new feature known as Proximity Services is specified within 3GPP[4]. Proximity Services Direct Discovery and Proximity Services Direct Communication enable Device-to-Device (D2D) communications [25]. However, this current release of the Proximity Services specification has not considered the requirements of V2X communications. It was designed for public safety and commercial consumer scenarios that lead to low mobility support. Therefore, D2D communications specified in Rel.12 is not really suitable for V2V communications, especially not in highway scenarios characterized by high speeds. Improvements of these specifications that take into consideration V2X characteristics are already planned and they are very likely to be introduced in 5G communications.

[1]IEEE 802.11p, "Draft Amendment to Standard for Information Technology-Telecommunications and Information Exchange Between Systems-Local and Metropolitan Area Networks- Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications- Amendment 6: Wireless Access in Vehicular Environment", 2010.

[2]ETSI EN 302 663 V1.2.1, "Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5GHz", 2013.

[3]ETSI TS 102 636-3 V1.1.1, "Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 3: Network Architecture", 2010.

[4]3GPP TS 23.303. (July 2015). Proximity-based services (ProSe); Stage 2.V12.5.0 (Rel-12).

2.2 VN Characteristics

VNs have specific characteristics that differentiate them from any other type of networks. Some of these characteristics are very attractive to the researchers, while others are creating new technical challenges that need to be addressed. Next, the most sought-after VN characteristics are discussed.

— Predictable mobility is possible in VNs due to the fact that vehicle movement is constrained by roads and traffic regulations.
— Theoretical unlimited power is possible due to the fact that the vehicle-node is capable of generating its own power while moving. Power is usually a very serious issue in the case of mobile nodes. This is not applicable for all types of vehicles (e.g., electric vehicles).
— High computational and storage capabilities: theoretically vehicles can afford significant storage, computational, and communication capabilities, but this is not applicable to all types of vehicles (e.g., bikes).

Most challenging VN characteristics are:

— *High Mobility*: The vehicle-nodes have often very high speeds: In highway scenarios, speeds of up to 200km/h may occur, while in city scenarios, speeds of up to 50–70km/h are encountered.
— *Frequent Disconnections and Rapidly Changing Topology*: The aforementioned high mobility of VN nodes leads to a frequent link disconnection between the vehicle nodes and consequently to a rapidly changing VN topology.
— *Potentially Large Scale*: VNs are networks with a potential high number of nodes. There is no limitation in terms of number of nodes, so vehicle nodes can potentially expand over the entire road network.
— *Diversity of Conditions and Applications*: Diversity of conditions mainly refers to the diversity of the network density that can be very sparse (e.g., highway scenario), very dense (e.g., city scenario during rush hours), or something in-between. As presented in the introduction, a large plethora of applications have been envisioned for VNs. The requirements of these applications are as diverse as their range. Consequently, VN-dedicated technology needs to be designed so these networks can cope with all this diversity.

In the presence of these characteristics, some of the main technical challenges of VNs are imposed in the context of MAC protocols, security, handover, and routing and data dissemination protocols [4, 6]. Later sections of this article will indicate how FL can be employed to address these challenges and other challenges imposed by VN applications.

3 FUZZY LOGIC AND VEHICULAR NETWORKS

FL, introduced by Prof. Zadeh in 1965, was defined as an “attempt to mimic human control logic” [26]. FL is the only mathematical framework able to do reasoning based on linguistic information (i.e., linguistic variables) and simulate human reasoning. It is also able to deal with uncertainty and imprecision, model non-deterministic problems and deal with multiple parameters that describe the problem modeled. These characteristics make FL a suitable tool to model and solve a large plethora of real-world problems. Therefore, FL has had a huge impact: it has been used on a wide scale and in various domains, including engineering, medicine, science, and business. There are many well-known, successful FL applications deployed in the industry (e.g., Sendai subway control (Hitachi), aircraft control (Rockwell Corporation), intelligent cruise control (Peugeot, Nissan vehicles) [27]). Besides these well-known industrial applications, many of the current research works in different areas further explore the huge potential of FL. Telecommunication is one of these areas where FL was both successfully employed in solving various challenges and its potential is still being explored. Some of the first FL applications in telecommunications are analyzed in Reference [16] and include: queue modeling, faults and other conditions detection in the telephone networks, dynamic assignments of radio channels, and control applications in the context of asynchronous transfer mode networks. The latest FL applications are in the context of wireless networks (e.g., decision-making in network selection [18]), wireless sensor networks (e.g., decision-making in energy-aware routing and clustering or security protocols [19]) and more recently in the context of cognitive radio networks (e.g., FL reasoning can be employed in inferring the "cognition"/knowledge of cognitive radio networks [20, 21]) and VNs.

3.1 Fuzzy Logic Concepts in Vehicular Networks Context

3.1.1 Linguistic Variables. Cognitive scientists state that humans tend to think in terms of concepts and images rather than in terms of numbers. Consequently, natural language is preferred in describing any kind of problem that requires reasoning. However, mathematical paradigms, with the exception of FL, do not allow for reasoning in natural language. FL allows reasoning in terms of natural language, expressed by means of linguistic variables. These were defined by the founder of FL, Zadeh, as "variables whose values are not numbers, but words or sentences in a natural or artificial language" [28]. The values of linguistic variables are named (atomic) terms. In linguistics, often to the fundamental term modifiers like: very, extremely, almost, approximately, and slightly are associated. FL allows for the usage of these modifiers that are named linguistic hedges.

Example 1. In the VN space, *speed* is one of the most common linguistic variables and it was used to refer either to the speed of the vehicle (e.g., [29]) or to the difference in speed of the vehicles (e.g., [30, 31]). In Reference [29], the linguistic variable *speed* has as main terms *slow, medium,* and *fast.* Additionally, a linguistic hedge is applied to *slow* and *very slow.* Thus, the speed is either: *very slow, slow, medium,* or *fast.* The terms of the *speed* linguistic variable in both References [30] and [31] are *low, medium,* and *high.*

3.1.2 Fuzzy Sets and Membership Functions. A fuzzy set is a fundamental concept in FL and represents a generalization of an ordinary set, called in FL, crisp set. A crisp set is defined either by listing its elements or by defining the condition that makes an element x member of the set. For any value x, there are only two possible statuses: member of the set or non-member. Thus, the statement: "x is member of set A" can be either false (value 0 in binary logic), either true (value 1 in binary logic). A function that illustrates the membership relation (i.e., membership function) can be defined. Let A be a crisp set and $\mu_A(x)$ its membership function described by Equation (1).

$$\mu_A(x) = \begin{cases} 0, & \text{if } x \text{ is not a member of set } A \\ 1, & \text{if } x \text{ is member of set } A \end{cases} \quad (1)$$

A fuzzy set F is described exclusively by its membership function $\mu_F(x)$ that unlike a membership function describing a crisp set, can take more than two values, 0 and 1, taking values in the interval [0, 1]. In this case, $\mu_F(x)$ shows the degree of truth of the element x being member of the set F. This is how fuzzy sets extend the crisp sets and fuzzy logic extends the binary logic from {0, 1} to [0, 1]. The formal definition of fuzzy set F is expressed as a set of pairs $(x, \mu_F(x))$ as in Equation (2)

$$F = \{x, \mu_F(x)) | x€X, \mu_F(x) : X \rightarrow [0, 1]\} \quad (2)$$

where X is called in FL the *universe of discourse* and defines all the possible values that x can take.

A fuzzy set describes what the atomic term of a variable signifies in a mathematical language. The most popular membership functions in FL are: singleton, triangular, trapezoidal, and Gaussian, named after the geometric figures that the pairs $(x, \mu F(x))$ are shaping. In increasing order of

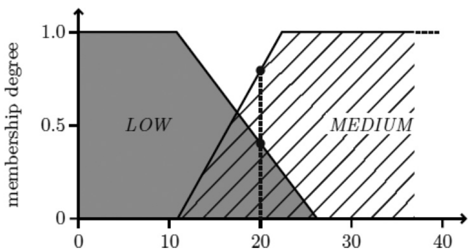

Fig. 1. Fuzzy sets of *speed* ([30]).

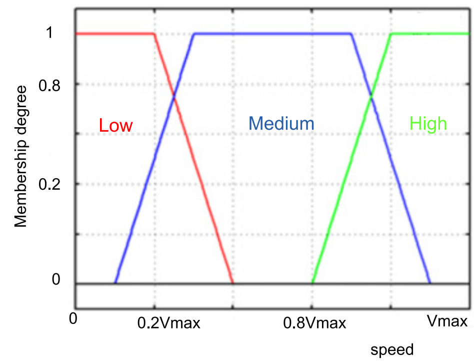

Fig. 2. Fuzzy sets of *speed* in Reference [31].

computational complexity of the membership functions, the singleton is followed by triangular, trapezoidal and Gaussian functions. These are the most common functions used in engineering applications [27].

Example 2. Example 1 mentioned that the linguistic variable *speed* is used in both solutions [30, 31] with the same atomic terms: *low, medium,* and *high*. The interpretation of these terms is however defined by the fuzzy sets associated to these terms. Figure 1 presents the fuzzy sets of *low* and *medium* terms as described in Reference [30] and Figure 2 illustrates the fuzzy sets of *low, medium,* and *high* as described in Reference [31]. It can be seen from the two figures how, for instance, the *low* term of *speed* is differently defined in the two approaches. Note that in both approaches trapezoidal membership functions are used.

3.1.3 Fuzzy Logic Inference Systems—Overview and Main Architecture. Solving problems in FL involves a FL inference system or a FL controller, concepts introduced in Reference [32]. Note that these two terms, *FL inference system* or simply *FL system (FLS)* and *FL controller (FLC)* are basically defining the same concept. However, sometimes FLC refers to a specific type of FLS that is designed for control purposes. Several types of inferences were imposed as models and therefore they are also called fuzzy models. Among these, the one introduced by Mamdani and Assilian in 1975 [32] is the most popular. Other fuzzy models were introduced in time, such as the Sugeno fuzzy model [33], also known as the Takagi and Sugeno fuzzy model, the Tsukamoto fuzzy model [34], and the Larsen fuzzy model [35]. FLSs can have multiple (M) single (S) inputs (I) and multiple (M) single (S) outputs (O) in any combination (i.e., FLSs can be MIMO, MISO, SIMO, or SISO systems). MISO FLSs are the most common ones and therefore this type is further considered for exemplification.

The architecture of a classic MISO FLS is presented in Figure 3, but its components and their description (adapted to the number of inputs/outputs) are valid for all FLSs. In the next paragraphs, each of the components of an FLS is presented, indicating the particularities of each fuzzy model.

The *Fuzzifier* takes the inputs of the system in the form of crisp values and fuzzifies these values: for each of the crisp values returns the corresponding fuzzy degree of membership as output. This *fuzzification* process is based on the membership functions correspondent to each input and stored in the Data Base.

The *Inference Engine* is in charge of the actual reasoning process, mapping the fuzzified inputs of the system on the output fuzzy set based on the rules defined in the *Rule Base*. FL operators are applied on these rules. Different operators are applied depending on the type of fuzzy model that is followed by the FLS: Mamdani, Sugeno, Larsen, or Tsukamoto. Therefore, in the FL terminology we

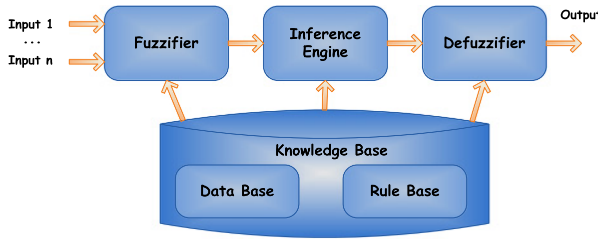

Fig. 3. Classic FLS Architecture.

talk for instance about Mamdani, Sugeno, Tsukamoto, and Larsen inferences, respectively. Detailed descriptions of the operators used in the inference can be found in References [27], [36], and [37].

Knowledge Base contains the Rule Base, a collection of "*IF-THEN*" rules expressed linguistically, and the fuzzy sets of inputs and outputs that are represented in the Data Base. An example of rule formalization is given in Equation (3)

$$\begin{aligned} &R^{(l)} : \boldsymbol{IF}\ u_1\ is\ F_1^l\ \boldsymbol{AND}\ u_2\ is\ F_2^l\ \boldsymbol{AND} \ldots \\ &u_p\ is\ F_p^l\ \boldsymbol{THEN}\ v\ is\ G^l, \end{aligned} \tag{3}$$

where l is the index associated to the rule in the context of the Rule Base, $u_{1\ldots p}$ and v are linguistic variables, p is the number of input variables considered in the rule, and ${F_{1\ldots p}}^l$ are fuzzy sets. G^l has different interpretations depending on the types of inference: it is a fuzzy set in Mamdani and Larsen inferences, a crisp function depending on the numerical values of $u_{1\ldots p}$, a singleton in the Sugeno inference, or a fuzzy set with a monotonic membership function in the Tsukomoto inference.

Example 3. In Reference [29], the authors have built an FLS for congestion detection. They linguistically expressed in their rules the dependency of the congestion on the speed of the vehicle and density of the vehicles around. Speed as described in Example 1 is a linguistic variable having as terms: *very slow, slow, medium,* and *fast.* Traffic density is a linguistic variable as well, having as terms *low, medium, high* and *very high.* An example of a rule is as follows: *If the speed is very slow and traffic density is medium, then the level of congestion is moderate.*

The *Defuzzifier* performs the *defuzzification,* the opposite process of fuzzification: a fuzzy set (resulted from the inference) is mapped to a crisp value, which is the FLS output. There are various defuzzification methods, including: centre of area (COA) also called centre of gravity or centroid, bisecter of area, weighted average, maximum, and mean of maximum. In the Sugeno and Tsukamoto fuzzy models, the defuzzification used is the weighted average method, while in Mamdani and Larsen the defuzzification method is not imposed by the fuzzy model applied, being decided when designing the FLS. The most popular defuzzification method is COA. Further discussions on defuzzification methods are included in References [27], [36], and [37].

3.2 Designing a Fuzzy Logic System in Vehicular Networks Context

This section outlines the steps that should be followed in the design of an FLS in tutorial manner. These steps are general and can be used for developing an FLS in any context, but there is specific particularization to VN context.

(1) *The first step* in designing an FLS is to define the inputs and outputs and their range: the linguistic variables representing the inputs and outputs are defined and their universe of discourse is identified. Moreover, the terms of the linguistic variables and their numerical range are also identified.

(2) *The second step* involves deciding the type of membership functions of the fuzzy sets describing the fuzzy terms of the inputs and outputs and in addition the setting of the parameter/coefficients of these functions. Initial setting of the parameters is based on *expert knowledge* of the designer or lessons learnt from the literature. Next, these parameters can be modified by performing either manual or automatic tuning.

(3) *The third step* in the design of an FLS requires defining the rules. This is a very important step for the performance of the system. Good rules can be defined based on the knowledge about how the system is supposed to work, its context and conditions. Similar to the case of the membership functions, initial rules should be based on *expert knowledge* or lessons learnt from the literature (Example 4). Further, manual tuning can be performed in order to adjust the rules.

Example 4. Example 3 presented a rule associated to an FLS designed to detect road congestion. The rule base for the same FLS was built based on the levels of vehicular traffic congestion estimated by Skycomp [38] that used data collected by aerial surveys of different freeways for this purpose.

(4) *The fourth step* involves defining the inference type and the defuzzification method and is often influenced by the second step. For instance, if the output terms are represented by singletons, then a Sugeno inference might be suitable, but not necessary. Also, if the fuzzy sets of the output are monotonic, we have an incentive to apply a Tsukomoto inference. Tsukomoto and Sugeno inferences come with their own defuzzification methods. In the case of Mamdani or Larsen inferences, selecting a defuzzification method is a design decision.

COA is the most popular defuzzification method. It is a bit more complex, but it is highly recommended as it is associated with higher accuracy. As in VNs, a high level of performance is often required, the complexity is an issue, and consequently a trade-off between reduced computation complexity and performance/accuracy should be made when designing a VN-based FLS. For instance, the membership functions employed should be less computationally complex (e.g., singleton, triangular, trapezoidal functions) in order to compensate for the increased complexity of the COA defuzzifier.

(5) *The last step* in the design of an FLS is about assessing and tuning of the system. Tuning the system refers to reviewing the range of the inputs/outputs and their terms, revising the fuzzy sets, tuning the membership functions (i.e., revising their parameters or shape), tuning the rules (i.e., adding, removing, assigning/modifying their weights), and experimenting with different types of inferences.

Most of the actions performed in tuning an FLS are done manually. However, for tuning the membership functions and the rules, automatic techniques were designed. These techniques are based on learning algorithms. In FLS tuning, two types of learning are used: supervised and reinforcement-based learning. In both cases, there is a variety of learning techniques that can be used: classic learning techniques (e.g., Q learning), techniques borrowed from the fields of Neural Networks and Genetic Algorithms, Particle Swarm Optimization, and H ∞ filtering.

In *automatic tuning based on supervised learning,* there is a need for a so-called *training set.* This is represented by numerical data for inputs and their corresponding outputs. In some fields where FL is applied, the FLS is basically created starting from such a data set. However, in the context of VNs, obtaining this training data set is not straightforward. One way to obtain it is via simulations, considering for certain inputs what the best output that can be obtained is. The final system evaluation should consider a different simulation scenario than the one used for obtaining the training set. The supervised-based learning is performed off-line, before the deployment of the FLS.

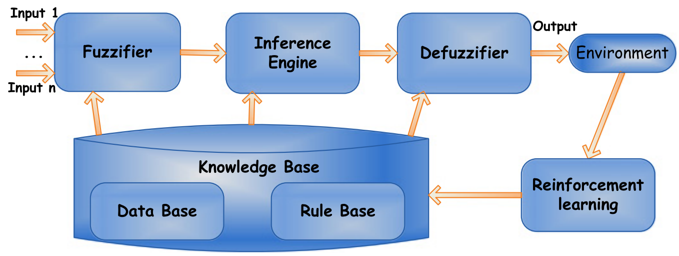

Fig. 4. Reinforcement learning-based real-time adaptive FLS architecture.

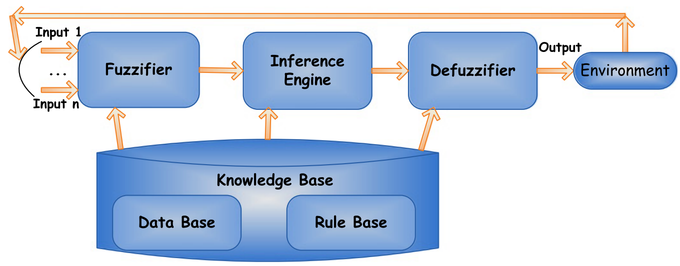

Fig. 5. Simple real-time adaptive FLS architecture.

In *automatic tuning based on reinforcement learning*, there is no need for a training set. The FLS is able to adapt its parameters dynamically, at run-time based on the output of the system and the impact of the output on the environment (e.g., process controlled, impact of the decision taken on the modeled problem, etc.).

3.2.1 Alternative FLS Architectures. An automatic tuning based on reinforcement learning brings slight changes to the classic architecture of an FLS (Figure 3) as the adaptation is based on run-time recursion. A reinforcement learning block is added to the classic architectural blocks that were discussed in Section 3.1.3). This implements the learning technique that is employed at run-time on the input(s) that are the results of the impact of the FLS output on the environment. The output(s) of this block is fed into the Knowledge Base. This architecture is presented in Figure 4.

There is also another type of real-time adaptive architecture used in the FLS design, which is presented in Figure 5. The adaptive mechanism is very simple and is not based on any kind of learning algorithms, but on the impact of the output on the environment only. This is a typical architecture used by FL controllers (i.e., FLSs designed for control). These three architectures illustrated in Figures 3, 4, and 5, respectively, are the main architectures for FLSs.

3.3 Fuzzy Logic Solutions in Vehicular Networks—Solutions

After surveying the FL solutions proposed in VN context, we propose a generic classification of these solutions in three main classes based on the process FL is mostly employed as follows: decision-making solutions, control solutions, and detection and prediction solutions, respectively. By far, the most popular class includes FL-based decision-making solutions, while the class which encompasses FL-based detection and prediction solutions is the least popular with few solutions only. This generic classification can be further granulized considering the architecture type adopted by the FLS employed in the outlined processes. The use of the classic FLS architecture leads to non-adaptive solutions, while using the other two architectures, reinforcement learning-based real-time adaptive architecture and simple real-time adaptive architecture, results in adaptive solutions.

Further on, we focus on solution classification based on FL areas of applicability in VN context. FL-based decision-making solutions in VNs include: routing protocols, clustering algorithms, handover schemes, and data aggregation mechanisms. Two major directions are identified within FL-based control solutions in VNs: MAC protocols and driving automation. Other FL-based control solutions include video transmission over VNs and beacon-rate control solutions. FL-based detection and prediction solutions involve routing protocols, clustering algorithms, highway tolling schemes, and road congestion detection mechanisms. Based on these considerations, a detailed taxonomy of the FL-based solutions in VNs is presented in Figure 6.

The next three sections present detailed analysis of the solutions that subscribe to each of the three main classes of FL solutions in VNs. The solution presentation follows the taxonomy presented. In the analysis of the solutions, a great emphasis is made on the VN-related aspects that FL is employed to address, why FL is suitable for addressing these aspects and what are the benefits of a FL approach in comparison to other approaches. The design decisions that relate to the FL system are also highlighted for the solutions as important lessons can be learnt for the further development of FL systems in VN context.

4 FUZZY LOGIC-BASED DECISION-MAKING SOLUTIONS IN VEHICULAR NETWORKS

4.1 Fuzzy Logic Solutions in VN Routing

4.1.1 Why Using Fuzzy Logic? In the literature, there are several surveys describing routing protocols in VNs [8, 9]. A routing protocol defines the data exchange between two entities that communicates in a network and usually includes network path/route selection, data forwarding, and route maintenance or recovering from route/link failure [8]. It is obvious that route selection involves a decision-making process. Less obvious is the fact that, in the context of data forwarding, we can also have a decision-making process, especially in the particular case of the broadcast protocols that use relay nodes for data forwarding. Relay nodes selection, which is a decision-making process, has a decisive role in protocol performance.

Routing is listed as a challenge in VNs because of their specific characteristics which include: rapid change in topology, relative high speed of nodes, dynamic information exchange, and frequent disconnections. Consequently, it is of great importance to consider these characteristics in VN routing protocols. A routing protocol in VNs considers multiple parameters in the decision-making process for either route or relay node selection. However, so far, there are no deterministic models that describe with precision the influence of these parameters in routing. As FL is known as an excellent mathematical framework for handling multiple parameters and dealing with non-deterministic problems, it is a perfect candidate to solving routing challenges in VNs.

4.1.2 State-Of-the-Art. Wang et al. [39] proposed the Fuzzy control-based AODV routing (Fcar), which enhances the performance of the classical *ad hoc* on-demand distance vector routing (AODV) by taking into consideration VNs specific criteria. Simulations performed show that Fcar improves the routing performance in comparison with AODV in a VN context. Several criteria are incorporated into the input parameters of the FLS designed for route selection: percentage of same-directional vehicles and route lifetime. Route lifetime is computed based on vehicle's speed and distance between vehicles related to the effective communication range between vehicles. The FLS described is very flexible and allows for multiple inputs which lead to multiple path/route selection criteria. This aspect was considered by the authors when choosing to employ FL in their solution. The best route is indicated by the output parameter of the FLS which is the selection

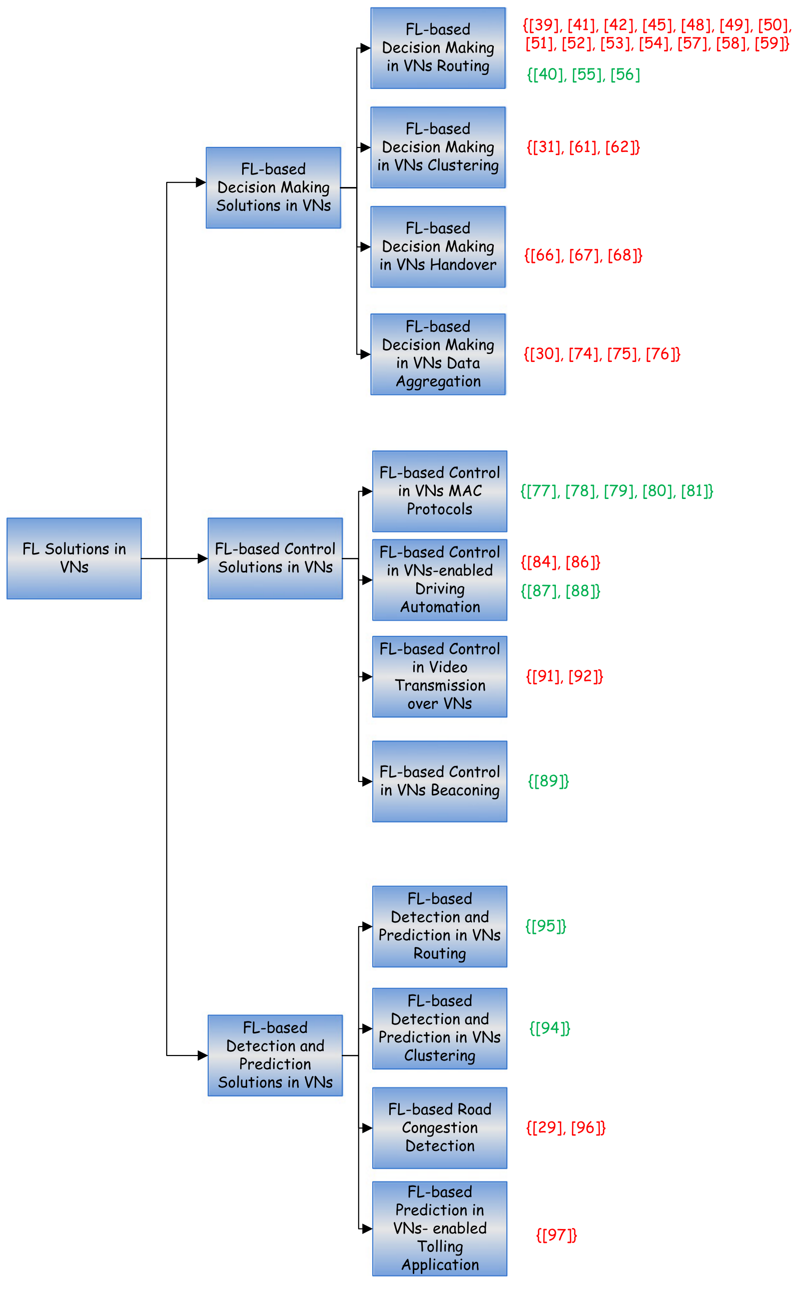

Fig. 6. FL Solutions in Vehicular Networks—Taxonomy.

probability. FLS has trapezoidal membership functions, the authors motivating their decision based on the fact that these are simple and computational efficient. In the design of the rule base, manual tuning was adopted.

In Reference [40], it was proposed a FL-based routing protocol aiming to comply with the demanding Quality of Service (QoS) requirements of real-time traffic over VNs. The proposed protocol is derived again from AODV that provides some basic mechanisms to allow nodes for the specification of QoS parameters. VN-specific criteria are taken into consideration by the designed protocol in order to address the rapidly changing topology. These parameters were incorporated into the inputs of an FLS that determines as output the suitability of the vehicle to be an intermediate node in the route. Namely, the inputs are: minimum bandwidth for the real-time traffic (R)/length of the queue for the non-real-time traffic (Q), the foreseen connection time between the vehicle and its neighbors (L), and the currently used bandwidth (B). The VN-specific criteria are incorporated into the L input which is dependent on vehicle's speed, position and vehicle's mobility model. The other two inputs are considered in such way to avoid congestion and to provide load balancing. In this way, the protocol proposes to satisfy the stringent QoS requirements in VNs, especially in the case of real-time traffic [1]. In the design of the FLS, triangular functions are the authors' option for the membership functions. The membership functions are automatically tuned using H-infinity filtering in order to adapt to the volatile characteristics of VNs. Thus, the FLS designed is adapting in real-time and subscribes to the generalized architecture illustrated in Figure 4. The authors experimented with the neural networks (NN)-based tuning and genetic algorithms (GA)-based tuning as well, but the performance tests have shown that, while the accuracy is comparable (i.e., NN is performing slightly better, while GA visibly worse), the learning time is much longer for these latter techniques. In VNs, due to the ongoing network topology changes, frequent training is required, and so NN- and GA-based learning techniques may not to be suitable. However, this last statement cannot be generalized, as not many details are provided about the specific learning algorithms implemented and multiple NN- and GA-based learning algorithms can be implemented. The rule base of FLS resulted from *expert knowledge* on the deep understanding of the influences of each parameter in the network. The Inference Engine is based on the Tsukamoto inference and consequently, the defuzzification process is also Tsukamoto.

Wu et al. [41, 42] proposed FL- and Q-learning- [43] based approaches for unicast routing. Both approaches derive from the AODV protocol. The routing protocol proposed in Reference [41] is called Fuzzy Q-Learning AODV-based protocol (FQLAODV), while the one proposed in Reference [42] is called Portable Fuzzy Q-Learning AODV-based protocol (PFQ-AODV). Both protocols have the same basic principle: they use an FLS for evaluating the link that is possible to be used in the routing path and based on the ranking provided by the FLS, Q-learning is applied for selecting a route that ensures multi-hop reliability and efficiency. The differences between FQLAODV and PFQ-AODV are in the inputs considered for the FLS and in the fact that the last one does not assume the existence of any GPS or other positioning system, making it portable and more practical. The FLS in FQLAODV has as inputs: bandwidth, mobility factor, and received signal strength indicator. PFQ-AODV is refining more its inputs, considering again bandwidth, mobility factor and a new input link quality. Mobility factor is computed different than in FQLAODV, as it does not make the assumption that the vehicles know their positions. Link quality is a more complex input than received signal strength taking into consideration in its computation a network metric, packet loss, and a topology metric- number of neighbors. In the design of both FLSs, the same decisions are applied: triangular and trapezoidal membership functions for their efficiency, Mamdani inference type, and COA defuzzification method. The membership function parameter selection is entirely based on the authors' experience and knowledge, thus *expert knowledge*, although the authors do state that automatic tuning is possible and brings an advantage to FLS in VNs—adaptability to any kind of conditions (e.g., sparse or dense network) in comparison to other solutions. Moreover, the benefits of employing FL in the proposed routing protocols are demonstrated against AODV and QLAODV [44]—AODV modified with Q-learning via both simulations and real testbed. In PFQ-AODV testing, real-world experiments were performed. However, no comparison is performed between the two FL-based approaches.

The FL-based routing protocol proposed in Reference [45] follows AODV principles as well, but also emphasizes the fact that it is not enough to consider only the high mobility of VN nodes, but also MAC layer characteristics, such as transmission rate at MAC layer. This observation is based on the analysis of the data collected via real-world experiments. Consequently, an FLS having as inputs transmission rate and link quality that considers vehicle mobility and signal stability is employed in route selection together with a Q-learning algorithm. The transmission rate is estimated based on the hello reception ratio. Q-learning and Transfer Learning are employed in this estimation. The output of the FLS is the rank of the path/link between the current vehicle and the vehicle sending the hello message. Based on the ranks provided by the FLS, a Q-learning-based component is performing the route selection. The approach is thoroughly tested via both simulations and real-world experiments.

The solution is demonstrated to perform better against AODV enhanced to consider MAC layer characteristics (i.e., AODV with ETX) [46] and a hybrid routing protocol for VNs that combines this enhanced AODV with greedy forwarding geographic routing [47]. The design decisions related to FLS are detailed in Table 1.

Similar to the previous approach, the following routing solutions also consider MAC layer characteristics. These routing protocols aim at reducing MAC layer contention by using backbone nodes for data forwarding. The performance assessment of these proposed approaches has shown a clear reduction of the number of packets transmitted (i.e., up to 25% in the case of the protocol proposed in Reference [48]) which results in a reduction in MAC layer contention time. In References [49] and [50], FL is employed in selecting backbone vehicles in order to create a reliable connected network for data forwarding. An FLS deployed in each vehicle is designed having as inputs antenna height factor and VN-specific criteria: the vehicle velocity and the number of vehicles traveling in the same direction. The output is the rank that characterizes the suitability of the vehicle to be a backbone node. The FLS structure is detailed: follows the Mamdani inference model, has triangular membership functions, the rule base is fully presented and uses COA as defuzzification method. The authors choose to use FL because it allows for a flexible design: the variability of VN environment from road segment to road segment makes it very difficult to create simple mathematical models to describe the relationship between the aforementioned metrics considered as inputs and the output. The FLS employed in these two solutions has been enhanced in the routing protocols proposed in References [48] and [51]. The antenna height input was replaced with a new channel condition-related parameter that is estimated based on exchange of HELLO messages.

Khokhar et al. [52] proposed a routing protocol dedicated to urban vehicular environment that employs novel concepts in order to address security challenges in VNs. The novel aspect is the rationale behind the routing protocol. The authors state that there are some social behavior patterns developed in urban environments and these should be exploited in order to make secure routing decisions. A friendship mechanism is developed that is used in taking routing decisions at intersections. The decision making is based on an FLS that has the following inputs: friends, friends of friends, and non-friends and gives as output the path fuzzy cost. The list of friends of a vehicle is made based on the social behavior pattern developed in traffic: if there was a V2V communication between two vehicles, there is a friend relationship between them. In addition, the social networks of the drivers are considered. For instance, if the drivers are friends on Facebook, their vehicles are friends as well. Regarding the FLS design decisions, the authors opted for triangular

Table 1. Summary of FL-Based Decision-Making Solutions in VNs Routing Protocols

		FLS design decisions						
Why using FL?	**Objective**	**μ functions type/definition method**	**Rule base building method**	**Inference Type**	**Defuzz.**	**FLS Arch.**	**Ref.**	**Testing**
Decision making based on multiple parameters; dealing with imprecision and non-deterministic problems.	Route/path selection	Trapezoidal functions/ n.a.	Manual tuning	-	-	Classic FLS => non-adaptive type	[39]	Simulations
	Route/path selection	Triangular functions/ Automatic tuning—H∞ filtering technique	Expert knowledge	Tsukamoto	Tsukamoto-specific	Reinforcement learning-based real-time adaptive FLS	[40]	Simulation
	Route/path selection	Triangular functions/ Manual tuning	Expert knowledge	-	COA	Classic FLS => non-adaptive type	[52]	Simulations
	Route/path selection	Triangular and trapezoidal functions/Expert knowledge	Expert knowledge	Mamdani	COA	Classic FLS => non-adaptive type	[41]	Simulations
The reason above + FLS design is transparent, easily tunable, adaptable to the variability of VN conditions	Route/path selection	Triangular and trapezoidal functions/Expert knowledge	Expert knowledge	Mamdani	COA	Classic FLS => non-adaptive type	[42]	Simulations and real-world experiments
	Route/path selection	Triangular functions/*Expert knowledge*	Expert knowledge	Mamdani	COA	Classic FLS => non-adaptive type	[45]	Simulations and real-world experiments
FL allows for a flexible design	Backbone node selection	Triangular functions/*Expert knowledge*	Expert knowledge	Mamdani	COA	Classic FLS => non-adaptive type	[49] [50]	Simulations
	Backbone node selection	Triangular functions/*Expert knowledge*	Expert knowledge	Mamdani	COA	Classic FLS => non-adaptive type	[48] [51]	Simulations
Dealing with imprecision; design flexibility, the system being easily tunable	Relay nodes selection	Triangular and trapezoidal functions/n.a.	-	Mamdani	COA	Classic FLS => non-adaptive type	[53] [54]	Simulations
	Relay nodes selection	Triangular and trapezoidal functions/Automatic tuning—Q learning and Transfer Learning	-	Mamdani	COA	Reinforcement learning-based real-time adaptive FLS	[55]	Simulations
Efficiency in dealing with multiple parameters in real-time	Relay nodes selection	Triangular and trapezoidal functions/Automatic tuning—Q learning	-	Mamdani	COA	Reinforcement learning-based real-time adaptive FLS	[56]	Simulations and real-world experiments
Dealing with multiple parameters, imprecision and non-deterministic problems	Relay nodes selection	Triangular functions/n.a.	-	Tsukamoto	Tsukamoto-specific	Classic FLS => non-adaptive type	[57]	Simulations
FL able to solve problems in fast ever-changing environment	Decide whether to re-broadcast or not	Triangular and trapezoidal functions/n.a.	-	Mamdani	COA	Classic FLS=> non-adaptive type	[58]	Simulations
		Trapezoidal functions/n.a.	-	Mamdani	COA	Classic FLS=> non-adaptive type	[59]	Simulations

functions as they are computationally efficient. Membership function parameters and rule base were established based on *expert knowledge* and manual tuning. The COA defuzzification method is the authors' option for defuzzification.

Wu et al. [53] introduced a FL decisional system to select the nodes where to relay the broadcast messages in the context of a new broadcast protocol for VNs. The technique of using only a few neighboring nodes for relaying broadcast messages ensures the efficiency of the proposed broadcast protocol. The FLS uses multiple parameters in the relay node selection which ensures high reliability of the protocol. These parameters are distance factor (inter-vehicle distance), mobility factor that considers both the current and future position and received signal strength and are used as inputs by the FLS which outputs the rank of a node. The node with the highest rank is selected as a relay node. All the FLS design decisions are presented in Table 1. The same FLS was employed in the broadcast protocol proposed in Reference [54] in the relay nodes selection, but the overall solution was enhanced with network coding in order to improve the packet reception ratio. In both approaches, the fact that FL is employed due to its capability of dealing with imprecision and its design flexibility is highlighted; the system being easily tunable.

Another broadcast protocol was proposed in Reference [55], which employs for relay node selection the same FLS used in References [53] and [54] enhanced with a real-time Q-learning-based tuning of the membership function parameters. Consequently, the new FLS has a reinforcement learning-based real-time adaptive architecture as shown in Figure 4. In order to speed up the learning time, the authors also employed a Transfer Learning technique that allows one vehicle node to make use of the lessons learnt by another vehicle node. The testing shows how the protocol based on the FLS with online automatic tuning clearly outperforms the protocol based on the classic FLS. The automatic tuning allows for a better adaptation of the protocol to the ever-changing VN environment. The tests were performed using different communication ranges. The authors do not make any specific comments regarding learning time; however, from the results presented, it is clear that this does not have any impact on the performance of the proposed protocol.

In Reference [56], another FLS with a reinforcement learning-based real-time adaptive architecture is employed for the relay nodes selection in a broadcasting protocol. Q-learning is again used for automatic tuning of the membership functions parameters. As a novelty, the authors consider the packet size in relay node selection, as it was demonstrated in the literature that this parameter can have a significant impact on the packet loss rate. This parameter is incorporated in what is called the *link quality metric*, considered as an input for the FLS together with distance and a mobility metric. Moreover, an extra step is considered in the decision-making process of the relay nodes. The FLS gives the rank for the nodes, but the selection is performed based on both rank and redundancy level using a heuristic approach. The purpose of jointly considering the relay node selection with the redundancy level is to improve packet forwarding probability in order to eliminate retransmissions. The proposed approach is tested both via simulations and real-world experiments. The design decisions related to FLS are listed in Table 1.

FL is also employed in the relay node selection described in Reference [57]. Two FLSs are employed in the decision making. The first FLS is used to decide upon the most appropriate vehicle from a list of candidates to become the relay node that is storing and forwarding the data to the requesting nodes. A second FLS is designed to decide from these requesting nodes if they are suitable to become relay nodes. The first FLS has as inputs bandwidth, overhead, and lifetime, the latter incorporating the VN-specific parameters: vehicle velocity, distance between vehicles, and direction of vehicles. The output is represented by the appropriateness of the node to be a relay node. The membership functions are triangular and the FLS has a Tsukamaoto inference type and defuzzification method. The design decisions of the FLSs are very poorly described; none of the design decisions are motivated. Regarding the second FLS, no details are provided except for the inputs that are represented by the capabilities of the requesting node such as computation capability, buffer size and stability of the signal strength, and the output which is the suitability of the node to become a relay.

Very recently, FL decisional systems were employed to help the node take the decision if to re-broadcast or not in the context of some receiver-based routing protocols. Such solutions are presented in References [58] and [59]. In Reference [58], each vehicle node has an FLS that uses as inputs: coverage (i.e., factor computed based on the distances to potential forwarder vehicles), connectivity (i.e., number of vehicle's neighbors) and mobility factors (i.e. factor computed based on the vehicle's speed) in order to decide where the node should re-broadcast or not. FL is employed due to its suitability in solving problems in a fast ever-changing environment that characterizes VN. Another argument in favor of FL is the success stories in the context of other VN solutions. A similar solution is proposed in Reference [59], where an FLS is used in deciding if a vehicle-node is suitable to re-broadcast or not. The inputs considered for the FLS are coverage and mobility factors, computed similarly as in Reference [58]. A set of re-broadcasting candidates is formed based on the suitability decided by the FLS and the node decides to re-broadcast if its coverage factor is the highest among the vehicles in the set. Again, the motivation for using FL stresses out its capability of solving problems in a fast ever-changing and uncertain environment. Moreover, in Reference [59], the fact that FL has improved the decision-making process in the general context of VN and has reduced delays in computation is emphasised. The details of the FLS design decision for both solutions are presented in Table 1.

4.2 Fuzzy Logic Solutions in VN Clustering

4.2.1 Why Using Fuzzy Logic? In networking, clustering techniques are used to partition the network by forming virtual and temporary groups of nodes (i.e., clusters) leading to a network with better manageability and improved performance. It can be said that clustering helps solve some of the main issues in VNs: scalability and stability [60]; therefore, it was widely adopted in VNs. Clustering algorithms were implemented in the design of a large variety of VN solutions: MAC protocols, routing protocols, data aggregation, security protocols, inter-vehicle communication, and data and infotainment dissemination solutions and architectures. In addition, various generic clustering algorithms were defined for VNs [11, 12].

A clustering algorithm considers that a node can be in one the following main states based on node membership and the task associated to the node: *unclustered* (i.e., non-clustered or independent, when the node does not belong to any cluster), *cluster member* (i.e., clustered, the node is within a cluster), gateway node (GW) (the node that ensures information exchange and relay with the neighboring clusters) and *cluster head* (CH) (i.e., the node has extra responsibilities in a cluster). Usually, CH is the main controller of the cluster, the main coordinator of the communication within the cluster (i.e., intra-cluster communication), and has a major role in the functionality that is supposed to be provided by the cluster.

In VNs, due to the ever-changing topology, some secondary node states are also considered including: candidate node and CH backup or CH candidate (quasi-CH) states. The candidate state was introduced by some approaches in order to obtain a better stability of the cluster. A node is not immediately given a cluster member state; it goes into the candidate state until it proves that it has certain stability in the cluster. The CH backup or CH candidate states were introduced to make the process of changing the CH faster and smoother.

Basically, a clustering algorithm involves decision-making processes that select the appropriate state of the node based on some clustering metrics. Most complex decision-making processes are the ones employed for CH selection and then for GW selection as these involve multiple clustering metrics reflecting the complex VN environment. A stable CH is mandatory for obtaining a stable cluster, as usually vehicles organize themselves around a CH in order to form a cluster. Therefore, most of the clustering algorithms consider VN-specific metrics that describe the high mobility, rapidly changing topology, and diversity of conditions (e.g., direction, vehicle's relative speed in comparison to other neighboring vehicles, vehicle's relative position, traffic flow, lane in urban scenarios, density of vehicles, etc.) and for the clustering algorithms designed for a specific problem/application/service, the corresponding metrics. Beside the fact that clustering algorithms need to consider multiple parameters, another issue is that it is impossible to define precisely how each of the clustering metrics influences the stability of CH or GW in particular and clusters in general. This is exactly the context FL is applied in so many domains successfully: to model and solve non-deterministic problems that need to consider multiple factors. Next, it is discussed how FL was employed in the literature in the context of VN clustering.

4.2.2 State-of-the-Art. FL is used in Reference [31] for selecting the most appropriate cluster heads in a cluster-based VN architecture. The option for FL is motivated based on the fact that it is an excellent mathematical framework for dealing with imprecision and multiple parameters. The inputs of the designed FLS are: average relative distance, average relative velocity, direction of traveling and average relative compatibility. The last parameter measures the compatibility in the users' (vehicles' drivers/passengers) preferences in certain data/content. The aim is to increase the probability of users being provided with data/content of their interest inside the cluster. The FLS output is called *cluster head eligibility*, a rank on the basis of which the most appropriate vehicle in a cluster becomes cluster head. The inputs and the output of the FLS have triangular and trapezoidal membership functions chosen because of their efficiency. The inference type is Mamdani, while the defuzzification method is COA.

In Reference [61], a new cluster-based vehicular cloud architecture is proposed for a better management of the limited resources in VNs. The cluster head selection algorithm is based on FL and it is proven to have better performance as compared to the previously described FL-based cluster head selection algorithm (i.e., [31]). FL is chosen due to its flexibility and adaptability to the dynamic VN environment. The inputs for the FLS that has a classic architecture are: neighborhood degree, average speed, and Road Side Unit (RSU) link quality and the output is the fit factor for a vehicle to be a cluster head. Trapezoidal and membership functions are used due to their reduced complexity. Their parameters and the rule base are chosen such as it subscribes to the following policy: a cluster head should have a high neighborhood degree and the RSU link quality should also be high.

A newer trend in VN clustering is the employment of clustering algorithms in designing reliable and efficient VN architectures that bring together multiple access technologies [11]. Such a cluster-based hybrid architecture is proposed in Reference [62]: the vehicles in the cluster communicate via V2V communications based on the IEEE 802.11p standard, while a GW in the cluster is chosen for connection to the LTE Advanced infrastructure. GW selection is based on an FL decision-making system. The selection takes into consideration multiple criteria, the decisional FLS having the following inputs: QoS classes, which are introduced as novelty element in comparison with other schemes, connectivity strength between the vehicle and infrastructure, connectivity strength between the CH and infrastructure, CH load and link connectivity between the vehicle and CH. The latter parameter considers the mobility. The FLS has a classic architecture with a Mamdani inference. It uses COA for defuzzification and trapezoidal membership functions. The knowledge base is built based on *expert knowledge* and careful analysis on the influence of the inputs on the output performed on extensive simulation results. The choice for a FL-based decisional system is based on the FL inherent strength of dealing with imprecision that characterizes the vehicular network

Table 2. Summary of the other FL-Based Decision-Making Solutions in VNs

Subclass	Why using FL?	Objective	FLS design decisions: μ functions type/definition method	FLS design decisions: Rule base building method	FLS design decisions: Inference Type	FLS design decisions: Defuzz.	FLS design decisions: FLS Arch.	Ref.	Testing
FL-based Decision-making Solutions in VNs Clustering	Dealing with imprecision; modeling linguistic information (human-like reasoning)	Cluster head selection	Triangular and trapezoidal functions/Manual tuning	Lessons learnt from the literature	Mamdani	COA	Classic FLS ⇒ non-adaptive type	[31]	Simulations
	FL allows for a flexible design	Cluster head selection	Triangular and trapezoidal functions	Lessons learnt from the literature	Mamdani	COA	Classic FLS ⇒ non-adaptive type	[61]	Simulations
	Dealing with imprecision	Gateway selection	Trapezoidal functions	Expert knowledge	Mamdani	COA	Classic FLS ⇒ non-adaptive type	[62]	Simulations
FL-based Decision-making Solutions in VNs Handover	Decision-making based on multiple parameters that describe the environment with a degree of imprecision	network selection	-	-	-	-	-	[66]	Simulations
		network selection	-	-	-	-	Classic FLS ⇒ non-adaptive type	[67]	Simulations
		network selection	Triangular/ expert knowledge	Expert knowledge	-	-	Classic FLS ⇒ non-adaptive type	[68]	Simulations
FL-based Decision-making Solutions in VNs Data Aggregation	FL is able to take decisions based on multiple criteria; An FL-based design allows for flexibility and extensibility	Deciding upon the data similarity	Triangular functions;Parameters' selection method not specified	-	Sugeno	Sugeno-specific	Classic FLS ⇒ non-adaptive type	[74]	Simulations
		Selection of most relevant data from the aggregates to be further disseminated	-	-	-	-	-		
		Deciding upon the data similarity	Trapezoidal functions;Parameters' selection method not specified	-	-	-	Classic FLS ⇒ non-adaptive type	[30], [75]	Simulations
		Deciding upon the data trustfulness	Trapezoidal functions;Parameters' selection method not specified	-	-	-	Classic FLS ⇒ non-adaptive type	[76]	Simulations

environment and the relationship between the clustering metrics considered and the suitability of a vehicle node to become GW.

4.3 Fuzzy Logic Solutions in VN Handover

4.3.1 Why Using Fuzzy Logic? Among the architectures presented for VNs, the hybrid architecture that supports vehicular heterogeneous networking is the most promising. One of the underlined challenges of vehicular heterogeneous networks is the handover (HO) [63]. HO in vehicular heterogeneous networks subscribes to the general HO problem in heterogeneous networks, but also due to the VN characteristics, needs specific approaches. An HO process has three phases: monitoring (i.e., collecting the information related to network conditions based on which the HO decision phase is triggered), HO decision (i.e., selecting the most suitable access network—network selection—and deciding whether to switch to this network), and HO execution (connecting to the pre-selected network). HO decision phase has an overwhelming importance in the HO process as its performance highly depends on how the targeted network is selected in order to secure the best communication performance possible.

An effective network selection in wireless heterogeneous networks takes into account multiple criteria including: network metrics, device-related metrics, application requirements, and user preferences [17]. Deciding accurately and often in real time, the influence level these parameters have on the degree of electability of a network is impossible. Therefore, FL, known for its capability of dealing with imprecision and multiple parameters and for its suitability for real-time systems, provides a robust framework for HO decision.

A considerable number of HO solutions for heterogeneous networks have employed FL in the network-selection process [64, 65]. This section focuses on the vehicular heterogeneous networks that require dedicated solutions that take into account their specific characteristics. Consequently, besides considering network and device-related metrics, application requirements and user preferences, a good network selection mechanism in VNs should take into account VN-specific characteristics in the decision-making process. FL-based network selection solutions dedicated to vehicular heterogeneous networks are further discussed.

4.3.2 State-of-the-Art. A general framework for network selection transparency in VNs is proposed in Reference [66]. This framework considers multiple decision criteria from each of the following classes: network and device-related metrics, user preferences and application requirements. However, no VN specific criteria are used. For the decision-making process, the authors propose FL as a math model. The authors provide only architectural details of the framework, where decision-making is one of the architectural blocks, leaving the implementation details out of the picture.

Ma and Liao [67] proposed a speed adaptive HO algorithm for vehicular heterogeneous networks based on FL. The HO decision is based on a FL decisional system that decides the best available network and whether HO should be performed. The input parameters of the system are a combination of network metrics and application requirements: bandwidth capacity, power charge, received signal strength (RSS) and delay. No VN-specific characteristics are directly considered in the inputs of FLS. These are incorporated in the speed-adaptive strategy that considers the high mobility of the vehicles. This strategy is applied in order to form the list of candidate networks that are then ranked by the FLS. The membership functions chosen for the designed FLS were triangular and trapezoidal based on the fact that these are known for their good performance especially in real-time systems. Except for specifying the type of the membership functions, their parameters are not specified either, no other design decisions are revealed about the FLS. The efficiency of the solution proposed is demonstrated against a classic FL-based solution for heterogeneous networks that does not take into consideration any VN-specific characteristics.

FL is also employed in network selection in Reference [68]. Multiple parameters are considered, including VN-specific characteristics, application requirements, cost and network metrics. In this approach, VN-specific criteria are considered directly in the FL-based decision-making process. The VN-specific characteristics considered are the vehicle speed and vehicular density around a vehicle. On the basis of these parameters plus RSSI, an FLS decides on the network metrics: throughput, latency, and packet loss. The latter are considered as inputs for another FLS together with application requirements. This FLS gives as output the weight of the network that is used as input together with the cost in the last FLS that is selecting the network. *Expert knowledge* and deep analysis on the influence that inputs have on the outputs are at the basis of defining the knowledge base. No other details related to FLS structure and design are presented.

4.4 Fuzzy Logic Solutions in VN Data Aggregation

4.4.1 Why Using Fuzzy Logic? This section presents the concept of data aggregation in VN context, its challenges, and discusses why FL is suitable to be employed in addressing some of these challenges.

Data aggregation can be the answer to some of the major challenges in VNs. For instance, one of the major challenges in VNs is the efficient usage of the available bandwidth [69–71]. Data aggregation can be employed to address this challenge in the context of data collection. Data aggregation is used to combine correlated information from different nodes before redistributing the information in the network. As such, the dissemination of similar information in the network will be avoided. The data aggregation process consists of the following functional components: decision, fusion, storage and dissemination [69, 70]. In the decision component, the data is analyzed to see whether there is any correlation between atomic data items and a decision is taken accordingly. If the decision component detects a correlation, then the data is fused (i.e., fusion component). Data, fused or not, depending on the correlation level, is disseminated in the network (i.e., dissemination component). Regarding the storage component, this is placed either before decision phase, in which case it stores all the collected data, or after the fusion, in which case it stores the aggregated data before disseminating it in the network [69].

Data aggregation schemes in VNs have several limitations and challenges. Among the main challenges is the flexibility in the criteria used to decide upon the data similarity and security [70]. The lack of flexibility in the criteria of similarity between data is a very common problem of data aggregation schemes in VNs. For instance, some of these schemes correlate the data based on fixed or structured segmentation of the road, which makes them dependent on these structures. An FLS has a flexible design, not only that allows for tuning the system, but also considers multiple criteria and enables adding easily new inputs to the system. Therefore, FL is a highly suitable technology in deciding the correlation between data.

The other highlighted challenge that arises in VN data aggregation is the security: it is more difficult to decide if an aggregated data can be trustful. In order to address this challenge, some data aggregation solutions rely on a tamper-proof service in each vehicle that randomly requests for integrity proofs. However, this service can be easily bypassed by attackers. Other approaches employ security mechanisms that are dependent on node reputation or on some fixed structures. Security schemes based on node reputations are very hard to employ in the self-organized networks such as VNs [72], while the security schemes based on fixed structures are demonstrated to have scalability issues [73]. Moreover, in the case of structure-based trust mechanisms implemented in data aggregation solutions, more or less of the same disadvantages are met as in the case of the data aggregation approaches that have their decision component dependent on such structures. There, the concern was the lack of flexibility in the correlation criteria; here, the issue is the lack of flexibility in the trust criteria. An FL-based design can address the flexibility issues in the context of trust criteria as well.

4.4.2 State-of-the-Art. FL was employed in the decision-making process not only in the decision phase of the proposed data aggregation schemes, but also in two other phases, fusion and storage, as it can be seen in the next paragraphs.

Caballero-Gil et al. [74] proposed a data aggregation solution that employs FL in the decision component. The FL decisional system is exemplified with two inputs: space—the approximate location the data pertains to, and time—the data lifetime. The output is represented by the correlation between the pieces of data to be aggregated. It has two possible values: YES and NO. The solution is open to extension and generalization: other inputs can be easily considered. In the given example, the membership functions used for inputs are triangular, while the output has a singleton function. The inference, as it appears to be from the rules description, is of Sugeno type. However, the focus is not on the FLS design, but instead of the benefits of using FL in the decision component of a data aggregation scheme: flexibility and extensibility in the set of criteria used for correlating the information for aggregation.

This approach also proposes an FL-based selection scheme to be implemented in the storage component that is placed in this solution immediately after data collection. This scheme aims to select the most relevant data items for aggregation in order to avoid the overloading of communication channel that could lead to restrictions in data that is sent in the network to the vehicles. However, the solution is not detailed, some inputs that could be considered in the selection are named only (e.g., severity or antiquity of data).

Dietzel et al. [30, 75] proposed an FL-based decision component in their data aggregation approach that does a step forward in showing the flexibility, extensibility, and generality that can be achieved via FL. The output parameter of the FL decisional system is the same as in the aforementioned approach, but the input parameters are generalized and called influences: *Influence 1*, ..., *Influence n*. This is as these parameters represent the influence on deciding upon data similarity. Examples of influences are speed difference [30, 75], and location difference between vehicles [75]. In exemplifying the fuzzification of the inputs, the authors use trapezoidal membership functions for the speed difference input. However, the focus is again not on the internal design decisions of the FL decisional system, but on its generalization. As already emphasized, all the proposed FL-based decision components ensure extensibility and flexibility in the set of criteria employed for correlating the data to be aggregated. These characteristics lead to structure-free and dynamic aggregation approaches unlike the others data aggregation approaches.

Dietzel et al. [76] proposed an enhanced FL-based data-centric solution in order to address the security challenge in VN data aggregation. A selective attestation process is employed based on a probabilistic scheme that results in clues leading to trust in the correctness of an aggregate. An FLS is designed in the fusion component having as inputs the aforementioned clues and as output the *Trust* in the {0%–100%} range. The inputs of the FLS are generalized/abstracted: *Clue 1*, ..., *Clue n*, as the focus of the solution is again on the generalization, extensibility, and flexibility that a FL-based design allows for. These clues can then be particularized to the specific type of VN application. Some examples of inputs are provided together with their fuzzification for which trapezoidal membership functions have been chosen. However, as in the previously presented FL-based approaches, the focus is not on the internal design decisions of the FLS, but on its flexibility and generalization. Basically, in each vehicle, the parameters influencing the trust can be different, the clues being selected locally through a probabilistic scheme. Moreover, this generalized design of the security scheme can be employed in any type of data aggregation solution independent of the application type. In addition, the FL-based approach solves some of the aforementioned limitations of the other existing security mechanisms in VN data aggregation solutions. It removes the need for a tamper-proof service in each car, and it is dependent neither on node reputation as the scheme is data-centric, nor on any kind of structures, being also highly flexible.

5 FUZZY LOGIC-BASED CONTROL SOLUTIONS IN VEHICULAR NETWORKS

5.1 Fuzzy Logic Control Solutions in VN MAC Protocols

5.1.1 Why Using Fuzzy Logic? MAC protocols are considered to be a key issue in VN design [10] and they are identified among the main technical challenges imposed by VNs [5, 6]. In a VN context, efficient MAC protocols need to be designed in order to cope with the highly dynamic environment. In addition, they need to be able to provide quality of experience (QoE) for non-safety applications and reliability for safety applications.

In Reference [10], a recent survey of MAC protocols in VNs is presented with the focus on TDMA-based MAC protocols that classifies VN MAC protocols in three broad classes: contention-based, contention-free, and hybrid. The first class has the advantage of not being influenced by the ever-changing topology of VNs, but their main problem relates to the random delay introduced in order to regulate the access to the medium so that the chances of collisions are reduced. This delay is bounded by an interval, called the *backoff interval,* that is statically increased/decreased, an approach that is not the best for a dynamic environment such as that of VNs. Thus, the delay is controlled through the increase/decrease of this backoff interval regulated by so-called backoff schemes. An inappropriate control of this delay can cause serious issues especially in the case of safety applications. On the other hand, the second class of MAC protocols does not have the delay issue, but it is influenced by the topology change—slot relocation may often occur due to the rapidly changing topology of VNs. The third class of MAC protocols combines the two previous classes in a single architecture in order to minimize their disadvantages. So far, FL has been employed in VNs MAC protocols pertaining to the first class for an appropriate control of the delay that takes into consideration the complexity of VN environment and the imprecision that characterizes the network conditions in this environment.

5.1.2 State-of-the-Art. Abdelkader et al. [77] proposed a feedback FLS that controls the changes in the backoff interval (i.e., the level of increase/decrease). They started from the premise that each node in the network should monitor network conditions and, based on this control, their backoff interval increase/decrease. However, there is no direct mathematical mapping between network conditions and backoff interval computation; therefore, an exact model cannot be built. This is the reason why FL, as being suitable for dealing with imprecise information, was selected for modeling the relationship between the network conditions and backoff interval computation. The network conditions monitored are successful transmission ratio (*S*) and the last backoff interval value (*Blast*) as a measure of the network load. These are used as inputs for the FLS designed for controlling the increase/decrease in the backoff interval. The output is the normalized amount of decrease/increase of the interval, *dB*. Note that this is an adaptive FLS as the output processed in *Blast* is fed back into the FLS as input. The rule base is designed based on *expert knowledge*—authors' knowledge regarding the influence of the network conditions taken into consideration and backoff interval—and manual tuning—"trial experiments". The other FLS design decisions are mentioned in Table 3.

Abdelkader et al. [78] proposed four FL-based backoff control schemes: three are built upon SISO FLSs that control and the other one is built upon a MISO FLS. All these FLSs are designed based on the previously proposed FLS [77]. The MISO FLS has the same inputs and output as in Reference [77]. The changes are in the membership function of the output, that now has 5 fuzzy

Table 3. Summary of FL-Based Control Solutions in VNs

Subclass	Why using FL?	Objective	FLS design decisions: μ functions & Fuzzy sets definition	FLS design decisions: Rule base building method	FLS design decisions: Inference Type	FLS design decisions: Defuzz	FLS design decisions: FLS Arch.	Ref.	Testing
FL-based Control Solutions in VNs MAC Protocols	Ability to qualitatively capture the attributes of a control system based on observable phenomena; dealing with the imprecision	Backoff interval control	Triangular functions/n.a.	Expert knowledge	Mamdani	COA	Simple real-time adaptive FLS	[77]	Simulations
		Backoff interval control	Triangular functions/ n.a.	Expert knowledge	Mamdani	COA	Simple real-time adaptive FLS	[78]	Simulations
		Contention window control	Triangular functions/Expert knowledge	Expert knowledge	-	-	Simple real-time adaptive FLS	[79]	Simulations
		Contention window control	Triangular functions/Expert knowledge	Expert knowledge	-	-	Simple real-time adaptive FLS	[80]	Simulations
		Contention window control	Triangular functions/Expert knowledge	Expert knowledge	-	-	Simple real-time adaptive FLS	[81]	Simulations
FL-based Control Solutions in VNs-enabled Driving Automation	Control processes difficult to model and linearize	Speed Control	Singleton, triangular and trapezoidal functions/n.a.	-	Mamdani	COA	Classic FLS ⇒ non-adaptive type	[86]	Simulations
	FL has powerful reasoning capabilities & adapts to real-time conditions through reinforcement learning	traffic control system for intersections	Triangular functions (shape can be changed in real-time)/neural networks based learning	Expert knowledge	modified fuzzy model	Mean of max	Reinforcement learning-based real-time adaptive FLS	[87]	Real-world experiments
	Dealing with uncertainty and complexity; "imitating a human driver while steering and managing speed"	Automatic control of the steering and speed	Trapezoidal, triangular & singleton membership functions/ Manual tuning	Expert knowledge	Mamdani	COA	Simple real-time adaptive FLSs	[88]	Simulations
	FL is suitable for real-time systems and when there is no clear dependency between inputs and outputs	Speed control	Trapezoidal & singleton functions/automatic tuning	Expert knowledge	Sugeno	Sugeno specific	Classic FLS ⇒ non-adaptive type	[84]	Simulations and real-world experiments

(Continued)

Table 3. Continued

Subclass	Why using FL?	Objective	FLS design decisions					Ref.	Testing
			μ functions & Fuzzy sets definition	Rule base building method	Inference Type	Defuzz	FLS Arch.		
FL-based Control Solutions in Video Transmission over VNs	Dealing with multiple parameters	Redundancy amount control	Triangular functions/lessons learnt & *expert knowledge*	Lessons learnt & *expert knowledge* & analysis on results obtained through simulation	-	-	Classic FLS ⇒ non-adaptive type	[92]	Simulations
	Dealing with imprecision;design flexibility, easily tunable and open to changes	Redundancy amount control	Triangular and trapezoidal functions/ n.a.	-	Mamdani	COA	Classic FLS ⇒ non-adaptive type	[91]	Simulations
FL-based Beacon Control in VNs	Dealing with the imprecision	Beacons amount control	Triangular functions	Lessons learnt from the literature and *expert knowledge*	Mamdani	COA	Simple real-time adaptive FLS	[89]	Simulations

terms instead of 3 and consequently the rule base is extended with new rules, but it is built based on the same considerations—*expert knowledge*.

Regarding the three SISO FLSs proposed, they have only one input, S (successful transmission ratio), and same output as the MISO FLS. Each of these FLSs implements one of the following policies: selfish policy (i.e., the node objective is to access the network independent of its limitations), generous policy (i.e., the channel is given to other nodes if found busy), and cautious policy or a fair policy (i.e., each network node has the objective of a fair access to the medium). The rule base for each of these systems is designed based on *expert knowledge* in terms of authors' understanding on how the inputs influence the output. This knowledge can be summarized as follows: if the success ratio is very low this means that the channel is busy so the probability of collision is very high, therefore the recommendation is to increase the backoff interval. This recommendation is reflected differently by the three FLS in their rule bases depending on the policy adopted: selfish, generous, or fair. For instance, the selfish scheme tends to rather decrease the backoff interval than to increase it in most of the circumstances. All FLSs have triangular functions.

Chrysostomou et al. [79] proposed an FLS to control the wireless access in an adaptive QoS-aware MAC protocol. The proposed MAC protocol presents a different approach for controlling the backoff value. It keeps the IEEE 802.11p basic principle of updating the backoff interval based on the contention window (*CW*) value, but controls *CW* based on network conditions reflected in channel traffic occupancy (*CTO*). Keeping the IEEE 802.11p basic mechanism allows for differentiation and prioritization of different traffic types in the proposed solution. This combined with the FL-based control scheme of *CW* results in improved QoS level compared to the classic IEEE 802.11p. The FLS with control of *CW* has as inputs *CTO* values for consecutive sampling periods. The membership functions are triangular, design decision motivated by the authors by their computational simplicity. Their parameters were selected based on "qualitative understanding of the system". The rule base was built on the understanding of the system, *expert knowledge*, and manual tuning. The philosophy behind the knowledge base: on one hand aggressive response when the density of the channel is very high for two consecutive periods of time and on the other hand smooth response when the density is low.

In Reference [80], the same authors proposed a FL-based mechanism for controlling min and max *CW* values. According to the authors, this is the first scheme in the literature in which CW_{min} and CW_{max} values are adapted based on network conditions. The FLS has the same inputs as the FLS proposed in Reference [79] and as output the value controlling the increase/decrease of CW_{min} and CW_{max} values called factor. Similar design principles are adopted for designing this proposed FLS as those employed in Reference [79].

The solutions proposed in References [79] and [80] were designed for unicast communications. In Reference [81], these solutions were extended for broadcast communications. The modifications are done at the level of rule base in order to meet the requirements of broadcast communications.

5.2 Fuzzy Logic-Based Control Solutions in VN-Enabled Driving Automation

5.2.1 Why Using Fuzzy Logic? Automated driving is one of the key transformations that are taking place in the automotive industry [3]. There are several levels of driving automation that are defined starting from driver assistance and ending up to full automation, a scenario for the future. Both research and industry agrees that full automation is not possible on the basis of on-board sensors only, wireless communications, namely VNs, being mandatory [3]. Full automated driving is one of the main directions followed by the research in academia and industry in the automotive space in general and VNs in particular.

As mentioned above, driver assistance applications are the first steps towards autonomous driving. Driver assistance solutions provide driver with useful information in the driving process and if these solutions are more automated, they would intervene actively in the driving process. However, a driver assistance solution can be overridden by the driver. There is a plethora of VN-enabled driver assistance applications, including, for instance, applications that give driving advice based on certain criteria, such as, for instance, how to drive in certain conditions in order to reduce gas emissions or fuel or energy consumption in the case of electric cars. Very recently, VN-based assistance applications started to target not only drivers but also other types of users such as cyclists. Riding assistance applications were also proposed in order to give riding advice for energy consumption reduction in the case of electric bicycles [82–85].

FL has a considerable long history of success in developing self-control/automatic systems, such as, for instance, the speed control system deployed in Sendai subways or in the automotive industry intelligent cruise control solutions developed by Nissan or Peugeot. Consequently, FL is most suitable to be used in VN-enabled driving automation.

5.2.2 State-of-the-Art. Milanés et al. [86] proposed a FL-based crossroad-traversing system for autonomous cars that aims to improve traffic flow. The FL controller is used to control the speed of the car without right-of-way according to the speed of the car with right-of-way. The input information of the FLS is based on the information provided from the vehicular network, i.e., speed and positions of other vehicles. The FLS has a Mamdani inference type and is a MIMO type system, having three inputs and two outputs: throttle (T) and break (B). The membership functions of the inputs are trapezoidal and triangular, while the outputs have singleton membership functions. The decisions of the FLS design are not motivated. The defuzzification process is based on COA. Another FL-based and VN-enabled solution that considers autonomous vehicles is the one proposed in Reference [87]. A reinforcement learning-based real-time FLS was designed in order to control vehicular traffic at the intersections. The learning is based on neural networks and it allows for the real-time adaptation of the membership functions to the current traffic conditions: both membership parameters and the shape can be changed as a result of the learning process. Using this FLS, groups of vehicles are scheduled to cross the intersection in a real-time adaptive manner that is demonstrated to avoid delays and congestion in the intersections. The authors claim that the powerful reasoning capabilities of FL empowered by the learning ability of neural networks made their traffic control system more efficient and more adaptive to real-time traffic conditions. The approach makes the assumption that all the vehicles are autonomous.

The system proposed in Reference [88] employs FL in automatic steering and speed control in the context of a driver assistance system for safe overtaking maneuver. There are two FLSs designed, one for steering control and one for speed control. V2V communications are employed in collection of the information that is fed into the two FL controllers. FLSs have a simple real-time adaptive architecture. The membership functions chosen for the two FLSs are trapezoidal, triangular and singleton membership functions for their reduced computational complexity. For their parameter selection, manual tuning was performed in order to ensure satisfactory driving behavior. The rule bases are built based on *expert knowledge* that is defined by the authors as engineering judgment plus driver knowledge. Other design decisions for the two FLSs include the use of the Mamdani inference system and COA for defuzzification. The overall system was deployed and tested in a commercial vehicle.

FL was also employed in speed control in Reference [84] that proposes a VN-based speed advisory system for electric bicycles. An FLS is employed to control the increase/decrease in the speed of reference that is the maximum speed the cyclist is advised to ride to in order to conserve the energy in certain weather conditions, while not affecting considerable the time travel. V2I is used again for data collection that is fed into the FLS. The FLS follows Sugeno model and has trapezoidal and singleton membership functions chosen for their suitability for real-time systems and reduced computational complexity. Their parameters are determined at the system initialization as they are dependent on electric bicycle characteristics. The overall system is again tested using a vehicle—an electric bicycle, but also via extensive simulations.

5.3 Other Fuzzy Logic-Based Control Solutions in VNs

Another example of FL in control process is provided in Reference [24], where an FLS is used to control the beacon rate in the vehicular network, depending on the traffic conditions: in dense traffic conditions the beacon rate is required to be low, and in sparse traffic conditions the beacon rate is required to be high in order to increase the cooperative awareness. The inputs of the FLS designed to control the beacon rate are the percentage of the vehicles travelling in the same direction and the vehicle emergency status. The first parameter is chosen based on the traffic flow theory of Kerner [90] that states this parameter is an indicator of traffic density, while the latter is imposed by the fact that an emergency vehicle has to continue sending its status in the network. Thus, the beacon rate is dependent not only on traffic density, but also on the emergency status of the vehicle. This is one of the considerations that, together with the *expert knowledge*, are at the basis of the FLS rule base. The output of the FLS is represented by the beacon rate. Both, the inputs and the output have triangular membership functions. The inference type used is Mamdani, while the defuzzification method is COA.

Ghafoor et al. [91] proposed a FL redundancy controller for controlling the amount of redundant packets depending on the traffic density and SNR of the channel. The controller is designed in the context of a video streaming solution for VNs. The traffic density and SNR of the channel are the inputs of the FLS, while the output is called the coding density: the ratio between the encoded packets and the whole amount of packets received. The option for FL employment in controlling the amount of redundant packets in order to improve the network load is motivated by the authors based on the capacity of FL of dealing with the uncertainty and imprecision, characteristics of the ever-changing VNs environment. Moreover, the authors emphasize the advantages of the FLSs/FL controllers: their modifiability, as it is easy to tune rules, membership functions or even change the parameters of the system in order to enhance its performance. All the FLS design decisions are presented in Table 3.

A FL-based redundancy controller is also employed in Reference [92]. The difference is that the amount of redundancy is considered for each packet delivered over a QoE-driven video transmission mechanism over VNs. In addition, more parameters are considered for controlling the redundancy. FL is employed due to its capability of dealing with multiple parameters. The authors do not describe the FLS in detail, and only an example of membership function is given, which is triangular.

6 FUZZY LOGIC-BASED DETECTION AND PREDICTION SOLUTIONS IN VEHICULAR NETWORKS

6.1 Why Using Fuzzy Logic?

FL has been employed in detection and prediction systems in various areas, including technical areas such as networking (e.g., detect some faults in the network, detect/predict network congestion) and automotive (e.g., different fault detections, prediction of driver's maneuver [93]) or less technical areas such as banking, elections, or medicine [27, 36]. Imprecision characterizes detection and especially prediction solutions independent of the area they are employed in. Therefore, FL is a powerful and suitable tool to be employed in such solutions. The detection and prediction FL-based solutions in the context of VNs were used to predict acceleration and speed, detect

Table 4. Summary of FL-Based Detection and Prediction Solutions in VNs

Why using FL?	Objective	FLS design decisions: μ functions & Fuzzy sets definition	Rule base building method	Inference Type	Defuzz.	FLS Arch.	Ref.	Testing
Dealing with the volatile characteristics of VNs	Prediction of speed and detection of network congestion	Trapezoidal functions/ Automatic tuning – PSO techniques	-	Tsukamoto	Tsukamoto-specific	Reinforcement learning-based real-time adaptive FLS	[95]	Simulations
Dealing with the uncertainty; An FLS is adaptable to external changes when combined with learning techniques	Acceleration prediction	Triangular functions/ automatic tuning using a basic reinforcement learning algorithm	Expert knowledge	Mamdani	COA	Reinforcement learning-based real-time adaptive FLS	[94]	Simulations
Addressing complex non-deterministic problem: detecting traffic congestion	Road traffic congestion detection	Singleton, triangular and trapezoidal functions/ Lessons learnt from the literature	Lessons learnt from the literature	Sugeno	Sugeno-specific	Classic FLS ⇒ non-adaptive type	[29] [96]	Simulations
FL allows for human like reasoning and for adapting the system through tuning	Ticket rate prediction	Triangular and Gaussian functions/n.a.	-	-	-	Classic FLS ⇒ non-adaptive type	[97]	Simulations

network congestion, and predict ticket rate in a tolling system used on highways. All these solutions are described next.

6.2 State-of-the-art

Hafeez et al. [94] described their new cluster-based VN MAC protocol which makes use of an adaptive FLS for predicting the speed and location in order to adjust the protocol to driver's behavior on the road. The FLS is used in the cluster maintenance process and based on the predictions made, the structure of the cluster is updated or not. The inputs of the FLS are the speed and inter-vehicle distance and the output predicts if the driver is going to accelerate or decelerate. Based on this prediction, the vehicle's speed and position in the near future are further predicted. The membership functions of inputs and output are triangular. A basic reinforcement learning algorithm is implemented for automatic tuning of the parameters of speed membership function. Basically, this mechanism adapts the FLS to the driver's behavior. The rule base of the FLS is fully described and it is based on *expert knowledge*. Other FLS design decisions are illustrated in Table 4.

A FL predictive system is employed in a routing protocol in order to design a mechanism for proactive recovery from failure [95]. This mechanism, deployed in each vehicle, involves two components: an alternative link construction component and a prediction component. The prediction component detects/predicts the congestion or the link failure and either if congestion is detected or link failure is predicted it activates the alternate link construction component. The prediction component has two modules: Fuzzy speed prediction module and Fuzzy congestion detection module. Basically, these modules are represented by two FLSs. The FLS predicting the speed incorporates some knowledge about driver's age as there is a connection between driver's age and driver's behavior. Thus, the inputs of this FLS are driver's age, distance between the vehicle and the front vehicle, and current speed. The output is the predicted velocity. The FLS for congestion detection has as inputs: queue length, hop count that the packets travel through in terms of number of vehicles and expected number of the vehicles within radio range during next time period, and as output the congestion indicator. Both FLSs have similar design: trapezoidal membership functions chosen to reduce computational complexity, and Tsukamoto inference type and defuzzification method. The design of the FLS follows the real-time adaptive design that allows for automatic tuning of the parameters of the membership functions based on Particle Swarm Optimization techniques.

A FL-based detection solution is described in References [29] and [96], where an FL-based system for road traffic congestion detection was designed starting from the premise that FL is a powerful tool to address complex nondeterministic problems as it is the case of traffic congestion detection. The FLS developed to determine the level of congestion is designed following some rules of congestion developed by Skycomp [38]. These rules express in a linguistic manner the level of congestion based on density of the vehicles and their speed. Thus, FL appears as the natural tool in solving this problem. An FLS is designed to be deployed on each vehicle for detecting the level of congestion around. The inputs of the FLS are the speed of the vehicle and the density that is determined based on the number of neighboring vehicles detected through V2V communications. The membership functions of the inputs are triangular and trapezoidal. The output of the FLS is the level of congestion and its membership function is a singleton. Although not explicitly specified, from the description and the results detailed, it is clear that the FLS has a Sugeno inference type.

In Reference [97], a V2V and FL-based prediction solution is proposed in order to predict the ticket rate depending on the vehicular fluctuation in the context of ticket generation system for highways that aims to address traffic congestion. The performance evaluation of the solution shows that the system reduces congestion on the highway and decreases the travel time. There are not too many details provided regarding the FLS design: the inputs are the congestion level and number of vehicles queuing and have triangular and Gaussian membership functions, respectively. The authors chose FL due to the fact that it allows for human-like thinking which made it more natural and easier to describe the ticket generation rate, depending on the traffic congestion level and number of vehicles queuing, as there is no exact dependency between inputs and the output. Another aspect highlighted by the authors is the fact that FLSs in general are open to change having a flexible design: the rules and membership functions defined in the context of an FLS can be easily modified.

7 CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

VNs are acknowledged for their great potential of supporting an extremely varied range of applications. The large positive impact of these applications is measured in crashes avoided, lives saved, traffic congestion reduced, improvements on environment, and increased satisfaction of the drivers in traffic. VNs have many challenges that arise mainly due to their dynamic and complex environment. FL known for its ability to deal with complexity and imprecision and model nondeterministic problems was considered for application in the dynamic and complex VNs context. In this article, we analyzed these FL-based solutions and, on the basis of this analysis, we present next some lessons learnt related to FL implementation in VN space and future research directions.

7.1 Lessons Learnt

Some of the lessons that can be learnt from the survey performed on FL solutions in VNs are summarized below:

Fig. 7. Why using FL—Keywords cloud.

— FL seems to be a powerful mathematical tool for dealing with imprecision and uncertainty of VN dynamic environment. FL seems also to be able to deal with multiple parameters that are necessary in order to describe the complexity of this environment.
— The previous considerations plus the fact that FL is a powerful decisional tool resulted in the employment of FL in decision making in the context of a large variety of VN-based solutions (see Figure 7). The results of these solutions recommend FL as being suitable to be employed in making complex decisions in the context of VNs.
— FL seems to be a very powerful tool that can be used for control processes that are difficult to model and this is exactly the case in VN environment.
— An FLS has a design that allows for flexibility and generality at conceptual, structural, and architectural levels.
— FLS are considered to be suitable to be applied in VNs as they have predefined automatic tuning techniques for adjusting membership functions and rules to accommodate the dynamic network environment. In VN context, real-time tuning (i.e., reinforcement learning-based tuning) appears to be more suitable than the off-line tuning (supervised learning-based tuning); except for the cases where the solution is designed for certain network architecture. Otherwise, in order to adapt to the diversity of conditions imposed by VNs, it is most appropriate that FLS is real-time adaptive. One issue that might arise is that the learning time at the run time might affect the performance of the FLS in such a dynamic environment. It is imperative for the researchers to consider this aspect and try to choose faster converging learning techniques. What can be fast for some domains, may not be fast enough for VNs, a highly volatile environment. Regarding this aspect, our survey reveals that $H\infty$ filtering and Q learning techniques appears to be the most suitable techniques applied so far in VNs context. The solutions that employed Q learning did not report any performance issue that might be caused by the learning time. Some other techniques might introduce a longer learning time which affects the performance. For instance, in Reference [40] the authors experimented with three techniques: $H\infty$ filtering, neural networks (NN)-based tuning and genetic algorithms (GA)-based tuning. The performance tests have shown that while the accuracy is comparable, the learning time is much longer for NN and GA-based tuning as compared to $H\infty$ filtering. In this context, a better option is to choose a simple real-time adaptive FLS architecture as this eliminates the complexity of learning algorithms and reduces the time needed for learning, but might still provide good adaptation.
— In the previous paragraph some performance tests in relation to different learning techniques were discussed. It is to be highlighted that more such tests are needed. More performance tests are required before FL can be considered as a feasible technology for VN dynamic environment with the emphasis on FL imposed complexity. This aspect is discussed in more details in Section 7.2.
— Regarding the other design decisions related to an FLS, Mamdani is the most popular inference type, together with its specific COA method for defuzzification. The complexity of

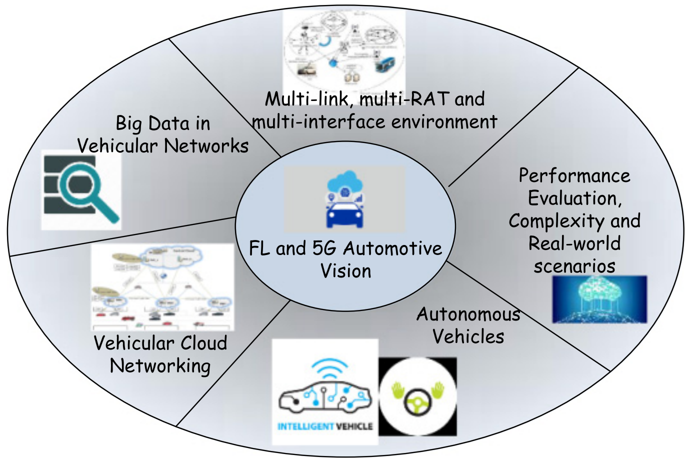

Fig. 8. Future research directions.

this is compensated by opting for singleton, triangular or trapezoidal membership functions that prove to provide good performance in this context and due to their reduced computational complexity and efficiency they are perfect candidates when choosing to design any FLS used in VN solutions.

7.2 Future Research Directions: The 5G Automotive Vision and Fuzzy Logic

Currently, steps are being made towards 5G, the next generation of mobile communication technology. VNs and their applications in automotive industry occupy a distinct place in the 5G network design [3]. The 5G networks will include novel technologies for V2X communications that will co-exist with existing VN-enabling solutions that were briefly introduced at the beginning of this survey. Consequently, 5G is foreseen to integrate a mix of Radio Access Technologies (RAT) and enable their cooperation and combination. This will lead to the creation of a multi-link multi-RAT multi-interface environment which will be associated with many other challenges. Some of these challenges are discussed in Section 7.2.1. A more detailed and comprehensive list of the issues that encourage new research directions in the 5G-enabled VNs and their applications is presented in Reference [3]. Some of the key avenues identified by both industry and academia in the 5G automotive vision are as follows: autonomous or self-driving vehicles, vehicular cloud and big data. As this paper focuses on FL-related aspects, the following subsections include discussions about the possibility of employing FL in these research directions. Additionally, another future research direction that can be identified in relation to FL, VN and VN-enabled applications is performance evaluation and complexity analysis of FL-based solutions. A summary of the identified future works is displayed in Figure 8.

7.2.1 Multi-Link Multi-RAT and Multi-Interface Environment. Supporting such an environment is a challenge in 5G-enabled VNs. In this "multi-multi-multi" environment, decision making has an important role and we have seen in this review that FL is a powerful decision-making framework.

In this context, the need for proposing new routing algorithms to encompass this "multi-multi-multi" aspect is highlighted. Path selection in a multi-link environment will be a main challenge to address; deciding dynamically which packets should be assigned to which link, when the links have various characteristics (e.g., different performances). As noted in this review, FL is highly successful in dealing with path selection in the context of different VN-proposed algorithms. The main reasons for employing FL relate to dealing with decision making in real-time based on multiple parameters that describe highly variable conditions. This is exactly the issue (only more augmented) that needs to be addressed in the context of 5G-enabled VNs, and therefore we believe to be worthwhile investigating the possibility of employing FL in the newly developed routing algorithms for 5G-enabled VNs.

Another use case for decision making in this "multi-multi-multi" environment is network selection. RAT selection or – as a new aspect – selection of multiple RATs, based on a variety of parameters in order to ensure QoS requirements for content delivery, is possible in 5G. Again, based on existing success stories that were also described in this review, investigation of the possibility of employing FL in this context is also interesting. Moreover, another argument in favor of FL is the fact that FL decisional systems are highly adaptable: they are easier to adapt in order to consider multiple outputs as it is the case for multiple selection of networks.

7.2.2 Autonomous Vehicles and Fuzzy Logic. As already mentioned, a major and highly promising fields in the automotive research includes self-driving or autonomous vehicles. The overall goal of the effort put in this space is to have the first such cars on the market by 2020. In this survey we discussed some VN FL-based control solutions that represent important steps made towards autonomous vehicles. FL can play a prominent role in this research direction as it is able to provide a human-like reasoning, has a transparent and flexible design, and is able to adapt to the dynamic conditions that characterize the VN environment. Moreover, FL has a long history of success in self-control systems that are already deployed in automotive and electronics industries.

7.2.3 Vehicular Cloud Networking and Fuzzy Logic. The support of VNs and the recent advancements in computing and storage technologies developed for vehicles have enabled the definition of vehicular cloud computing. In the vehicular cloud, any vehicle with computing and storage capabilities can be not only service consumer, but also a service provider. By being part of a vehicle cloud, any vehicle could share its computing, storage, and sensing resources to support advanced services.

The research in this area is in early stages [14], but as stated before, it is considered a very promising research direction and a main component of the 5G-enabled automotive vision [3]. In comparison to the traditional cloud paradigm, vehicular cloud has its specific characteristics and challenges. One of its main challenges is the variability of the available resources [14]. This is due to the dynamic vehicle behavior as they can join and leave the cloud at any given time. To address this issue, FL could be used for instance in predicting if a vehicle will leave the cloud in order to avoid relying on its resources/services. Moreover, as we have seen in this survey, clustering can address the issue of resource management in VNs in general and in vehicular cloud in particular and FL was successfully employed with cluster-based architectures. We argue that this could be another FL-based viable solution to the resource variability problem in vehicular cloud and a very promising research avenue.

7.2.4 Big Data in Vehicular Networks and Fuzzy Logic. Tremendous amount of data is expected to be generated, disseminated, processed, and stored in this vehicular landscape: a network with almost no limit in terms of the number of nodes and with a plethora of applications. However, existing methods, models, and algorithms for general big-data processing and analysis may not be suitable for VNs. One of the most important aspects in VN context is real-time or near-real-time big-data analysis requirement, which is not present in the big-data paradigm. Some steps were made in this direction, but more work needs to be performed [69, 98]. One of the research avenues relates to data aggregation techniques: there is a need for new, efficient, and secure data aggregation schemes in VN context. As presented in this work, FL was successfully applied so far in VN data aggregation schemes. A FL-based solution in data aggregation provides flexibility in the criteria used for deciding upon the data similarity and deciding upon the trustfulness of data, too. The design of such solution is flexible, open to changes as imposed by the VN requirements.

Along with data aggregation schemes, other data manipulation techniques are required for big data in VN context. Decision making occupies a central role in big-data paradigm in general (e.g., decision in classifying data, decision if the data is trustful or not, etc.) [99] and as we have seen in the survey, FL is a powerful tool in decision-making processes applied to VNs, and also in other domains. Moreover, FL was successfully employed in big data analysis, for instance in the analysis of data in social networking [100, 101].

Based on all these considerations, we conclude that the applicability of FL in the context of big data in VNs is a promising research direction.

7.2.5 Performance Evaluation, Complexity and Real-World Scenarios. Most of the proposed solutions were tested through simulations. Although most of these were large-scale simulations, the simulation models tend to be simplified in comparison to the real-world scenarios. Moreover, in the performance evaluation of the solutions discussed, there is little analysis of the complexity introduced by FL reasoning in each solution. As mentioned in the Lessons Learnt section, there were some performance measurements regarding the learning time with various techniques, but no real complexity analysis related to the impact of applying FL in any of the solutions. This aspect should be investigated in further works. Other aspects of high interest to be considered in future works include comparison-based assessments between different FL approaches, and, knowing that the complexity of FLSs increases with the number of inputs, investigation in terms of the number of inputs/parameters which could be successfully handled in different VN scenarios (e.g., sparse, dense, high speeds, low speeds, etc.).

Although the majority of the analyzed FL-based solutions were tested through simulations only, some of these were also tested through real-world experiments as it can be seen in the tables summarizing these solutions (i.e., Tables 1, 2, 3, 4). It is noteworthy that the results show there are no performance drawbacks imposed by FL reasoning; on the contrary, the FL-based solutions outperform other solutions. However, the real-world experiments are not based on a large-scale deployment of the solutions. For instance, the testing of the routing protocols proposed in References [42] and [45] consider a VN based on 10 vehicle-nodes in three different scenarios (imposed by three different road types). The ultimate confirmation of the feasibility of FL reasoning in VN context would be obtained through large-scale deployment of such solutions in real-life.

REFERENCES

[1] Georgios Karagiannis, Onur Altintas, Eylem Ekici, Geert Heijenk, Boangoat Jarupan, Kenneth Lin, and Timothy Weil. 2011. Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions. *IEEE Communications Surveys & Tutorials* 13, 4 (2011), 584–616.

[2] Riccardo Coppola and Maurizio Morisio. 2016. Connected car: Technologies, issues, future trends. *ACM Computing Surveys (CSUR)* 49, 3 (2016), 46.

[3] "5G Automotive Vision". 2015. White paper. Retrieved from https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-White-Paper-on-Automotive-Vertical-Sectors.pdf.

[4] Soufiene Djahel, Ronan Doolan, Gabriel-Miro Muntean, and John Murphy. 2015. A communications-oriented perspective on traffic management systems for smart cities: Challenges and innovative approaches. *IEEE Communications Surveys & Tutorials* 17, 1 (2015), 125–151.

[5] Hassnaa Moustafa and Yan Zhang. 2009. *Vehicular Networks: Techniques, Standards, and Applications.* CRC Press, 2009.

[6] Mukesh Saini, Abdulhameed Alelaiwi, and Abdulmotaleb El Saddik. 2015. How close are we to realizing a pragmatic VANET solution? A meta-survey. *ACM Computing Surveys (CSUR)* 48, 2 (2015), 29.

[7] Yasser L. Morgan. 2010. Notes on DSRC & WAVE standards suite: Its architecture, design, and characteristics. *IEEE Communications Surveys & Tutorials* 12, 4 (2010), 504–518.

[8] Kevin C. Lee, Uichin Lee, and Mario Gerla. 2010. Survey of routing protocols in vehicular ad hoc networks. *Advances in Vehicular ad-Hoc Networks: Developments and Challenges* (2010), 149–170.
[9] Salim Bitam, Abdelhamid Mellouk, and Sherali Zeadally. 2015. Bio-inspired routing algorithms survey for vehicular ad hoc networks. *IEEE Communications Surveys & Tutorials* 17, 2 (2015), 843–867.
[10] Mohamed Hadded, Paul Muhlethaler, Anis Laouiti, Rachid Zagrouba, and Leila Azouz Saidane. 2015. TDMA-Based MAC protocols for vehicular ad hoc networks: A survey, qualitative analysis, and open research issues. *IEEE Communications Surveys & Tutorials* 17, 4 (2015), 2461–2492.
[11] Irina Tal and Gabriel-Miro Muntean. 2014. Towards smarter cities and roads: A survey of clustering. *Convergence of Broadband, Broadcast, and Cellular Network Technologies* (2014), 16.
[12] Rasmeet S. Bali, Neeraj Kumar, and Joel JPC Rodrigues. 2014. Clustering in vehicular ad hoc networks: Taxonomy, challenges and solutions. *Vehicular communications* 1, 3 (2014), 134–152.
[13] Anna Maria Vegni and Valeria Loscri. 2015. A survey on vehicular social networks. *IEEE Communications Surveys & Tutorials* 17, 4 (2015), 2397–2419.
[14] Euisin Lee, Eun-Kyu Lee, Mario Gerla, and Soon Y. Oh. 2014. Vehicular cloud networking: Architecture and design principles. *IEEE Communications Magazine* 52, 2 (2014), 148–155.
[15] Fabrício A. Silva, Azzedine Boukerche, Thais R. M. Silva, Linnyer B. Ruiz, Eduardo Cerqueira, and Antonio A. F. Loureiro. 2016. Vehicular networks: A new challenge for content-delivery-based applications. *ACM Computing Surveys (CSUR)* 49, 1 (2016), 11.
[16] Sumit Ghosh, Qutaiba Razouqi, H. Jerry Schumacher, and Aivars Celmins. 1998. A survey of recent advances in fuzzy logic in telecommunications networks and new challenges. *IEEE Transactions on Fuzzy Systems* 6, 3 (1998), 443–447.
[17] Ramona Trestian, Olga Ormond, and Gabriel-Miro Muntean. Game theory-based network selection: Solutions and challenges. *IEEE Communications Surveys & Tutorials* 14, 4 (2012), 1212–1231.
[18] Lusheng Wang and Geng-Sheng GS Kuo. 2013. Mathematical modeling for network selection in heterogeneous wireless networks—A tutorial. *IEEE Communications Surveys & Tutorials* 15, 1 (2013), 271–292.
[19] Raghavendra V. Kulkarni, Anna Forster, and Ganesh Kumar Venayagamoorthy. 2011. Computational intelligence in wireless sensor networks: A survey. *IEEE Communications Surveys & Tutorials* 13, 1 (2011), 68–96.
[20] Liljana Gavrilovska, Vladimir Atanasovski, Irene Macaluso, and Luiz A. DaSilva. 2013. Learning and reasoning in cognitive radio networks. *IEEE Communications Surveys & Tutorials* 15, 4 (2013), 1761–1777.
[21] Mario Bkassiny, Yang Li, and Sudharman K. Jayaweera. 2013. A survey on machine-learning techniques in cognitive radios. *IEEE Communications Surveys & Tutorials* 15, 3 (2013), 1136–1159.
[22] Mohammad Hossein Manshaei, Quanyan Zhu, Tansu Alpcan, Tamer Bacşar, and Jean-Pierre Hubaux. 2013. Game theory meets network security and privacy. *ACM Computing Surveys (CSUR)* 45, 3 (2013), 25.
[23] Zhongshan Zhang, Keping Long, Jianping Wang, and Falko Dressler. 2014. On swarm intelligence inspired self-organized networking: Its bionic mechanisms, designing principles and optimization approaches. *IEEE Communications Surveys & Tutorials* 16, 1 (2014), 513–537.
[24] Giuseppe Araniti, Claudia Campolo, Massimo Condoluci, Antonio Iera, and Antonella Molinaro. 2013. LTE for vehicular networking: A survey. *IEEE Communications Magazine* 51, 5 (2013), 148–157.
[25] Xingqin Lin, Jeffrey G. Andrews, Amitabha Ghosh, and Rapeepat Ratasuk. 2014. An overview of 3GPP device-to-device proximity services. *IEEE Communications Magazine* 52, 4 (2014), 40–48.
[26] S. Lancaster, S. 2008. *Fuzzy Logic Controllers*, PSU: Maseeh College of Engineering and Computer Science.
[27] Jerry M. Mendel. 1995. Fuzzy logic systems for engineering: A tutorial. *Proceedings of the IEEE* 83, 3 (1995), 345–377.
[28] Lotfi A. Zadeh. 1974. The concept of a linguistic variable and its application to approximate reasoning. In *Learning Systems and Intelligent Robots*. Springer, 1–10.
[29] Ramon Bauza and Javier Gozálvez. 2013. Traffic congestion detection in large-scale scenarios using vehicle-to-vehicle communications. *Journal of Network and Computer Applications* 36, 5 (2013), 1295–1307.
[30] Stefan Dietzel, Boto Bako, Elmar Schoch, and Frank Kargl. 2009a. A fuzzy logic based approach for structure-free aggregation in vehicular ad-hoc networks. In *Proceedings of the 6th ACM International workshop on VehiculAr Inter-NETworking, ACM*, 79–88. 2009.
[31] Irina Tal and Gabriel-Miro Muntean. 2013. User-oriented fuzzy logic-based clustering scheme for vehicular ad-hoc networks. In *Proceedings of the 77th IEEE Vehicular Technology Conference (VTC Spring)*. IEEE, 1–5.
[32] Ebrahim H. Mamdani and Sedrak Assilian. 1975. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies* 7, 1 (1975), 1–13.
[33] Tomohiro Takagi and Michio Sugeno. 1985. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics* 1 (1985), 116–132.
[34] Yahachiro Tsukamoto. 1979. An approach to fuzzy reasoning method. *Advances in Fuzzy Set Theory and Applications* 137 (1979), 149.

[35] P. Martin Larsen. 1980. Industrial applications of fuzzy logic control. *International Journal of Man-Machine Studies* 12, 1 (1980), 3–10.

[36] Timothy J. Ross. 2010. *Fuzzy Logic with Engineering Applications*. John Wiley & Sons.

[37] Nazmul Siddique and Hojjat Adeli. 2013. *Computational Intelligence: Synergies of Fuzzy Logic, Neural Networks and Evolutionary Computing*. John Wiley & Sons.

[38] Skycomp. 2009. Major highway performance ratings and bottleneck inventory - State of Maryland – Spring 2008. *Inc. In association with Whitney, Bailey, Cox and Magnani*. Retrieved from http://www.skycomp.com/MDSHA/resources/Spring_2008.pdf.

[39] Xiao-bo Wang, Yu-liang Yang, and Jian-wei An. 2009. Multi-metric routing decisions in vanet. In *Proceedings of the 8th IEEE International Conference on Dependable, Autonomic and Secure Computing (DASC'09)*. IEEE, 551–556.

[40] Chenn-Jung Huang, I-Fan Chen, Kai-Wen Hu, Hung-Yen Shen, You-Jia Chen, and Dian-Xiu Yang. 2009a. A load balancing and congestion-avoidance routing mechanism for teal-time traffic over vehicular networks. *J. UCS* 15, 13 (2009). 2506–2527.

[41] Celimuge Wu, Satoshi Ohzahata, and Toshihiko Kato. 2012. Routing in VANETS: A fuzzy constraint q-learning approach. In *Global Communications Conference (GLOBECOM)*. IEEE, 195–200.

[42] Celimuge Wu, Satoshi Ohzahata, and Toshihiko Kato. 2013a. Flexible, portable, and practicable solution for routing in VANETs: A fuzzy constraint q-learning approach. *IEEE Transactions on Vehicular Technology* 62, 9 (2013), 4251–4263.

[43] Christopher John Cornish Hellaby Watkins. 1989. *Learning from Delayed Rewards*. Ph.D diss., University of Cambridge.

[44] Celimuge Wu and Kazuya Kumekawa. 2010. Distributed reinforcement learning approach for vehicular ad hoc networks. *IEICE Transactions on Communications* 93, 6 (2010), 1431–1442.

[45] Celimuge Wu, Yusheng Ji, Fuqiang Liu, Satoshi Ohzahata, and Toshihiko Kato. 2015a. Toward practical and intelligent routing in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology* 64, 12 (2015), 5503–5519.

[46] Douglas S. J. De Couto, Daniel Aguayo, John Bicket, and Robert Morris. 2005. A high-throughput path metric for multi-hop wireless routing. *Wireless Networks* 11, 4 (2005), 419–434.

[47] Mohammad Al-Rabayahand Robert Malaney. 2012. A new scalable hybrid routing protocol for VANETs. *IEEE Transactions on Vehicular Technology* 61, 6 (2012), 2625–2635.

[48] Celimuge Wu, Satoshi Ohzahata, Yusheng Ji, and Toshihiko Kato. 2016. How to utilize interflow network coding in VANETs: A backbone-based approach. *IEEE Transactions on Intelligent Transportation Systems* 17, 8, 2223–2237.

[49] Celimuge Wu, Satoshi Ohzahata, Yusheng Ji, and Toshihiko Kato. 2014. Making inter-flow network coding possible for unicast routing in VANETs. In *Proceedings of the IEEE 2014 International Conference on Connected Vehicles and Expo (ICCVE)*. 829–835.

[50] Celimuge Wu, Satoshi Ohzahata, and Toshihiko Kato. 2013b. Can we generate efficient routes by using only beacons? Backbone routing in VANETs. In *Proceedings of the 2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, 2929–2934.

[51] Celimuge Wu, Xianfu Chen, Yusheng Ji, Satoshi Ohzahata, and Toshihiko Kato. 2015b. Efficient broadcasting in VANETs using dynamic backbone and network coding. *IEEE Transactions on Wireless Communications* 14, 11 (2015), 6057–6071.

[52] Rashid Hafeez Khokhar, Rafidah Md Noor, Kayhan Zrar Ghafoor, Chih-Heng Ke, and Md Asri Ngadi. 2011. Fuzzy-assisted social-based routing for urban vehicular environments. *EURASIP Journal on Wireless Communications and Networking*, 1 (2011), 1.

[53] Celimuge Wu, Satoshi Ohzahata, and Toshihiko Kato. 2010. Fuzzy logic based multi-hop broadcast for high-density vehicular ad hoc networks. In *Proceedings of the IEEE Vehicular Networking Conference (VNC)*. IEEE, 17–24.

[54] Celimuge Wu, Satoshi Ohzahata, Yusheng Ji, and Toshihiko Kato. 2015c. Joint fuzzy relays and network-coding-based forwarding for multihop broadcasting in VANETs. *IEEE Transactions on Intelligent Transportation Systems* 16, 3 (2015), 1415–1427.

[55] Celimuge Wu, Yusheng Ji, Xianfu Chen, Satoshi Ohzahata, and Toshihiko Kato. 2015e. An intelligent broadcast protocol for VANETs based on transfer learning. In *Proceedings of the 2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*. IEEE, 1–6.

[56] Celimuge Wu, Xianfu Chen, Yusheng Ji, Fuqiang Liu, Satoshi Ohzahata, Tsutomu Yoshinaga, and Toshihiko Kato. 2015d. Packet size-aware broadcasting in VANETs with fuzzy logic and rl-based parameter adaptation. *IEEE Access* 3 (2015), 2481–2491.

[57] Chenn-Jung Huang, You-Jia Chen, I-Fan Chen, and Tsung-Hsien Wu. 2009b. An intelligent infotainment dissemination scheme for heterogeneous vehicular networks. *Expert Systems with Applications* 36, 10 (2009), 12472–12479.

[58] Elnaz Limouchi, Imad Mahgoub, and Ahmad Alwakeel. 2016. Fuzzy logic-based broadcast in vehicular ad hoc networks. In *Proceedings of the IEEE Vehicular Technology Conference (VTC-Fall)*. 1–5.

[59] Elnaz Limouchi and Imad Mahgoub. 2016. BEFLAB: Bandwidth efficient fuzzy logic-assisted broadcast for VANET. In *Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI)* 1–8.
[60] Jane Yang Yu and Peter Han Joo Chong. 2005. A survey of clustering schemes for mobile ad hoc networks. *IEEE Communications Surveys & Tutorials* 7, 1 (2005), 32–48.
[61] Hamid Reza Arkian, Reza Ebrahimi Atani, Abolfazl Diyanat, and Atefe Pourkhalili. 2015. A cluster-based vehicular cloud architecture with learning-based resource management. *The Journal of Supercomputing* 71, 4 (2015), 1401–1426.
[62] Zhioua El-Mouna, Ghayet, Nabil Tabbane, Houda Labiod, and Sami Tabbane. 2015. A fuzzy multi-metric QoS-balancing gateway selection algorithm in a clustered VANET to LTE advanced hybrid cellular network. *IEEE Transactions on Vehicular Technology* 64, 2 (2015), 804–817.
[63] Kan Zheng, Qiang Zheng, Periklis Chatzimisios, Wei Xiang, and Yiqing Zhou. 2015. Heterogeneous vehicular networking: A survey on architecture, challenges, and solutions. *IEEE Communications Surveys & Tutorials* 17, 4 (2015), 2377–2396.
[64] Jindong Hou and Dominic C. O'Brien. 2006. Vertical handover-decision-making algorithm using fuzzy logic for the integrated Radio-and-OW system. *IEEE Transactions on Wireless Communications* 5, 1 (2006), 176–185.
[65] Shubha Kher, Arun K. Somani, and Rohit Gupta. 2005. "Network selection using fuzzy logic. In *Proceedings of the 2nd International Conference on Broadband Networks*. IEEE, 876–885.
[66] Inthawadee Chantaksinopas, Phoemphun Oothongsap, and Akara Prayote. 2010. Framework for network selection transparency on vehicular networks. In *Proceedings of the 2010 International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI-CON)*. IEEE, 593–597.
[67] Bin Ma and Xiaofeng Liao. 2012. Speed-adaptive vertical handoff algorithm based on fuzzy logic in vehicular heterogeneous networks. In *Proceedings of the 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, IEEE, 371–375.
[68] Tarek Bouali and Sidi-Mohammed Senouci. 2015. A fuzzy logic-based communication medium selection for QoS preservation in vehicular networks. In *Proceedings of the 5th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications*. ACM, 101–108.
[69] Irina Tal and Gabriel-Miro Muntean. 2012. Using fuzzy logic for data aggregation in vehicular networks. In *Proceedings of the 2012 IEEE/ACM 16th International Symposium on Distributed Simulation and Real Time Applications*. IEEE Computer Society, 151–154.
[70] Stefan Dietzel, Jonathan Petit, Frank Kargl, and Björn Scheuermann. 2014. In-network aggregation for vehicular ad hoc networks. *IEEE Communications Surveys & Tutorials* 16, 4 (2014), 1909–1932.
[71] Sergio Ilarri, Thierry Delot, and Raquel Trillo-Lado. 2015. A data management perspective on vehicular networks. *IEEE Communications Surveys & Tutorials* 17, 4 (2015), 2420–2460.
[72] Sonja Buchegger, Jochen Mundinger, and Jean-Yves Le Boudec. 2008. Reputation systems for self-organized networks. *IEEE Technology and Society Magazine* 27, 1 (2008), 41–47.
[73] Björn Scheuermann, Christian Lochert, Jedrzej Rybicki, and Martin Mauve. 2009. A fundamental scalability criterion for data aggregation in VANETs. In *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*. ACM, 285–296.
[74] Pino Caballero-Gil, Jezabel Molina-Gil, and Cándido Caballero-Gil. 2011. Data aggregation based on fuzzy logic for VANETs. In *Computational Intelligence in Security for Information Systems*. Springer, Berlin, 33–40.
[75] Stefan Dietzel, Elmar Schoch, Boto Bako, and Frank Kargl. 2009b. A structure-free aggregation framework for vehicular ad hoc networks. In *Proceedings of the 6th International Workshop on Intelligent Transportation*, Hamburg, Germany, 61–66.
[76] Stefan Dietzel, Elmar Schoch, Bastian Konings, Michael Weber, and Frank Kargl. 2010. Resilient secure aggregation for vehicular networks. *IEEE Network* 24, 1 (2010), 26–31.
[77] Tamer Abdelkader, Kshirasagar Naik, Amiya Nayak, and Fakhry Karray. 2009. Adaptive backoff scheme for contention-based vehicular networks using fuzzy logic. In *Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2009)*. IEEE, 1621–1626.
[78] Tamer Abdelkader, Kshirasagar Naik, and Amiya Nayak. 2011. Using fuzzy logic to calculate the backoff interval for contention-based vehicular networks. In *Proceedings of the 2011 7th International Wireless Communications and Mobile Computing Conference*. IEEE, 2011, 783–788.
[79] Chrysostomos Chrysostomou, Constantinos Djouvas, and Lambros Lambrinos. 2011. Applying adaptive QoS-aware medium access control in priority-based vehicular ad hoc networks. In *Proceedings of the IEEE 2011 Symposium on Computers and Communications (ISCC)*. IEEE, 741–747.
[80] Chrysostomos Chrysostomou, Constantinos Djouvas, and Lambros Lambrinos. 2012. Dynamically adjusting the min-max contention window for providing quality of service in vehicular networks. In *Proceedings of the 2012 11th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*. IEEE, 16–23.

[81] Chrysostomos Chrysostomou, Constantinos Djouvas, and Lambros Lambrinos. 2014. Contention window adaptation for broadcast beaconing in vehicular ad hoc networks. In *Proceedings of the 2014 International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, 2014, 1039–1044.
[82] Irina Tal, Tianhua Zhu, and Gabriel-Miro Muntean. 2013. Short paper: On the potential of V2X communications in helping electric bicycles saving energy. In *Proceedings of the 2013 IEEE Vehicular Networking Conference*. IEEE, 218–221. IEEE.
[83] Irina Tal and Gabriel-Miro Muntean. 2013. V2X communication-based power saving strategy for electric bicycles. In *Proceedings of the 2013 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 1338–1343.
[84] Irina Tal, Bogdan Ciubotaru, and Gabriel-Miro Muntean. 2016. Vehicular communications-based speed advisory system for electric bicycles. *IEEE Transactions on Vehicular Technology*, 65, 6, 4129–4143.
[85] Irina Tal, Aida Olaru, and Gabriel-Miro Muntean. 2013. eWARPE-Energy-efficient weather-aware route planner for electric bicycles. In *Proceedings of the 2013 21st IEEE International Conference on Network Protocols (ICNP)*, IEEE, 1–6.
[86] Vicente Milanés, Joshué Pérez Rastelli, Enrique Onieva, and Carlos González. 2010. Controller for urban intersections based on wireless communications and fuzzy logic. *IEEE Transactions on Intelligent Transportation Systems* 11, 1, 243–248.
[87] Jialang Cheng, Weigang Wu, Jiannong Cao, and Keqin Li. 2017. Fuzzy group-based intersection control via vehicular networks for smart transportations. *IEEE Transactions on Industrial Informatics* 13, 2, 751–758.
[88] Hsin-Han Chiang, Yen-Lin Chen, Bing-Fei Wu, and Tsu-Tian Lee. 2014. Embedded driver-assistance system using multiple sensors for safe overtaking maneuver. *IEEE Systems Journal* 8, 3 (2014), 681–698.
[89] Kayhan Zrar Ghafoor, Kamalrulnizam Abu Bakar, Martijn van Eenennaam, Rashid Hafeez Khokhar, and Alberto J. Gonzalez. 2013. A fuzzy logic approach to beaconing for vehicular ad hoc networks. *Telecommunication Systems* 52, 1 (2013), 139–149.
[90] Boris S. Kerner. 2012. *The Physics of Traffic: Empirical Freeway Pattern Features, Engineering Applications, and Theory*. Springer.
[91] Kayhan Zrar Ghafoor, Alberto J. Gonzalez, Ramon Piney, Andre Rios, Jesus Alcober, and Kamalrulnizam Abu Bakar. 2011. Fuzzy redundancy adaptation and joint source-network coding for VANET video streaming. In *WWIC 2011*.
[92] Roger Immich, Eduardo Cerqueira, and Marilia Curado. 2015. Adaptive qoe-driven video transmission over vehicular ad-hoc networks. In *2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 227–232.
[93] Jens Heine, Michael Sylla, Ingmar Langer, Thomas Schramm, Bettina Abendroth, and Ralph Bruder. 2015. Algorithm for driver intention detection with fuzzy logic and edit distance. In *Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems*. IEEE, 1022–1027.
[94] Khalid Abdel Hafeez, Lian Zhao, Jon W. Mark, Xuemin Shen, and Zhisheng Niu. 2013. Distributed multichannel and mobility-aware cluster-based MAC protocol for vehicular ad hoc networks. *IEEE Transactions on vehicular Technology* 62, 8 (2013), 3886–3902.
[95] Chenn-Jung Huang, Yi-Ta Chuang, Dian-Xiu Yang, I-Fan Chen, You-Jia Chen, and Kai-Wen Hu. 2008. A mobility-aware link enhancement mechanism for vehicular ad hoc networks. *EURASIP Journal on Wireless Communications and Networking* 2008, 1 (2008), 1–10.
[96] Ramon Bauza, Javier Gozalvez, and Joaquin Sanchez-Soriano. 2010. Road traffic congestion detection through cooperative vehicle-to-vehicle communications. In *Proceedings of the 2010 IEEE 35th Conference on Local Computer Networks (LCN)*. IEEE, 606–612.
[97] Rola Naja and Roland Matta. 2014. Fuzzy logic ticket rate predictor for congestion control in vehicular networks. *Wireless Personal Communications* 79, 3 (2014), 1837–1858.
[98] Jelena Fiosina, Maxims Fiosins, and Jörg P. Müller. 2013. Big data processing and mining for next generation intelligent transportation systems. *Jurnal Teknologi* 63, 3.
[99] Aisha Siddiqa, Ibrahim Abaker TargioHashem, Ibrar Yaqoob, Mohsen Marjani, Shahabuddin Shamshirband, Abdullah Gani, and Fariza Nasaruddin. 2016. A survey of big data management: Taxonomy and state-of-the-art. *Journal of Network and Computer Applications* 71 (2016), 151–166.
[100] Li Bing and Keith CC Chan. 2014. A fuzzy logic approach for opinion mining on large scale twitter data. In *Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*. IEEE Computer Society, 652–657.
[101] Goldina Ghosh, Soumya Banerjee, and Neil Y. Yen. 2016. State transition in communication under social network: An analysis using fuzzy logic and density based clustering towards big data paradigm. *Future Generation Computer Systems* 65 (2016), 207–220.

Received January 2017; revised June 2017; accepted July 2017