A Comparative Survey of VANET Clustering Techniques

Craig Cooper*, Daniel Franklin†, Montserrat Ros‡, Farzad Safaei*, and Mehran Abolhasan†
*ICT Research Institute – University of Wollongong, Wollongong NSW 2500, Australia
email: {cc808|farzad}@uow.edu.au
†School of Computing and Communications – University of Technology, Sydney, Broadway NSW 2007, Australia
email: {daniel.franklin|mehran.abolhasan}@uts.edu.au
‡School of Electrical, Computer and Telecommunications Engineering – University of Wollongong, Wollongong NSW 2500, Australia
email: montse@uow.edu.au

***Abstract*—A vehicular ad hoc network (VANET) is a mobile ad hoc network (MANET) in which network nodes are vehicles – most commonly road vehicles. VANETs present a unique range of challenges and opportunities for routing protocols due to the semi-organised nature of vehicular movements subject to the constraints of road geometry and rules, and the obstacles which limit physical connectivity in urban environments. In particular, the problems of routing protocol reliability and scalability across large urban VANETs are currently the subject of intense research. Clustering can be used to improve routing scalability and reliability in VANETs, as it results in the distributed formation of hierarchical network structures by grouping vehicles together based on correlated spatial distribution and relative velocity. In addition to the benefits to routing, these groups can serve as the foundation for accident or congestion detection, inforomation dissemination and entertainment applications. This paper explores the design choices made in the development of clustering algorithms targeted at VANETs. It presents a taxonomy of the techniques applied to solve the problems of cluster head election, cluster affiliation and cluster management, and identifies new directions and recent trends in the design of these algorithms. Additionally, methodologies for validating clustering performance are reviewed, and a key shortcoming – the lack of realistic vehicular channel modelling – is identified. The importance of a rigorous and standardised performance evaluation regime utilising realistic vehicular channel models is demonstrated.**

***Index Terms*—Clustering, VANET, comparative analysis**

I. Introduction

In a Vehicular Ad hoc NETwork (VANET), participating vehicles are equipped with wireless transceiver which allow them to exchange data with other neighbouring vehicles, and, where necessary, to route packets via neighbouring vehicles to destinations that are not within direct communications range. One-hop connectivity to external infrastructure is not necessary, although stationary roadside units may also participate in a VANET. Such an architecture potentially enables the creation of applications ranging from improved traffic safety and congestion avoidance to in-car information and entertainment systems.

VANETs operate in a challenging communications environment, which to date have limited the practical deployment of the technology. VANETs are particularly susceptible to the hidden node problem; in addition, they must contend with limited spectral bandwidth and a highly variable channel influenced by both stationary and mobile obstructions and interference sources. In such an environment, infrastructure-based networks hold a significant advantage over ad hoc networks: access points allows optimal scheduling of channel access and distribution of network resources in a relatively simple manner, at the cost of needing to deploy a large number of access points throughout the intended coverage area. To achieve some of the benefits of an infrastructure-based network without the need for physical infrastructure, researchers have investigated the idea of *clustering* in VANETs, whereby a hierarchical network structure forms in a distributed manner throughout the network via some sort of *clustering algorithm*.

Since the introduction of the earliest clustering algorithms in the late 1980s, a wide range of approaches to the problem have been proposed in the context of MANETs in general and VANETs in particular. Each approach is directed toward different classes of problems and often towards specific applications envisioned for VANET technology. From the literature, it is apparent that attempts to validate proposed protocols often utilise simple channel and mobility models that do not realistically reflect the VANET environment; they also frequently compare new VANET clustering techniques to well-known MANET approaches that have not been designed for the characteristics of the VANET scenario, resulting in a favourable comparison but without direct reference to other state-of-the art approaches. In this paper, we intend to address these problems in the literature and provide a clear taxonomy and comparative performance review for all clustering strategies presently employed in VANETs.

A. Structure and Contributions

This review paper makes three significant contributions to the field of VANET clustering:

1) The main applications of clustering are described, with *general purpose* clustering algorithms being considered separately to *application-specific* algorithms; this discussion is presented in Section III.
2) The methods by which the major contemporary and historical algorithms approach the main aspects of the

clustering problem are discussed in detail, in particular: how the cluster head is elected, how unclustered nodes affiliate with a head, and how cluster heads manage interactions with other clusters. This review suggests a taxonomy of VANET clustering techniques which is structured around these design questions; although VANET clustering taxonomies have previously been described in the literature, most notably in [1] and [2], these taxonomies have combined application and design approach within their classification system. This publication makes a clear distinction between application and design. The proposed taxonomy classifies protocols according only to how the algorithm solves the facets of the clustering problem, which is a novel approach to the subject. The survey is presented in Section IV.

3) Finally, the problem of evaluating and comparing the performance of clustering algorithms is considered. Section V discusses the simulator frameworks, channel models, and approaches for comparing protocols against competing algorithms. To the knowledge of the authors, this is the first such analysis of clustering benchmarking.

A number of potential new directions for clustering research are identified. The paper concludes with a call for standardised verification methodologies, and a commentary regarding the approach to VANET research.

II. Clustering Concepts

A. Development of VANET technology

Serious interest in VANET technology started to develop in the early 1990s, and has increased in recent years. The need for standards in this area become apparent after the emergence of electronic road tolling systems based on a variety of proprietary active RFID transponder systems. Realising that this simple concept could be generalised to support a variety of vehicular communications applications, several major manufacturers of electronic toll collection systems established the Dedicated Short-Range Communications (DSRC) Industry Consortium, which worked towards a common physical layer standard for short-range vehicular communication. This effort led to the development of a physical layer based on the ad hoc mode of IEEE 802.11 operating in the 5.9 GHz band; this is known as the Wireless Access in Vehicular Environments (WAVE) / Dedicated Short Range Communications (DSRC) standard and has been formalised by the IEEE as 802.11p [3], [4]. Building on this physical layer, a range of standards for various other parts of the network stack are currently under development by the IEEE, ISO, ETSI and other bodies – in particular, the IEEE 1609 working group is developing standards for security and applications for VANETs built on top of 802.11p [5]–[8].

Over the last few years, projects such as Keystone Architecture Required for European Networks (KAREN) [9] attempted to provide a framework whereby policies and plans regarding vehicular technology could be translated into system specifications. The EU co-funded Cooperative Vehicle-Infrastructure Systems (CVIS) project aims to construct a VANET architecture for the provision of a number of services, with consideration of roll-out and public adoption of the technology [10]. The Cooperative Systems for Intelligent Road Safety (COOPERS) project aims to develop telematics applications for vehicular communication systems, and construct a cooperative traffic management between vehicle and infrastructure [11]. Another project, SAFESPOT, works toward developing systems for reoad and vehicle safety [12]. Meanwhile, the NSF Project is focused on developing safety-specific applications in Vehicular Networks [13].

VANETs offer exciting opportunities in the areas of traffic safety and road network efficiency. Cars can avoid collisions by conversing and exchanging information regarding driver intention at a level not possible with basic communication mechanisms such as turn signals. Emergency vehicles can alert cars ahead, potentially well beyond the human visual or auditory range, so that a path can be cleared ahead of the vehicle, thereby reducing response times of ambulences and police cars. Traffic conditions can be monitored and congestion alerts issued in real time to enable vehicle flows to be re-routed around obstructions. When merged with emerging technologies in driver awareness monitoring, vehicles could alert a central authority when a driver is tired or intoxicated or when another vehicle appears to be driving abnormally; most importantly of all, VANETs offer a form of real-time accountability when accidents *do* occur. VANET technology may also be able to reduce the environmental impact of vehicle emissions, by selecting routes that result in less fuel consumption and lower emissions [14]. However, a number of technical and ethical issues arise when considering such applications, such as security and privacy of a driver, and the ability to prevent malicious agents from interfering with the network's operation, e.g. through modification, jamming or fraudulent generation of vehicular traffic data. These issues will become even more critical with the roll-out of autonomous or semi-autonomous vehicles such as Google Car [15], especially with respect to accountability when the autonomous navigation systems fail. Several detailed studies of these issues have previously been presented in the literature [16], [17].

B. Development of Clustering

A VANET clustering algorithm works by associating mobile nodes into groups – *clusters* – according to some rule set, and selecting a node known as the *cluster head* (CH) to mediate between the cluster and the rest of the network in much the same way as an infrastructure wireless access point. The specific functions of the cluster head differ depending on the application, as does the method by which it is selected. The *clustering algorithm* used to associate nodes with clusters should ideally be robust to node mobility and sudden changes in network and cluster topology, and should provide reliable end-to-end communication across the VANET.

The earliest notable work on clustering began with the DARPA packet-radio network [19], the intention being to autonomously form subnets within a Mobile Ad hoc NETwork (MANET) to facilitate the distribution of network resources. This work was built upon by Gerla et al., who proposed the popular Lowest ID and Highest Degree (LID/HD) clustering algorithms for MANETs [18]. Mobility Clustering (MOBIC)

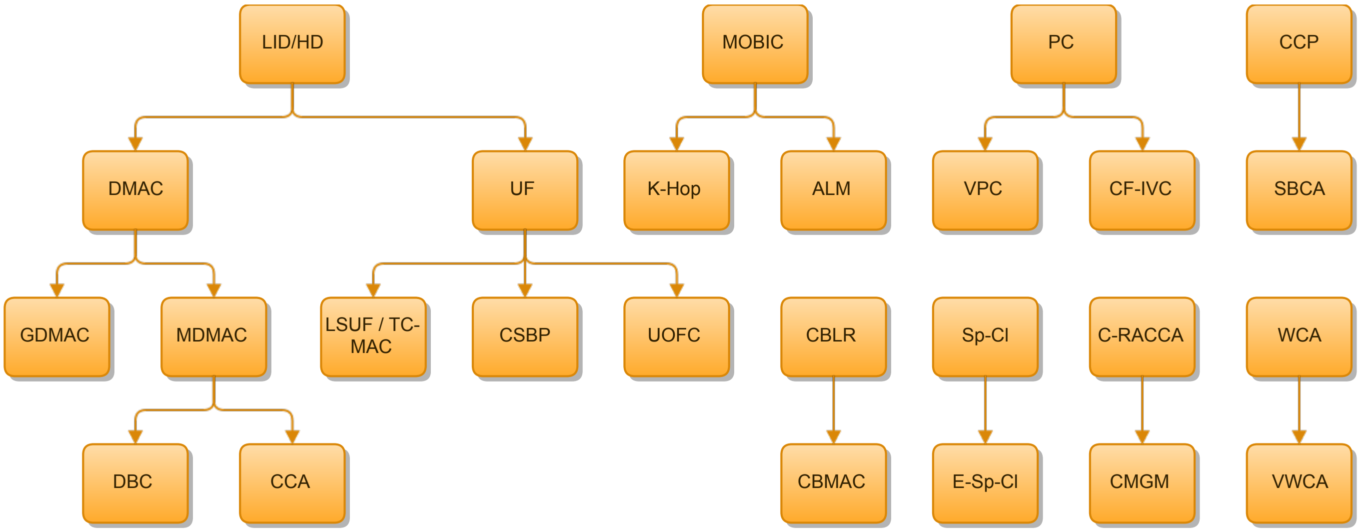

Fig. 1: Lineage of VANET clustering algorithms. LID/HD [18] is noted as the earliest significant VANET clustering scheme, as can be seen from its influence on subsequent algorithms

was later presented in [20], attempting to incorporate mobility considerations into the clustering phase. Several other algorithms including Distributed Group Mobility Adaptive (DGMA) clustering [21] and MobHiD [22] were later proposed, each designed for clustering in MANETs and demonstrating progressive improvements in performance under simple mobility models such as random wayoint. As a result of this practice of building upon earlier designs, a lineage of various algorithms developed, which is shown in Figure 1.

As a result of the mobility and channel conditions of VANETs – which distinguish them from MANET scenarios – the approaches to clustering have been adapted to these unique properties. Methods for establishing clusters, detecting and affiliating with established clusters, and maintaining existing clusters in VANETs range from channel monitoring, mobility prediction, machine learning, and security assessment. More recently, methods of grouping vehicles according to their shape profile, dimensions, and classification have arisen, e.g. [23], [24]. Methods of extending cluster lifetime by monitoring changes in vehicle topology have been investigated e.g. [25].

As VANET research slowly moved away from its MANET origins, the methodologies whereby a cluster was formed diversified. Algorithms were designed with specific applications in mind, including peer-to-peer (P2P) file sharing and channel access management. Specific cluster head selection and member affiliation schemes were adapted to or invented for these applications. The diversity in clustering strategies has grown to the point where adequately classifying these algorithms and identifying new avenues for research is difficult.

C. Anatomy of a Clustering Algorithm

A series of fundamental procedures are involved in the formation and maintenance of clusters, which may need to be repeated depending on the rules of the algorithm and the mobility dynamics of the network. The general procedural flow of a clustering algorithm is shown in Figure 2. Nodes participating in or seeking to participate in a cluster will typically perform some or all of the procedures described below, with references to Figure 2.

1) **Neighbourhood Discovery:** When a vehicle initially joins the road network and decides that it is willing to participate in a VANET, its communications system will be turned on and the node is considered to have entered a network, initially with only itself as a member. A node will begin by announcing its existence to its neighbours through a periodic broadcast, while simultaneously gathering similar information from its n-hop neighbours – either passively by listening for broadcasts, or actively through neighbour solicitation requests (1). This information typically includes position information for the neighbouring nodes, and is stored in a neighbour table for use by the clustering algorithm.
2) **Cluster Head Selection:** After gathering data about its environment, a node will then examine the neighbour table to find a suitable node to act as its CH (2). The role of the cluster head varies depending on the clustering algorithm – it may include routing or relaying functions, and it may also be responsible for determination of cluster membership. During this process a node will also assess its own suitability to be a CH. If the chosen CH is found within the neighbour table (3), a node will proceed to step 3; otherwise, if the node itself is best suited to be CH, it will proceed to step 4.
3) **Affiliation:** The node will contact the neighbour it determined to be the optimal CH from its own perspective, and attempt to become a member of that cluster (4). Some algorithms may require the addressee to already be an established CH, while others may allow the addressee to be an unclustered or regular cluster member. There may be an additional step where a positive or negative acknowledgement of the affiliation request is returned to the joining node, possibly followed by an authentication

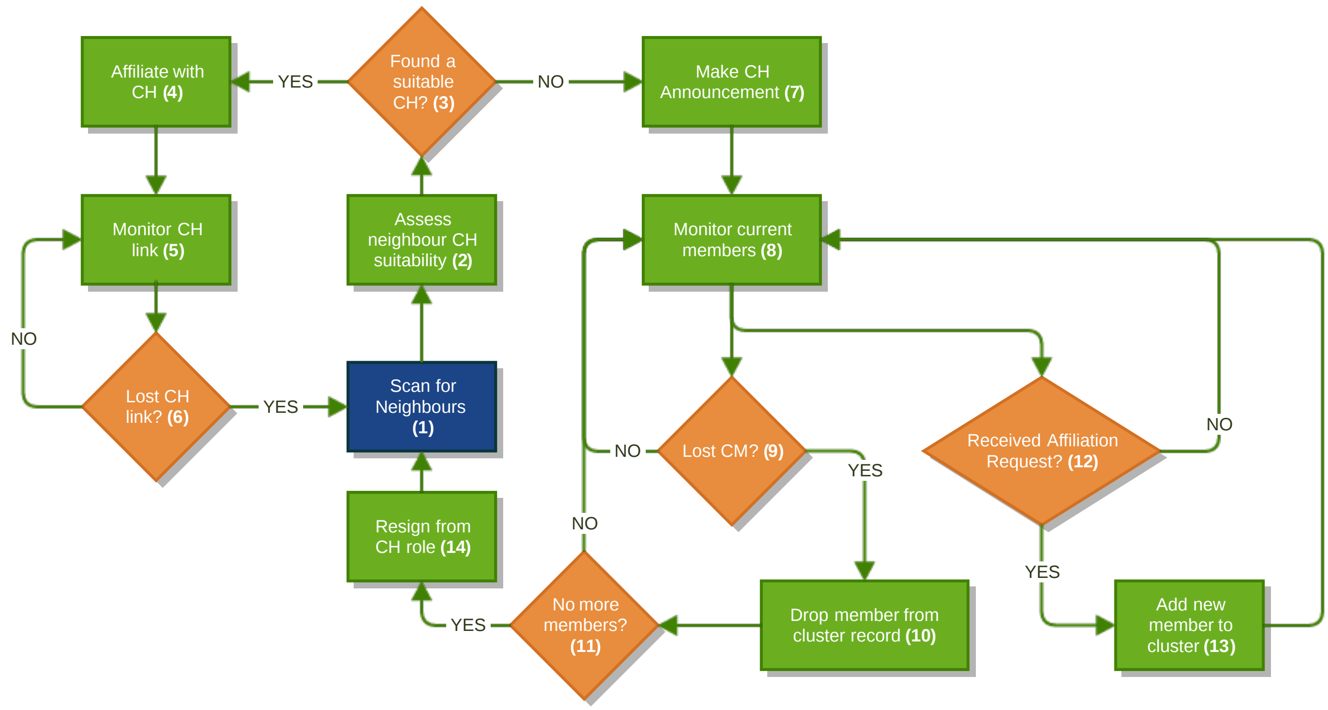

Fig. 2: The basic flow of a clustering algorithm. There are variations on this method, but each algorithm in this survey follows this conceptual flow.

step in the case of algorithms targeted toward security-sensitive applications. Once a node has become a cluster member (12,13), it will enter step 5b.

4) **Announcement:** The node, having determined itself to be the most suitable CH, may then send out an announcement message to its neighbours to begin the formation and affiliation process (7). When the node has accrued cluster members, it proceed to step 5a.
5) **Maintenance:** This step is different based on whether the node has become a CH or member:
   a) *As a Head:* the node will poll the members of its cluster and assess the status of the cluster (8). Several algorithms have multi-step maintenance processes that allow clusters to change heads, merge with neighbouring clusters, and track lost links to members, e.g. due to transient disconnections (9,10). A number of events may change the state of the cluster: if a CH loses all of its members (11), the cluster is said to have *died* (14), and the node returns to step 1; alternatively, one cluster may merge with another, and the CH of the smaller cluster may become an ordinary member of the new, larger cluster. This is common in algorithms that place an emphasis on creating large clusters for increased coverage. In this case, the node will go to step 5b.
   b) *As a Member:* the node will periodically evaluate its link to its CH (5), either by waiting for a poll frame from the CH, or by actively sending "alive" messages. If the node's link to its CH fails (6), it will return to step 1. If the node receives an affiliation request from an unclustered node, it may withdraw from its parent cluster to become a CH and continue to step 5a; or in the case of hierarchical algorithms, it may transition to a combined state where it is a head of a nested cluster. In this case, it may perform steps 5b and 5a simultaneously.

These steps are common across clustering algorithms for MANETs and VANETs. The key difference between algorithms for the two classes of network is in the methods of selecting the CH. MANET clustering algorithms use generic methods of CH selection that typically consider location, velocity, or node density. In practice, there are many clustering algorithms that work well in generic unconstrained MANET scenarios, but which perform poorly in vehicular scenarios due to the unique channel and mobility dynamics of such environments. By contrast, VANET-specific clusterng algorithms are aware of the restrictions on node movement imposed by the road network and normal traffic flow behaviour, and therefore utilise information such as traffic mobility metrics and lane structure which are useful in characterising the position and behaviour of vehicles on a road. For instance, although two vehicles may be well within theoretical free-space communication range, they may be unable to communicate because a building, large vehicle, or some other obstacle is in the way. Clustering algorithms that are effective in achieving steps 3 and 5 in a vehicular scenario may therefore be quite different from those which are used in MANETs.

The question of how to optimally design algorithms to carry out the affiliation and maintenance steps in a generic vehicular network is not yet settled. Most published work on VANETs focus on CH selection, with comparatively few addressing other important aspects of clustering algorithm design. Furthermore, minimal comparative analysis of designs has been performed between VANET algorithms; rather, proposed solutions to the clustering problem are most commonly compared with MANET algorithms. Therefore, the remainder of this paper will focus on comparing a variety of proposed VANET clustering algorithms on a common basis.

III. Applications of Clustering

Recent VANET clustering surveys principally categorise algorithms in terms of their primary *application* [1], [2], [80]. This is a logical approach since the design of a clustering algorithm is often influenced by the application for which it is to be used. This section will introduce the surveyed algorithms in terms of their principal application; section IV will then take a different approach, developing an algorithm taxonomy on the basis of common aspects of algorithm design (algorithm *behaviour* and *function* as opposed to *purpose*).

Several application-specific algorithms have been proposed in the literature, while others aim to satisfy the requirements of a very broad and varied range of applications. Table I summarises the algorithms discussed in the survey, grouped by their principal application.

A. Applications

The following subsections describe the application classes as shown in Table I, and list several examples of each class.

1) General-Purpose Algorithms: A significant subset of proposed clustering algorithms can be classified as general purpose, in that they were designed for the purpose of grouping vehicles without any specific application in mind. General purpose algorithms tend to be designed with a high emphasis on robustness in the face of vehicular mobility. They are intended to provide generic platforms upon which a range of applications – for example, routing, vehicular safety, and traffic management – can be built. Gerla's LID/HD is an early example of general purpose MANET clustering [18]. Later work by Chiang et al. extended Gerla's algorithm with support for routing [81], and Fan et al. have adapted the technique for channel access management [57].

Most general-purpose protocols in this survey were based on MANET algorithms, and had been modified to account for vehicular mobility. An example of this is the Distributed Clustering Algorithm (DCA) [26]. The authors based their work on LID/HD, but replaced the use of ID or Degree with a weighted sum of metrics such as inter-node distance. This is a simple message-driven method, which assumes that nodes are stationary during the clustering phase. The concept was then extended to account for mobility metrics, resulting in the Distributed Mobility-Aware Clustering (DMAC) algorithm. DMAC incorporated a mobility-aware maintenance phase, in which member nodes detected new and failed links, adjusting the cluster structure as necessary.

DMAC was specifically designed for ad hoc networks with mobile nodes that are not specifically road vehicles. Ghosh et al. evaluated DMAC performance and behaviour under three different mobility models: Random Way Point (RWP), Brownian motion (BM), and Manhattan grid (MG) [27]. The authors found that DMAC tended to create short-lived clusters, with cluster lifespan lasting a maximum of approximately 100 seconds. BM resulted in clusters lasted the shortest time – less than one minute on average – although it is noted that average node speed has little impact on cluster lifespan. Cluster member stability, represented by *cluster residence times*, was similarly poor, with high rates of reaffiliation under all mobility models. The authors then introduced a derivative of DMAC, *Generalised DMAC* (G-DMAC), which incorporated additional mobility measurements in the cluster head selection mechanism, which was shown to improve the efficiency of cluster formation and stability.

Of all the algorithms discussed in this survey, the general-purpose algorithms most frequently utilise vehicular mobility metrics for clustering decisions, such as vehicle destination, intended turning direction at the next intersection, and relative speeds of different vehicles. Section IV gives a detailed discussion of approaches to clustering that calculate cluster head selection metrics based on vehicular mobility.

2) Routing: Clustering can serve as a mechanism for dynamically building a hierarchical infrastructure-like overlay on top of an underlying ad hoc network, which can be used to route packets. Most often, the cluster head performs the routing between its members and neighbouring clusters. As a result, VANET routing algorithms tend to form hierarchieical structures in one of three ways. Firstly, one cluster head can also be a member of another cluster, creating a tree-like structure. A classic example of this approach is Robust Mobility-Aware Clustering (RMAC) [44]. In the second major structure type, a dynamic backbone structure is formed, where all cluster heads exchange routing message with or via each other; Dynamic Backbone-Assisted MAC (DBA-MAC) is a representative example of this method [49]. Finally, a hierarchy can be constructed based on gateway nodes, which are members of multiple clusters. Cluster-Based Location Routing (CBLR) is an example of such a scheme [82], [83].

An interesting subset of the routing-oriented clustering algorithms listed in Table I consists of Passive Clustering (PC) [45] and its derivatives. These have been designed as extensions to the classic MANET routing protocol Ad hoc On-demand Distance Vector (AODV) [84], piggy-backing their control data on existing traffic. The passive approach is discussed further in Section IV-B.

3) Channel Access Management: An advantage of clustering is that it segments a VANET in a similar way to subnetworks in an infrastructure network. This allows a cluster head to coordinate activities amongst its members, in particular, by scheduling channel access. Algorithms intended for this application typically utilise a TDMA-based approach, whereby the cluster head assigns timeslots to its members – for example, [64]. Such clustering algorithms allow QoS assurance, and eliminate the need for RTS/CTS frames, which have been shown to be ineffective in VANETs due to the hidden node problem [85].

4) Security: The need for nodes to affiliate with cluster heads in order to join their clusters provides a potential

TABLE I: List of clustering algorithms in this survey

Application	Algorithm Name	Acronym	Citation
General Purpose	Distributed Mobility Aware Clustering	DMAC	[26]
	Generalised Distributed Mobility Aware Clustering	GDMAC	[27]
	Modified Distributed Mobility Aware Clustering	MDMAC	[28]
	Density Based Clustering	DBC	[29], [30]
	Criticality-based Clustering Algorithm	CCA	[31]
	K-hop	K-hop	[32]
	Aggregate Local Mobility	ALM	[33]
	Spring Clustering	Sp-Cl	[34]
	Enhanced Spring Clustering	E-Sp-Cl	[23]
	Position-based Prioritised Clustering	PPC	[35]
	Utility Function	UF	[36]
	Threshold-Based Clustering	TBC	[37], [38]
	Fuzzy-Logic-Based Algorithm	FLBA	[39]
	User Oriented Fuzzy-logic-based Clustering	UOFC	[40]
	Mean Connection Time Clustering	MCTC	[41]
	Neighbour Mobility-based Clustering Scheme	NMCS	[42]
	Distributed Multihop Clustering using Neighbourhood Follow	DMCNF	[25]
Routing	Cluster-Based Location Routing	CBLR	[43]
	Robust Mobility Adaptive Clustering	RMAC	[44]
	Mobile Infrastructure in VANET	MI-VANET	[24]
	Passive Clustering	PC	[45]
	Vehicular Passive Clustering	VPC	[46]
	Cluster Formation for IVC	CF-IVC	[47]
	Trust-dependant Ant Colony Routing	TACR	[48]
	Dynamic Backbone-Assisted MAC	DBA-MAC	[49]
	Cellular Automata Clustering	CAC	[50]
	Multihoming Clustering Algorithm for VANETs	MCA-VANET	[51]
Channel Access Management	Clustering based on Direction in Vehicular Environments	C-DRIVE	[52]–[54]
	Adaptable Mobility-Aware Clustering Algorithm based on Destination	AMACAD	[55], [56]
	Clustering with Scalability for Broadcast Performance	CSBP	[57]
	Affinity Propagation in Vehicular Environments	APROVE	[58], [59]
	Cluster-Based MAC	CBMAC	[60]
	Hierarchical Clustering Algorithm	HCA	[61], [62]
	TDMA Cluster-based MAC	TC-MAC	[63]–[65]
Security	Vehicular Weighted Clustering Algorithm	VWCA	[66]
	Cluster-Based Public Key Infrastructure	CBPKI	[67]
	Agent Learning-based Clustering Algorithm	ALCA	[68]
QoS Assurance	Cluster Configuration Protocol	CCP	[69], [70]
	Stability Based Clustering Algorithm	SBCA	[71]
	Multi-Channel Cooperative Clustering-based MAC	MCC-MAC	[72]
Traffic Safety	Cluster-based Risk-Aware Cooperative Collision Avoidance	C-RACCA	[73]
Topology Discovery	Clustering Protocol for Topology Discovery	CPTD	[74]
Combination with cellular infrastructure	Cluster-based Multimetric adaptive Gateway Management mechanism	CMGM	[75], [76]
	Vehicular Multi-hop algorithm for Stable Clustering	VMaSC	[77] [78]
	Fuzzy QoS-balancing Gateway Selection	FQGWS	[79]

VANET security mechanism. Several of the surveyed algorithms employ multistate authentication mechanisms in the affiliation process to protect against malicious vehicles, and often include degree-of-trust or historical reliability estiamtes in the computation of the cluster head selection metric.

5) Vehicular Network Topology Discovery: Multihop clustering structures provide a useful means of analysing the topology of the network of road vehicles. Cluster heads collect information from affiliated nodes and neighbouring clusters in order to build maps of connectivity between nodes in the network. For a VANET, this information is not only important to vehicular communications protocols, but also provides valuable information to vehicular traffic management systems – for example, identifying regions of traffic congestion by analysing patterns of vehicular movement. This information can either be distributed via routing protocol updates or via a dedicated information dissemination protocol.

6) Traffic Safety: A VANET can use a clustering structure to anticipate vehicular collisions. Cluster-based Risk-Aware Cooperative Collision Avoidance (C-RACCA) incorporates an estiamte of a vehicle's expected braking time and its corresponding collision-avoidance potential into its cluster head selection scheme, ensuring that each node has sufficient separation from the vehicle in front to be able to brake and avoid a collision [73]. When an emergency situation is identified, a warning is disseminated through the cluster and recipient vehicles take appropriate action – either automatic actuation of vehicle controls or some sort of emergency warning for the driver – depending on the event type specified in the warning message [73].

7) Combination with Cellular Infrastructure: In the literature, VANETs are often assumed to be integrated with cellular infrastructure to support access to the Internet. Cluster-based Multimetric adaptive Gateway Management (CMGM) uses a clustering structure to facilitate integration while mitigating load on the cellular network [75], [76]. The algorithm selects a node equipped with cellular connectivity as an Internet gateway for clusters. The selection is made based on cellular received signal strength, and potentially removes the need for dedicated stationary road-side units. The Fuzzy QoS-balancing Gateway Selection (FQGWS) algorithm uses signal quality, link expiration time, and load profile to select an appropriate gateway between the cluster and a nearby cellular network node [79]. Vehicular Multi-hop algorithm for Stable Clustering (VMaSC), first proposed in [77], extends prior work in [78] to integrate the cluster structure with a network of LTE base stations for data dissemination purposes.

B. Discussion

The most significant previous surveys of clustering methods presented in [1] and [2] gave a comprehensive overview of the literature, as of 2012 and 2014 respectively, but did so while combining analysis of applications and design methodology in their classification system. To date, a specific review of each of the most important aspects of the clustering problem and the most prominent solutions proposed for each has not been published. The following section details a taxonomy of VANET clustering techniques organised according to the strategies employed toward each of the stages detailed in Section II.

IV. Taxonomy of VANET Clustering Techniques

Establishing a clustering structure is a multifaceted problem, due to the unique mobility and channel dynamics of the VANET environment. First a node must obtain information about its neighbouring vehicles, in order to identify potential cluster heads. The criteria for head selection includes mobility, signal quality, and capability to provide specific traffic and infotainment functions. The means whereby this criteria is expressed also varies: an algorithm may specify criteria in the form of a scalar indicating its fitness to be a cluster head, or it may compute a countdown timer after which it announces its self-appointment of the role. There are also multiple methods whereby a node joins a cluster, a cluster head maintains its members, and determines when the cluster should be disbanded or merged with another cluster.

This section presents a taxonomy of the approaches to each of these problems. Each method is presented with an analysis of their advantages and disadvantages with respect to the VANET environment. Additional discussion is given on how disadvantageous approaches have been adapted or modified to improve their applicability to vehicular networks.

A. Cluster Head Selection Strategy

The method whereby a cluster head is selected, and the parameters used in the decision, are described in this section. Three main assessment mechanisms are identified in the literature, based on self-reported weighted network metrics, precedence, and timers respectively. These each use various parameters of the network in an attempt to find the node most suitable to be cluster head.

1) Weighted Network Metrics: The most common approach for cluster head selection requires each node to calculate an index quantifying its fitness to act as a cluster head for its neighbours, and to advertise this index within its 1-hop neighbourhood. Nodes wishing to affiliate with a cluster head will then rank all neighbours in their neighbour table and request association with the most highly-ranked candidate node, which may be the node itself. The index is typically a weighted sum of various network metrics, such as the degree of connectivity, link stability, node uptime etc., with the weights reflecting the relative importance of the selected metrics for this network. Twenty-eight of the surveyed algorithms use this approach, making it by far the most popular selection mechanism. The fitness index or score is sometimes itself referred to in the literature as a *node weight*, for instance in [26]; however, in this survey we use the term 'weight' in the traditional sense of coefficients used to adjust the proportions of different metrics to compute the overall fitness score.

The necessary information is distributed to neighbouring nodes via one of the mechanisms discussed in Section IV-E. The criteria defining "best score" may mean the lowest among the neighbours, as with Adaptable Mobility-Aware Clustering Algorithm based on Destination (AMACAD) [55] and Vehicular Weighted Clustering Algorithm (VWCA) [66]; "best score" may alternatively mean highest, as with Mean Connection Time Clustering (MCTC) [41] and Multihoming Clustering Algorithm for VANETs (MCA-VANET) [51]. The common metrics for the selection index are discussed in Section IV-C. Table II contains all the algorithms applying this selection strategy.

The advantage of this approach is its simplicity and its ability to guarantee that only a single cluster head exists within a node's communication range. An additional advantage is that the relative importance of each network metric can be tuned for a specific network in the calculation of the weighted fitness index by adjusting the value of the weight coefficients.

Cluster head selection strategies based on weighted network metrics need to contend with the dynamism inherent in the VANET environment, which can cause the selection metrics, and hence the optimal choice of cluster head, to change over

TABLE II: Algorithms using Weighted Metric as the Selection Strategy

Algorithm	**Selection Metric(s)**	**Neighbour Discovery**	**Affiliation Handshake**	**Cluster Head Handoff**
DMAC	ID, Degree	Hello	No	No
MDMAC	ID, Degree, LET	Hello	No	No
CCA	Network Criticality	Hello	No	No
K-hop	Propagation Delay Ratio	Hello	No	No
ALM	Distance	Hello	No	No
PPC	ID, Travel Time, Average Relative Velocity	Hello	No	No
UF	Distance, Relative Velocity, ID, Degree	Hello	No	No
CSBP	Distance, Relative Velocity, ID, Degree	Hello	No	No
TC-MAC	Distance, Relative Velocity, Turning direction	Hello	No	No
UOFC	Distance, Relative Velocity, Acceleration, Driver Intention	Hello	No	No
VWCA	Direction, Degree, Distrust Level, Velocity	Hello	No	No
MCTC	Mean Connection Time	Hello	No	No
MCA-VANET	Consecutive Hello message reception	Hello	No	No
NMCS	Change in degree	Hello	No	No
FLBA	Relative Velocity	Hello	No	Yes
MI-VANET	Vehicle Class	Hello	Yes	No
Sp-Cl	Distance and Relative Velocity, forming weight analogous to Coulomb's Law	Hello	Yes	No
E-Sp-Cl	Distance, Relative Velocity, and Vehicle Height, forming weight analogous to Coulomb's Law	Hello	Yes	No
APROVE	Distance, Relative Velocity	Hello	Yes	No
CBPKI	Trust Level	Hello	Yes	No
AMACAD	Distance, Relative Velocity, Destination	Hello	Yes	Yes
DMCNF	Propagation Delay Ratio, Number of following cars	Hello	Yes	Yes
GDMAC	ID, Degree	Hello	No	No
TBC	Distance, Relative Velocity	Inquiry	No	No
TACR	Distance, Relative Velocity, Traffic Obedience, Packet Forwarding Reputation	Inquiry	No	No
DBC	SNR, Distance, Velocity	Inquiry	Yes	No
CAC	Driver Interest	Unclear	Unclear	No
ALCA	Velocity	Unclear	Unclear	No

time. Often the changes are only temporary; therefore, several authors have applied low-pass filters to the cluster head fitness indices to prevent transient changes from causing unnecessary reclustering [30], [39]. In [68], a learning mechanism is used to select cluster heads based on the relative velocity of vehicles. Agents are deployed at junctions to determine the velocity and behaviour of nodes entering/leaving the intersection. The learning algorithm rewards positive results, such as selection of a cluster head which exhibits a long lifespan and high packet delivery ratio for routed packets, by an increase in a learning factor, while penalising negative results, such as a poorly-performing cluster head, with a decrease. This progressively results in better cluster heads being selected, while resisting short-lived changes in cluster head selection metrics. In Criticality-based Clustering Algorithm (CCA) [31] and G-DMAC [27], a hysteresis threshold is used to prevent thrashing of the cluster structure; a cluster member will only change to a new cluster head if the proposed new cluster head's selection metric exceeds that of the current cluster head by a set threshold. Aggregate Local Mobility (ALM) [33] employs a contention timer to prevent reclustering when two clusters move within communication range of each other, and to avoid the formation of single-member clusters. Position-based Prioritised Clustering (PPC) [35] specifies a *dismiss threshold*, which prevents the merging of clusters until they are physically closer than the specified threshold distance.

Several possible issues arise with this cluster head selection methodology. Since the index representing a node's suitability to lead the entire cluster is based on its relationship with its entire neighbourhood, a node may be identified as the best candidate as a cluster head for a particular node even when its connectivity is quite marginal, because it has good connectivity with the other nodes in the cluster. Additionally, different nodes may be visible to a different subset of the entire network, and small movements may result in a very different assessment of the best node to select as cluster head. Depending on the distribution, density and behaviour of traffic, this may result in poor clustering performance for some nodes, particularly at the margins of the network.

2) Precedence: Unlike the weighted-metric fitness index approach, in which nodes advertise their own fitness to be a cluster head, in precedence-based schemes, a node seeking a cluster head will make its own assessment of the suitability of potential cluster heads, again based on a weighted sum of metrics. As the assessment is made by the cluster-head-seeking node, it can act selfishly and choose a cluster head which suits its requirements rather than the aggregate of the other nodes in the network. An example of this so-called precedence-based approach is RMAC [44], which sorts its neighbours in order of relative distance, relative velocity, and whether or not a neighbour is already a cluster head; another is Multi-Channel Cooperative Clustering-based MAC (MCC-MAC) [72], an 802.11p extension in which vehicles use the Control Channel to search for cluster heads with the lowest velocity relative to their own. Precedence-based algorithms allow for higher cluster member stability, particularly for marginal nodes, by allowing each node to choose the cluster head that best suits them. Precedence-based algorithms are thus best suited to forming hierarchical structures. Table III contains the surveyed algorithms that employ the Precedence method.

This higher stability comes at the expense of greater control overhead, as a larger number of cluster heads result from the selfish approach. Additionally, if cluster members have the capability to host other nodes (thereby forming a hierarchical structure), there may be a tendency to form chains rather than trees. While this can be advantageous in certain scenarios, such as platoon formation on a narrow road, it can result in snaking structures of several cluster heads within a single transmission range, as shown in Figure 3. When such an algorithm is used to support VANET routing, boundary nodes can be several more hops apart than necessary, increasing the hop-count and reducing data throughput.

Vehicular Multi-hop Algorithm for Stable Clustering (VMaSC) [77] aims to mitigate this problem: although it employs multi-hop affiliation instead of a hierarchical scheme, it applies a configurable limit on the number of hops between a head and its members, thereby preventing the formation of snaking structures.

3) Timer: A timer-based approach, employed by the algorithms listed in Table IV, has nodes remaining in an unclustered state for a set period of time, waiting for a cluster head to announce itself. If a node detects the presence of a cluster head within the timeout period, it affiliates with it; otherwise, if the timer expires, the node declares itself a cluster head. The waiting period may be set statically, as in [43] and [83], or randomly as in [62]. A potential cluster head may be further evaluated by nodes that overhear its broadcasts to determine its suitability as their cluster head [69], for example, a node may affiliate with a cluster head if the RSS of the received broadcasts exceeds a set threshold. In [71], relative velocity is also used to evaluate a candidate cluster head, and link expiration time is used in [49]. The principal advantages are the speed and simplicity of the algorithm; a neighbourhood discovery phase is not required to select cluster heads, and the channel bandwidth will only be consumed by cluster head announcement broadcasts.

However, in networks with regions of high node density, this selection scheme can fall victim to the well-known hidden node problem. Multiple nodes can reach the end of their timers simultaneously or near-simultaneously, and broadcast at the same time. Thus, nodes within range of multiple candidate cluster heads will experience packet collisions and their ability to detect nearby cluster heads within their own time period will be impaired.

B. Passive Clustering

Weighted-metric, precedence and timer-based clustering techniques have the common trait of being *active* approaches. That is, they use a common channel for cluster formation and maintenance together with other network traffic. This can cause contention between clustering traffic and routing traffic, as both compete for finite channel bandwidth, even when clustering is used to facilitate routing. The algorithms listed in Table V employ a different approach known as Passive Clustering (PC).

The advantage of passive techniques is that there is no contention between clustering and routing traffic, allowing the former to assist in achieving the goal of the latter. An active clustering strategy can significantly interfere with the effectiveness of an ad hoc routing protocol due to the additional control overhead required for cluster formation and reformation. By contrast, passive approaches exploit the synergy between the two systems to achieve cluster formation while avoiding the need for extra traffic.

First proposed in [45], PC addresses this problem by piggy-backing cluster control data on outgoing routing traffic, eliminating the need for separate broadcast phases for clustering. The authors build their approach on AODV [84], utilising a “first declaration wins” approach for cluster head selection. Because the architecture waits for data to be sent from the upper layers, vehicles running active data-generating services are more likely to be chosen as cluster heads. PC was compared with LID-assisted AODV. PC significantly improves packet delivery ratio while reducing end-to-end delay for a given node density (or conversely, maintaining high throughput with low delay for higher node densities) compared to LID-assisted AODV.

VANET Passive Clustering (VPC) is an extension to PC proposed in [46]. The protocol is also built on top of AODV, but discards the “first declaration wins” selection approach in favour of cluster head selection based on channel quality measurements, which results in better performance than PC.

TABLE III: Algorithms using Precedence as the Selection Strategy

Algorithm	Selection Metric(s)	Neighbour Discovery	Affiliation Handshake	Cluster Head Handoff
C-DRIVE	Front of platoon, Direction of travel	Hello	No	Yes
MCC-MAC	Relative Velocity	Hello	Yes	No
VMaSC	Relative Velocity	Hello	Yes	No
CPTD	ID, Degree, Link Expiration Time	Hello	Yes	No
RMAC	Cluster head status, Distance, Velocity, Size	Inquiry	Yes	No
CMGM	Relative velocity, Deceleration capability, In front or behind	Inquiry	Yes	No
C-RACCA	Relative velocity, Deceleration capability, In front or behind	Inquiry	Yes	Yes

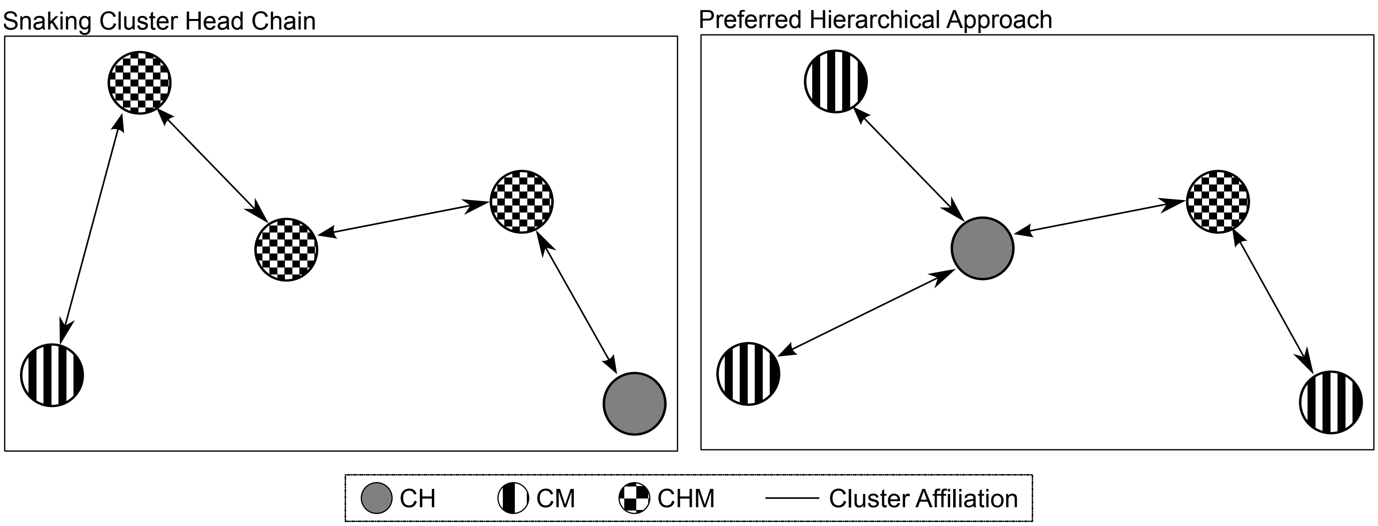

Fig. 3: An illustration of the snaking cluster head chain. The precedence approach may result in chains of subclusters, in which cluster heads are closely grouped together, as seen in the left-hand figure. The boundary nodes are four hops apart in this example. On the right a master cluster head is shown, one of whose members is also a subcluster head. In this scenario, the boundary nodes are only three hops apart. A purely selfish method may not result in the best clustering structure.

While these techniques mitigate the contention between clustering and other VANET processes, existing network traffic is required in order to provide a carriage service for the clustering data, therefore cluster formation will be inhibited by low levels of network traffic. In practice, network traffic is usually sporadic, meaning updates to neighbour data required for cluster formation and maintenance will also be sporadic; this makes the system more vulnerable to high node mobility. A hybrid or dynamically-switched active-passive system could potentially solve this problem, but to date there has been no proposals of this type.

C. Cluster Head selection Criteria

The criteria used to rank potential cluster heads are usually chosen with the objective of mitigating the adverse effects of mobility on communications reliability and throughput, the performance of some particular application, or both.

Early VANET clustering schemes were extensions of MANET algorithms, such as LID/HD [18], DMAC [26], or MOBIC [20]. Of these protocols, the former two used some unique node identification number such as the MAC address or the degree of the node (i.e. number of neighbours within range) as a cluster head selection parameter, while the latter attempted to model the mobility of nodes by calculating the ratio of RSS measurements from consecutive broadcasts. Node ID is by far the least suitable metric for VANET clustering as it is entirely arbitrary and offers no way to account for mobility – it only guarantees that one node will have the lowest ID number, preventing a tie in the selection process. The VANET channel is known to be subject to highly variable fading behaviour [86]–[88], meaning that a single pair of consecutive RSS measurements, upon which MOBIC's cluster head selection mechanism is based, may give a quite unreliable estimate of the relative movement of nodes. The degree of connectivity provides a more useful selection metric, as it allows nodes to infer which potential cluster head will be able to serve the largest number of vehicles. However, *quantity* does not necessarily imply *quality*, especially when the network topology changes frequently.

VANET research is slowly moving toward innovative cluster head selection criteria, making clusters more robust to topology changes. Several recently-proposed strategies utilise node

TABLE IV: Algorithms using Timer as the Selection Strategy

Algorithm	Selection Metric(s)	Neighbour Discovery	Affiliation Handshake	Cluster Head Handoff
CCP	RSS	Hello	No	No
SBCA	RSS, Relative Velocity	Hello	No	Yes
CBMAC	Static Timeout; Multiple heads decide with Mean Distance and Velocity	Hello	Yes	No
HCA	Random Timeout	Hello	Yes	No
DBA-MAC	Link Expiration Time	Hello	Yes	No
CBLR	Static Timeout	Inquiry	Yes	No

TABLE V: Algorithms using Passive Clustering as the Selection Strategy

Algorithm	Selection Metric(s)	Neighbour Discovery	Affiliation Handshake	Cluster Head Handoff
PC	First Declaration Wins	None	No	No
VPC	Degree, Link Quality, Link Expiration Time	None	No	No
CF-IVC	Speed Grouping	None	No	No

mobility parameters, signal quality indicators, link expiration time, platoon leadership, vehicle class, trustworthiness, and driver intention.

1) Mobility and Proximity: Relative mobility between a node and its potential cluster heads is a logical choice as a cluster head selection parameter, since nodes which are moving together with small relative velocities are more likely to be able to maintain a stable communications link. Similarly, potential cluster heads which are in close proximity to a node are more likely to be able to support a reliable link, at least in the short term, compared to more distant nodes due to the inverse square law of electromagnetic signal propagation. Therefore, many clustering algorithms use one or both of these metrics to rank potential cluster heads.

Proposed in [33] as an extension of MOBIC, ALM modifies MOBIC's selection metric to utilise GPS location data distributed in *Hello* messages, rather than the limited-accuracy method of using consecutive RSS measurements.

RMAC [44] prefers potential cluster heads with more members, lower speed relative to the joining node, smaller distance to the joining node, and higher link expiration time. This strategy is quite successful in identifying cluster heads which are well-connected to a group of nodes moving in a coordinated fashion, resulting in large cluster sizes and long average affiliation times. The preference for nodes with good proximity to many similarly-behaving nodes is very important, since it increases the probability of successful packet reception.

Utility Function (UF) [36] uses a weighted sum of ID or degree of connectivity and either average relative velocity or average relative distance of neighbours. This algorithm performs better than plain LID and HD, providing longer lived and more stable clusters than either.

In PPC [35], the weight is calculated by hashing the system time and a node's ID, and XOR-ing the result with an eligibility function. The eligibility of a node is determined by travel time and the difference between the node's speed and the average speed of its neighbours. In this way, cars having a longer route and moving with the bulk of the vehicle group are regarded as better choices for cluster head. The hash function allows the assignment of unique weight values to prevent cluster head collisions.

Affinity Propagation in Vehicular Environments (APROVE) [58] is unique as it applies a statistical analysis method called *affinity propagation* to the clustering problem. Vehicles are grouped according to proximity and velocity relative to neighbours. The node that is most representative of its neighbours' location and behaviour is selected cluster head. The algorithm is multi-pass and must be executed periodically, with the outcome of previous clustering phases retained in order to give preference to current cluster heads and prevent unnecessary reclustering. However, the algorithm is also asynchronous – the received availability and responsibility messages will be at least one period old, therefore they correspond to the previous iteration. The age of these messages may be even greater if a channel is lossy.

Quantitative comparisons of the performance of MOBIC, ALM, PPC and APROVE with more recent and VANET-specific clustering algorithms are provided in [89]. Performance under lossy channels was evaluated as well, using a uniform error model with increasing channel error probabilities. The study showed that MOBIC, ALM, and PPC's clusters are shorter-lived with packet loss rates of greater than 10%, while APROVE's clusters lasted up to a minute with loss rates of up to 30%; reaffiliations were also much less frequent when compared against the competing approaches. APROVE's performance began to drop significantly beyond a loss rate of 40%.

A passive clustering algorithm called Cluster Formation for IVC (CF-IVC) is introduced in [47]. It uses a speed-based grouping approach to segment the network. Each vehicle knows the speed group to which it belongs, and affiliates with a CH in that grouping; this speed grouping then becomes part of the addressing scheme. A speed threshold is used to detect when a vehicle has changed speed groups and whether it should change clusters. A CDMA scheme assigns codes to cluster members to uniquely identify them and avoid packet collisions.

Threshold-Based Clustering (TBC) [37], [38] is specifically designed for highway mobility, in which the slowest vehicle amongst local neighbours initiates the cluster formation process. Neighbours within range of this node will affiliate with it if the relative velocity between them is less than a specified threshold. The cluster head selection process is similar to HD, with consideration of proximity to mean position and velocity – the cluster head need not necessarily be vehicle that initiated the clustering process. The result is clusters that tend to form in long trains.

A unique approach to cluster head selection is taken by Spring Clustering (Sp-Cl), which uses a selection metric inspired by Coulomb's Law [34]. Cars are considered as point charges which exert forces on each other in proportion to the inverse square of inter-vehicular distance. In place of Coulomb's constant is a function of the relative mobility between a given vehicular node and each of its neighbours; the charge of each node is determined by the sign of the number returned by this function. The net force exerted on a vehicular node by each of its neighbours is then calculated, and the node with the most positive net force is selected as cluster head. The metric intrinsically reduces the score of a node whose neighbours are moving away from it, giving a higher preference for potential cluster heads around which other vehicles tend to group.

The Neighbour Mobility-based Clustering Scheme (NMCS) algorithm is introduced in [42]. NMCS models mobility using a mobility-oriented variant of the node degree metric. Instead of simply counting the number of neighbours of a given node, the node degree is defined as the sum of the number of neighbours that have left a node's range and the number of new neighbours acquired since the last processing round. The sum is then divided by the total number of neighbours to normalise the degree of mobility in the vicinity of the node: therefore, nodes with lower values for this metric are located in a relatively stable environment, indicating that such nodes are good candidates for the cluster head role.

2) Signal Quality: In a simulation, the real-time location and velocity of vehicles is readily available. However, in reality, GPS suffers inaccuracies as a result of poor satellite reception, such as the well-known *urban canyon* effect. Several surveyed algorithms solve this problem by employing signal quality as a clustering metric. The quality of a communications link is principally determined by the signal to noise ratio (SNR) of the received signal. If the noise floor is assumed to be a particular level or can be otherwise estimated, the received signal strength (RSS) is a reasonable proxy for direct measurements of the SNR. An estimate of RSS is available upon the receipt of any packet at a given node, while noise levels may either be measured during gaps in transmission, inferred from bit error rate and signal transmission parameters, or assumed to be constant, as is usually necessary when using most common commercial off-the-shelf 802.11 transceivers. Due to the time-varying nature of the channel – fast and deep fading due to signal reflections from buildings and other vehicles – and transient interference from other localised sources of electromagnetic radiation, RSS or SNR measurements may vary significantly from packet to packet; therefore it is often necessary to perform some sort of low-pass filtering on SNR and RSS metrics to avoid undesirable thrashing of the network topology.

If relative velocity and/or position are known with sufficient accuracy, these can be combined with RSS or SNR measurements to compute an overall cluster head selection metric, with the relative weights adjusted as desired. Density Based Clustering (DBC) [29], [30] is one algorithm that takes this approach. It incorporates SNR measurements into a weighted sum of Euclidean distance and relative velocity in assessment of the link quality between nodes. This mitigates the aforementioned difficulties with signal quality measurements. SNR may also be used in initial assessment of cluster head suitability, such as in Cluster Configuration Protocol (CCP) [69], [70], which requires that a node accepts a cluster head advertisement message only if the RSS exceeds a threshold. This ensures cluster head affiliation will only occur with a sufficiently stable connection. Stability Based Clustering Algorithm (SBCA) [71] extends CCP, complementing the RSS metric with relative velocity to improve cluster head lifetime.

Channel metrics other than SNR or RSS may be used. K-hop clustering [32] uses propagation delay in the calculation of the cluster head selection weight. Changes in propagation delay indicate that a potential cluster head is either moving towards or away from a node. Distributed Multihop Clustering using Neighbourhood Follow (DMCNF) [25] uses a similar propagation delay metric in its own calculation.

VPC [46] discards the "first declaration wins" selection approach used by PC in favour of vehicular mobility metrics. The authors tested their approach using three different metrics at both low and high vehicle speeds: degree, link quality, and link expiration time. It was found that using degree as the selection metric allowed the highest packet delivery ratios when node density was low, however, using the link quality metric resulted in the best PDR at higher node densities. A similar pattern was observed at both low and high vehicle speeds. In all cases, application of the vehicular mobility metrics resulted in an improvement compared to the original PC algorithm, however, since the choice of metric which resulted in the maximum improvement changed depending on vehicle density, overall performance could be further improved by adding context-sensitivity, such that a different set of metrics selected depending on the current vehicular traffic density.

3) Link Expiration Time: Modified Distributed Mobility Aware Clustering (MDMAC) [28] extends the LID/HD ranking scheme used by DMAC [26] through the addition of an estimate of expected link lifespan or *link expiration time* between a cluster-head-seeking node and a potential cluster head. To be considered for cluster head status, a node's cluster head suitability score must be better than the current cluster head, and the link expiration time must exceed a specified threshold. This prevents fast-moving vehicles from taking over the cluster head role before disappearing and causing another reclustering event. It also prevents vehicles travelling in the opposite direction from momentarily taking over the role of cluster head. Link expiration time is estimated based on relative position and velocity of the node and the potential cluster head, and an assumed maximum communication range. Similarly, Clustering Protocol for Topology Discovery (CPTD) [74] uses an estimate of expected link lifespan to avoid excessive reclustering with temporarily local cluster heads.

In DBA-MAC [49], a vehicle waits a set time before declaring itself a *backbone member* (BM) and broadcasting a *Hello* message containing its mobility metrics. Vehicles overhearing this declaration before their own timers expire then compute the expected period for which they expect to be within range of the first-to-declare BM node. If this period exceeds a set threshold, the node calculates and waits for a randomly-generated contention window, at the end of which it transmits a candidature frame to the BM. The BM declares the first node to transmit a candidature frame to be the next BM node in the chain, and the process repeats until all nodes have joined a chain.

In MCTC [41], the expected connection lifetime between nodes i and j is calculated using a different method to the previously discussed protocols; the difference between the assumed communication range and the inter-node distance is divided by the difference in speed between nodes i and j. If node i is travelling at a greater absolute velocity than node j, then j is ignored in the calculation. This mechanism ensures that nodes are properly separated into groups based on node velocity. Nodes with the greatest link expiration time are selected cluster head.

4) Platoon Leadership: Another clustering metric uniquely available to VANETs is the position of the node along the road relative to its neighbours. Clustering based on Direction in Vehicular Environments (C-DRIVE) [52]–[54] selects the front-most car in a platoon as the cluster head, with cluster formation events triggered when traffic is stopped at intersections. Additionally, vehicles attempt to affiliate with cluster heads that intend to turn in the same direction as themselves (See Section IV-C6).

C-RACCA [73] checks whether cluster head announcements come from behind or in front, as well as whether the distance between it and the responding node exceeds a threshold calculated as a function of difference in velocity and deceleration capability. It then assesses whether it should join the responding cluster or assume the role of cluster head itself.

5) Vehicle Class: The concept of selecting a cluster head based on vehicle class is considered with Mobile Infrastructure in VANET (MI-VANET) [24]. In this architecture, private cars never act as cluster heads – this function is performed only by buses. Cars then affiliate with the nearest bus they can locate based on *Hello* received from buses. This algorithm is unique in that there is no distributed selection system; therefore, the architecture of MI-VANET shares some characteristics with a traditional infrastructure network, albeit one with mobile access points.

Sp-Cl is extended to differentiate nodes using a height-based metric in [23]. In the resultant Enhanced Spring Clustering (E-Sp-Cl) protocol, taller vehicles are assigned a higher charge value than shorter ones, ensuring that the tallest vehicle in the vicinity would be selected cluster head. The taller vehicle can then route packets between nodes whose Line-of-Sight path they may otherwise obstruct. Using a shadowing model to account for vehicular obstructions, the authors compared the performance of E-Sp-Cl to Sp-Cl, and showed that performance improved as a result of the increased network coverage resulting from its preference for taller cluster heads. This approach implies cluster head selection based on vehicle class, as buses and trucks are taller than cars.

6) Driver Intention: In [63], a selection metric which extends the approach used by UF is presented, called Lane-Sense Utility Function (LSUF). It aims to improve cluster lifespan by selecting a cluster head from a set of vehicles travelling in the same direction, which are expected to continue to take the same path at the next intersection. The overall cluster head selection score for a candidate cluster head is a function of relative position and velocity, number of connected neighbours, and the lane the vehicle is currently occupying. It is assumed that the lane determines whether the vehicle will turn left, right, or continue straight at the next intersection, and a weight is assigned to each traffic flow based on the number of lanes belonging to each flow. LSUF functions better than LID, HD, and UF in creating stable clusters on a highway scenario with multiple highway exits. LSUF is used to construct clusters in a TDMA Cluster-based MAC (TC-MAC) [64], [65].

User Oriented Fuzzy-logic-based Clustering (UOFC) [40] uses a fuzzy classifier to combine vehicular velocity and position information with a measure of driver intention, with the latter inferred by observing passenger interest in certain goods and services accessed via an in-car entertainment system. Vehicles which appear to have a common likely intended destination are then clustered together. The authors' simulation results indicate that clusters last significantly longer than with regular UF, and are more stable. This study also considers the number of lanes and vehicle speed in analysis, with results showing the emergence of fewer clusters with higher stability on roads with more lanes. While an interesting and unique approach to the problem of predicting vehicle behaviour, most passengers would now tend to use their own mobile devices, limiting the amount of information from which these inferences can be made. The extent to which passenger interaction with in-car entertainment systems accurately reflects driver intention is also highly debateable; for many journeys, passengers – children in particular – may be more interested in consuming media unrelated to their intended destination.

Fuzzy-Logic-Based Algorithm (FLBA) [39] uses relative velocity between vehicles as a cluster head selection metric. As in UOFC, a fuzzy-logic algorithm assesses driver intention in order to detect long-term changes in cluster head eligibility.

AMACAD [55] computes a selection metric as a weighted sum of inter-node distance, relative velocity, and Euclidean distance between vehicle destinations. This allows the algorithm to cluster vehicles which are taking similar routes.

Cellular Automata Clustering (CAC) [50] groups vehicles together according to the level of interest in nearby services, as indicated by the driver. This forms interest groups that can be used to route data relevant to those interests between the vehicles. Each interest has a channel allotted to it, with an additional channel set aside for emergency data. Unfortunately, little information is presented on how the cluster head is selected or how neighbourhood data is collected.

7) Security Reputation: Security and trust in VANETs is essential in order to prevent malicious agents sabotaging road safety systems built upon the VANET framework, potentially causing serious disruption to traffic flows or safety hazards. Several authors have proposed cluster head metrics which can assist in identifying malicious vehicles and mitigating their impact by denying them access to cluster resources.

VWCA [66] computes a cluster head selection metric based on vehicle direction, degree of connectivity, an entropy value calculated from the mobility of nodes in the network, and a *distrust level* based on the reliability of a node's packet relaying. Vehicles are assigned *verifiers*, which are neighbours with a lower distrust value. Verifiers monitor the network behaviour of a vehicle, and confirm whether it is routing packets and advertising mobility and traffic information that is consistent with the verifier's own view of the neighbourhood. The distrust value for nodes which behave abnormally is then automatically increased, while it is decreased for nodes which perform reliably. In this way, the trustworthiness of a node is accounted for by the cluster head selection process.

Trust-dependant Ant Colony Routing (TACR) [48] supplements position and velocity with Certificate Authority trust metrics to compute a weighted cluster head selection metric. These metrics include *traffic obedience* and *reputation* for correctly forwarding data packets. This has a dual purpose: it firstly ensures that the cluster head is capable of reliably functioning in this role, and secondly creates a layer of security around clusters, preventing malicious vehicles from joining a cluster and becoming cluster head.

Cluster-Based Public Key Infrastructure (CBPKI) [67] selects the most trustworthy vehicle in a neighbourhood as its cluster head. Affiliation is handled by a cluster head's 1-hop neighbours. This neighbourhood is called the VANET Dynamic Demilitarised Zone (VDDZ) and prevents malicious vehicles from accessing the cluster head directly. A cluster member can only directly communicate with the cluster head when its trust value exceeds a set threshold.

8) Other Criteria: PC's [45] "first declaration wins" approach is unique in that nodes with data to transmit are more likely to become cluster heads; PC's cluster head selection metric is therefore dependant on the behaviour of the application layer. Various factors can affect the behaviour of the application layer, such as driver intention and passenger interests. However, this does not necessarily imply the node will be able to perform the cluster head role adequately.

CCA [31] defines a metric denoted *network criticality* as "the normalised random walk betweeness of a node on the network" (sic). That is, how often a node is visited when traversing the network from a source to destination. Through experimental analysis, local network criticality is determined using the MP-inverse of the Laplacian Matrix of local network topology. The metric is computed as the average criticality between a node and its neighbours. Using this metric for cluster head selection ensures that a cluster head is chosen through which most nodes will naturally have to forward packets in order to deliver them to the intended network destination; essentially this approach identifies natural high-level packet relays in a dynamic hierarchical network infrastructure.

MCA-VANET [51] utilises an unusual metric for cluster head selection. Rather than vehicles joining the network in an initially unclustered state and then selecting a cluster head, all nodes start in the head role, and subsequently decide whether to step back to the role of a normal member, while selecting another node as its cluster head. The decision is based on an 8-bit counter maintained by each node for each of its neighbours. The counter is incremented for every *Hello* message received from the neighbour per beaconing period, and halved every time a period expires without reception of a message. As a result, nodes decide the eligibility of their neighbours to be cluster heads independently from other nodes, with mutual eligibility slowly increasing between cars that remain close for a sufficient period of time, then rapidly decreasing as beacons are missed, indicating that the nodes are no longer in proximity. This method is unique in that it does not need GPS data, which can be inaccurate in built-up areas, but instead estimates proximity from a connectivity standpoint. Additionally, nodes can have access to multiple cluster heads as a form of route redundancy. A node backs down to cluster member status if there are more than $numCH$ cluster head nodes with an eligibility level above a set threshold.

D. Cluster Member Affiliation Strategy

Once a node identifies its preferred cluster head, it will send a join request message to the prospective cluster head, unless it has decided to take on the role itself. When it receives the message, if the cluster head accepts the request, it will insert the requesting node's identification token, for example its MAC address, into the cluster table. Most algorithms surveyed will stop there, while others require a confirmation frame to be returned to the cluster member to inform it of success or failure. This *handshaking* is an important distinction between algorithms, and it results in a number of specific advantages and disadvantages, which are discussed below.

1) Cluster Control Capability: Using handshaking, a cluster head can exercise control on the topology and composition of the cluster by deciding whether or not to admit a prospective cluster member. In RMAC [44], a cluster head will deny access to a cluster if it has reached a predefined maximum size. This method would be particularly useful in channel access schemes such as CBMAC [43] and TC-MAC [64], where the number of channel slots limits the number of members that a cluster head can serve. In AMACAD [55], the handshake allows a cluster head time to calculate the new member's suitability to be a new head, and transfer leadership to the new node if necessary (see Section IV-G2).

Handshaking also allows the development of multi-state affiliation strategies. Nodes may enter an intermediate cluster membership state, allowing them to temporarily join clusters in order to gather more information about the environment before deciding whether or not to join. This approach is used in DBC [29], [30] and CCA [31]. Through this 'trial membership' period, the cluster head has the opportunity to acquire more information about a cluster member, such as obtaining statistics for routing reliability and trust metrics by observing its behaviour in the cluster over time. This approach is used in CBPKI [67].

2) Improved Integrity: The VANET channel has a high potential for packet loss due to vehicular obstructions and multipath fading. An affiliation message can be lost or dropped by the cluster head, and without confirmation requirement a cluster member will erroneously believe that its affiliation was successful. This is called a *faulty affiliation*, and is a serious problem for clustering algorithms intended for use in routing and security applications. An instance of faulty affiliation is shown in Figure 4. Beacon-based maintenance mechanisms (see Section IV-E1) isolate faulty nodes from the network, as the node may continue to hear a cluster head's signal for up to a minute, during which it will not restart the clustering process. If such an algorithm were used as a foundation for data dissemination or routing, a faulty affiliation will isolate potential members from the network and they will be unable to disseminate or receive data. Therefore, this method is not well-suited for real-time traffic collision warning applications in which a CH distributes collision warnings. Fault nodes will not receive the warning data, as the CH is unaware of them, and appropriate avoidance action will not be taken.

The simulation study in [90] evaluated the occurrence of faulty affiliations in MDMAC [28], which does not incorporate handshakes in its affiliation strategy, and compared its behaviour with that of RMAC [44], which uses handshaking. Figure 5 shows the incidence of faulty affiliation as a percentage of reaffiliation. Regardless of the cluster head selection metric used by MDMAC, almost half of reaffiliations resulted in faulty affiliations, whereas RMAC experienced none at all due to the usage of handshakes. The choice of cluster head selection metric and the particular road geometry, i.e. the number of lanes, makes some difference to the incidence of the malfunction, but it is clear that the addition of affiliation handshaking eliminates it completely.

3) Increased Overhead: The disadvantage of protocols with handshaking is the increased data overhead, which consumes some fraction of bandwidth. As shown from the results in the analysis of PC [45], clustering overhead can interfere with both routing control traffic and other data traffic, and the use of handshaking will cause even greater reduction in performance. A passive clustering approach in which handshakes are piggybacked onto existing traffic could potentially solve this problem, provided that a sufficient level of background traffic exists in the network.

E. Neighbourhood Discovery

There are two neighbourhood discovery mechanisms used in VANET cluster design. *Hello* messages are periodic beacons that are either broadcast by all nodes, as in MDMAC [28], or only by those that have declared themselves to be a cluster head, as with timer based approaches such as CBLR [83]. The alternative is *Inquiry* messages, which are polls of the local network topology – a node sends out an *Inquiry* frame, and waits for responses from nearby nodes. Thus neighbour discovery is either proactive, via *Hello* broadcasts, or reactive, via *Inquiry* requests.

1) Hello *Messages: Hello* messages are a simple method for local topology discovery, with the added benefit that missed beacons may be used to detect failed or failing connections to cluster members. However, if a multi-hop algorithm forwards *Hello* messages it receives – as with MDMAC [28] and VMaSC [77] – these broadcasts consume a non-trivial fraction of available bandwidth in high node density scenarios, while increasing collision rate due to increased medium contention. This reduces the reliability and speed of information dissemination, which may compromise vehicle safety applications. Furthermore, the use of *Hello* messages without handshaking for affiliation can result in isolated nodes suffering from faulty affiliation.

2) Inquiry *Messages:* These are broadcast by nodes as data solicitation messages. Nodes that receive an *Inquiry* message will respond with a unicast acknowledgement that contains their data. The transmission of *Inquiry* messages may only be needed when a node enters an unclustered state, as in RMAC [44]; or it may be periodic, as in DBC [29] and CBLR [82]. Algorithms such as RMAC, which only perform the *Inquiry* phase once, can then update their neighbour table opportunistically by receiving other *Inquiry* frames, as well as topology updates from their cluster head.

Inquiry frames allow neighbour information to be disseminated only on demand, without risking a broadcast storm. Response frames can incur a slightly larger overhead than *Hello*. Nodes will send out *Inquiry* frames while responding to other nodes. By contrast, *Hello* messages are advertisements of a node's own information, meaning nodes need only store the data from messages they receive without sending their own data in unicast form.

F. Gateway selection Metrics

Several of the algorithms discussed in this paper use clustering to assist with routing or integration between cellular networks and VANETs. These algorithms identify the cluster member which is best suited to act as a *gateway* to other clusters – which is not necessarily the same node as the cluster head. The specific selection criteria for each algorithm are listed in Table VI. The criteria tend to be application specific.

Routing algorithms employ cluster membership as a requirement for gateway status, while algorithms targeted at security-sensitive applications additionally require gateway nodes to have a sufficient trust level to guard against attack from malicious nodes.

G. Cluster Maintenance

After a cluster has been formed, the cluster head is responsible for coordinating communication between members while maintaining a stable cluster topology. As identified in [90], road network structure and vehicle density both have a strong influence on cluster stability. In order to adequately maintain the stability of a cluster, the cluster head must actively maintain a database of its members and respond in

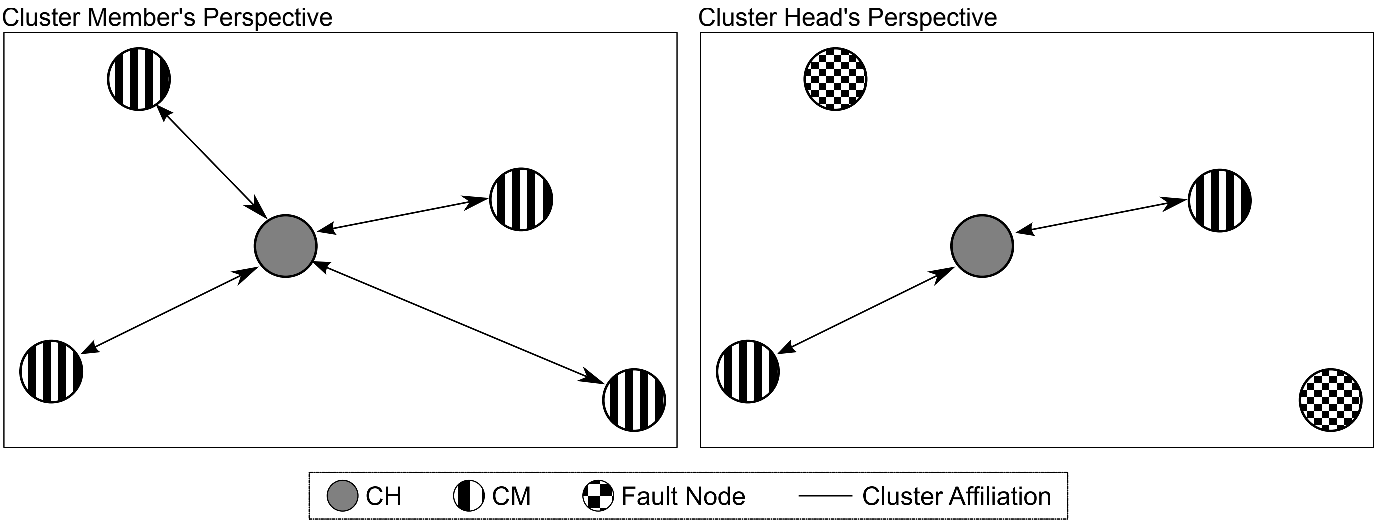

Fig. 4: An illustration of the *faulty affiliation* malfunction. Join messages are lost due to packet collisions or fading; these cannot be detected by an affiliating node, unless the join requires an explicit acknowledgement on the part of the cluster head. In some cases, faulty nodes become isolated from the network, as broadcast-based maintenance mechanisms prevent the faulty node from disconnecting and seeking out other clusters to join. This limits the use of algorithms such as MDMAC [28] in highly dynamic networks.

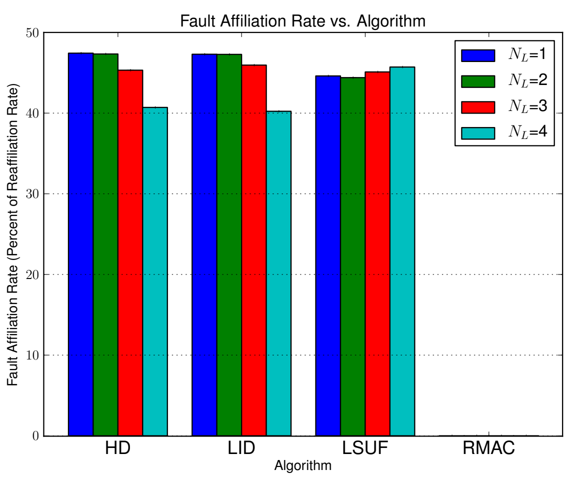

Fig. 5: Fault affiliation as a percentage of reaffiliation, from the simulation survey in [90]. MDMAC was simulated using three different cluster head selection metrics: Lowest ID, Highest Degree, and LSUF [63]. N_L stands for the number of lanes on the highway that comprised the simulation scenario.

a timely way to changes in membership. Two major database update mechanisms are described in the literature; these, along with potential causes of instability and proposed mitigation strategies are presented in the following sections.

1) Cluster Member Data Updates: Two main mechanisms for updating cluster membership information are described in the literature. The simplest approach is to utilise the *Hello* messages sent by all nodes for neighbour discovery, although this approach is subject to the disadvantages previously discussed in Section IV-E1. MDMAC [28] and AMACAD [55] employ the *Hello* broadcast in their cluster membership update mechanisms.

TABLE VI: Gateway selection metrics

Algorithm	Gateway selection Metric(s)
CBLR	Cluster Membership
CBMAC	Cluster Membership
CMGM	3G RSSI
CSBP	Randomised Timeout
CF-IVC	Speed Grouping, RSS w.r.t cluster head
MCTC	Relative Velocity w.r.t cluster head
CBPKI	Trust Level w.r.t both cluster heads
FQGWS	RSSI, LET, throughput, and network load

An alternative approach is to periodically poll the cluster. A cluster head sends out a broadcast, containing cluster-wide information for all cluster members, which can include mobility metrics for all nodes. Each member then responds with a unicast acknowledgement containing its own current state information, such that the cluster head's database can be updated for the next broadcast, if necessary. Thus members can obtain up-to-date information about their cluster head, and potentially about other nodes in the cluster which are not all necessarily within 1-hop range. Polling also provides another layer of protection against nearby candidate cluster heads that may temporarily have a higher cluster head selection metric. RMAC [44] and DBC [29] are examples of protocols with a poll-based data update mechanism.

Polling shares some traits with *Inquiry* mechanisms, including higher control overhead. It also makes a cluster vulnerable to transient channel dropouts due to fading, and packet collisions. If a cluster member does not respond to a poll, the cluster head assumes that cluster member has left;

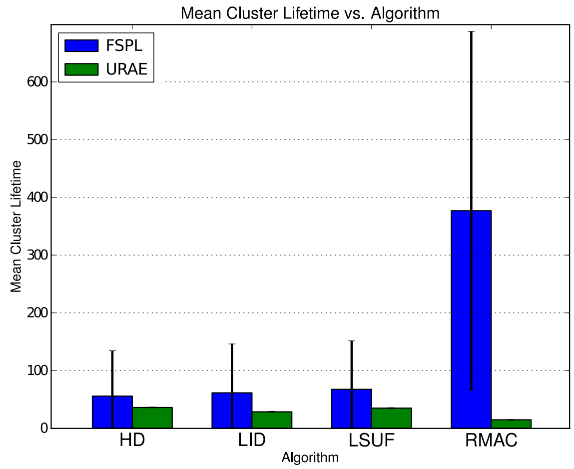

Fig. 6: Cluster lifetime of *Hello*-based MDMAC and polling-based RMAC under a simple free-space path loss channel mode, and a more sophisticated urban radio channel model, URAE [91]. This was taken from the simulation survey in [90] (See Figure 5). The disparity of RMAC's lifetime shows the vulnerability of polling to complex channel phenomena such as fading.

similarly, if a cluster member does not hear a poll request from its cluster head within the expected time period, it considers the connection to the cluster head to be dead. It is possible that both nodes are still within range of each other, and either a poll or the corresponding response was lost due to collisions or transmission errors. This results in an unnecessary reaffiliation attempt and decreases the stability of the cluster.

Figure 6 shows the performance of MDMAC and RMAC under two channel models: a simple free space path loss channel, and a more complex model for urban environments, the Urban Radio Channel [91]. RMAC exhibits a much greater performance disparity between the different channel models due to its more complex maintenance scheme, illustrating the vulnerability of polling mechanisms to vehicular shadowing and fading. While it obtains a much longer mean cluster lifespan for the free space channel model, the addition of fading and shadowing in the URAE model causes its performance to drop substantially compared to all flavours of MDMAC. Despite this, polling is a much more robust update mechanism compared to a *Hello*-based scheme and allows greater control over the clustering structure. Therefore, research should be carried out to supplement the polling approach with analysis of channel metrics such as SNR and BER, in order to better assess the quality of cluster member connections.

2) Cluster Head Depreciation: As the cluster members move relative to one another due to normal traffic behaviour, the cluster head selection metrics naturally vary. A cluster head eventually becomes a sub-optimal choice for the role, as it may be unable to communicate with all of its members, leading to the collapse of the cluster structure. In response to this, a cluster head can intentionally disband the cluster, enabling its members to find a new cluster head in a controlled manner. C-DRIVE [53] actively detects when a cluster is approaching an intersection and forces it to disband. This results in a reclustering event which may result in one of the ordinary members of the previous cluster becoming a new cluster head. Alternatively, a secondary or alternative cluster head, which can assume the role of cluster head when required, can be explicitly nominated by an existing cluster head. The secondary cluster head can be chosen on demand, in response to a decision by the current cluster head to relinquish its role, after which a handoff procedure is initiated and the secondary becomes the new primary. Alternatively, a standby secondary cluster head is selected immediately after the selection of the primary cluster head, potentially changing as decided by the cluster head based on its knowledge of the cluster membership and topology. Five of the surveyed algorithms employ a secondary cluster head handoff mechanism.

AMACAD [55] achieves this by evaluating new cluster member affiliations, determining the cluster head selection metric for the new node with respect to the members of the current cluster. The cluster head decides to relinquish its role if the newly affiliated cluster member has a better selection score. FLBA [39] uses the second-best cluster head candidate as a backup cluster head. C-DRIVE [52]–[54] passes the cluster head role to overtaking vehicles.

C-RACCA [73] decides whether a node should take over as cluster head based on the inter-vehicular distance between them. The distance between the cluster head and the new cluster member must be such that, in the event of an accident, there is sufficient time for the following vehicle to brake and avoid collision with the vehicle ahead. If there is not, the cluster head role is transferred to the new member.

SBCA [71] selects secondary cluster heads based on a mobility calculation similar to its cluster head selection metric, except that it is performed only with respect to the current cluster head. The implication is that a node nearest to the current cluster head would be a good candidate to take the role. However, this does not account for the possibility that the current cluster head has already become a sub-optimal choice, and thus a node with similar mobility patterns would also be equally inappropriate.

DMCNF [25] employs a novel affiliation scheme in which a node "follows" a one-hop neighbour. If the neighbour is a CH, this behaviour is considered to be a normal direct affiliation; if it is a cluster member, the node is said to have affiliated *indirectly* with the CH. This method can then be used for dynamically reconfiguring the clustering structure, since a normal cluster member may potentially assume the CH role if it has accrued more "followers" than its current CH and it has a lower average relative velocity with respect to those followers.

3) Merging Clusters: Due to node and cluster mobility, it is likely that at some point two cluster heads will pass within range of each other. Most clustering algorithms try to form a small number of large clusters; hence when two clusters approach one another, they may merge to form a single large cluster. However, for the case of two clusters with different group velocities – as is the case when one group of vehicles overtakes another group on a highway – the merger will only be temporary as the faster-moving nodes will subsequently

TABLE VII: List of simulators and the publications in which they were first described.

Simulator	First described in
ns-2	[92]
Unspecified	N/A
Custom Testbed	N/A
JiST/SWANS++	[93]
ns-3	[94]
MATLAB	[95]
OmNeT++/VEINS	[96] [97]
GloMoSim	[98]
Traffic Simulation 3.0	[99]
VANETMobiSim	[100]
NCTUns	[101]
SIDE/SMURPH	[102]
OPNET	[103]

move out of range. This will result in a partial cluster collapse, with both groups of nodes potentially needing to recluster.

To avoid this problem, some algorithms take measures to prevent unnecessary cluster absorption. GDMAC [27] employs two additional checks in its selection mechanism, such that members will not reaffiliate unless the new candidate's cluster head selection metric exceeds that of the current cluster head by a set threshold. ALM [33] avoids reclustering when dealing with nearby cluster heads by means of a contention time-out, requiring that the heads be within range for a certain minimum period of time before merging. MDMAC [28] uses link expiration time to excise high-speed cluster head candidates, which also avoids the merging of clusters with significantly different group velocities.

Hierarchical clustering algorithms actively try to merge adjacent clusters as an intended part of their design, with the smaller cluster becoming a sub-cluster of the other. As a result, a backbone or hierarchy is formed, which can then be used for routing and data dissemination protocols. Such algorithms extend the network coverage at the expense of increased overhead, and must avoid the formation cyclical structures and backbones consisting only of cluster heads. RMAC [44], DBA-MAC [49], and CBLR [82], [83] employ this approach.

V. Verification Methodology

Rigorous experimental evaluation of proposed VANET protocols and applications in urban environments requires the deployment of hundreds or thousands of wireless vehicular nodes. Even if a research program is able to recruit enough volunteer drivers, the cost and time expenditures become highly prohibitive beyond a few tens of nodes, severely limiting the value of experimental results. For this reason, VANET research is heavily reliant on software-based simulations for protocol and application development and evaluation. The principal advantage of such methods is that a wide range of proposed protocols can be evaluated under identical or

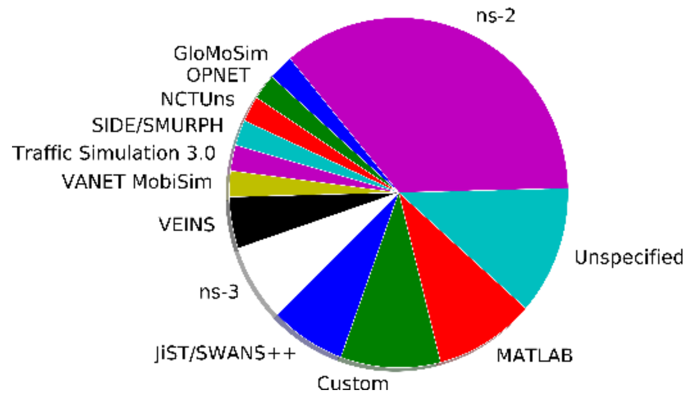

Fig. 7: Simulators utilised in validation of the surveyed algorithms.

equivalent channel and vehicular mobility conditions, with arbitrary vehicle distributions, vehicle behaviour and network traffic loads. As the simulation is not constrained to run in real time, the only limiting factors are the execution time, processing and storage/memory requirements. Despite these limits, very large simulations can be run in a reasonable period of time with a modest amount of commodity computing hardware, with the additional benefit of being able to quickly and easily modify the simulated environment.

Both the choice of channel model and the choice of simulation framework itself have a significant impact on the accuracy of the results of simulations. Several alternatives for each of these parameters are discussed in the following sections.

A. Channel Models

The method by which the wireless signal propagation environment is modelled has a significant impact on the performance of the protocol or application under evaluation [90]. Numerous authors – for example [87], [104], [105] – have previously demonstrated the need for accurate modelling of channel dynamics for meaningful evaluation of the performance of VANET protocols and applications. A model that accurately accounts for many known physical channel properties – such as path loss, fading and shadowing – will provide a much better indication of the likely practically achievable performance of a proposed protocol or application; however, despite the known sensitivity of VANETs to channel properties, much of the literature assumes a simplistic channel model such as free-space or two-ray path loss, or simple Rayleigh or Rician fading channels. These models do not provide a realistic representation of the characteristics of signal propagation, particularly in urban environments. This can result in misleading relative performance estimates between protocols under test, and makes direct comparison of results from different publications almost impossible. The use of simplistic channel models persists despite on-going research and development of more realistic and sophisticated urban channel models [86], [104], [105].

We have identified seven distinct wireless channel models currently in use in the VANET literature, and thirteen simulators including very general-purpose platforms such as Matlab.

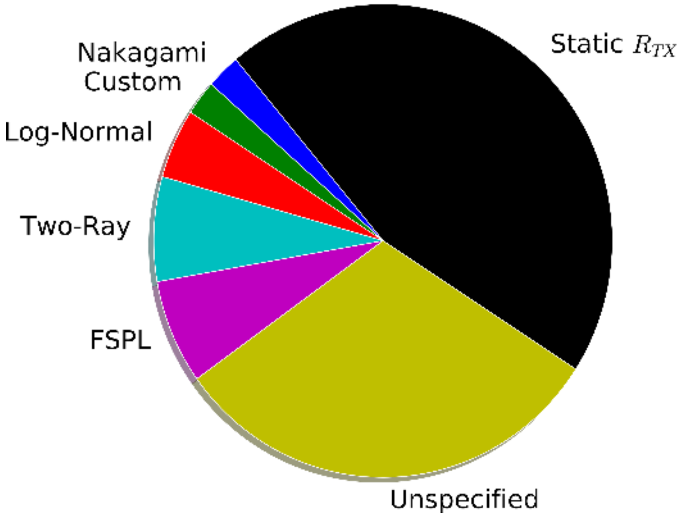

Fig. 8: Channel models utilised in validation of the surveyed algorithms. The two most popular models are *Static* R_{TX} and *unspecified*, which makes direct comparison beween algorithms based on performance reported in the literature difficult.

Figures 7 and 8 show the distribution of channel models and simulators commonly used in the validation process. The channel models are presented below, listed in order of increasing popularity, as measured by the number of published clustering protocols using that model for performance evaluation.

1) *Custom Models*: The authors have developed their own channel model, giving a brief description of its principles.
2) *Friis Free-Space Path Loss (FSPL)*: The simplest form of deterministic channel model, with only a direct path between source and destination. Some published algorithms are evaluated using the Friis channel model with statistical extensions to represent fading and/or shadowing; and
3) *Nakagami-M*: A well-known fading model for communication in urban environments, which assumes one transceiver is at a higher elevation than the others – this model is best suited to cellular networks;
4) *Log-Normal*: A shadowing/fading model based on the log-normal distribution;
5) *Two-Ray*: A pathloss model more sophisticated than FSPL, modelling the direct path and a single reflection from the ground;
6) *Unspecified*: The authors have not given any details on the channel model;
7) *Static* R_{TX}: A maximum transmission range has been specified such that nodes separated by more than this distance are unable to communicate;

The simulators are listed in Table VII. They have been arranged in order of popularity with respect to the papers surveyed in Sections III and IV.

B. Simulators and Channel Models

1) Mainstream Models: The most popular simulator amongst those used in the surveyed papers is ns-2 [92]. It regards a packet as having been successfully received if the received power exceeds a specified threshold while assuming a constant noise floor, based on the chosen channel model. The current stable release of ns-2 is version 2.35, which includes support for five well-known channel models: Friis free-space path loss, two-ray ground, Nakagami-M, log-normal shadowing and obstacle-based shadowing. ns-3, the successor to ns-2,also provides the same set of channel models. Obstacle-based shadowing is the most interesting model for VANET research; it uses a bitmap of the simulation environment to determine whether there are static objects occluding the direct line-of-sight path between the source and destination nodes. One of two alternative configurations of a log-normal shadowing model are then used depending on whether or not such a direct path exists. This model follows a similar approach to the CORNER channel model [104], [106], which also attempts to model the channel based on the specific physical environment in the vicinity of a given pair of nodes. However, in all the cited works using ns-2, none use these more advanced models, choosing to rely only on free-space, two-ray, or Nakagami-M channel models.

OMNeT++ itself is not specifically a network simulator but rather a general purpose simulation framework; for network simulation it must be combined with appropriate simulation models, of which a number are available [96]. The most comprehensive simulation model for wireless networks is MiXiM, [107] which offers free space path loss, two-ray, log-normal shadowing, Rayleigh fading, and simple obstacle shadowing models. Dror et al. [61], [62] use a simulation platform called VEINS [97], which combines the OMNeT++/MiXiM wireless network simulator with the SUMO vehicular traffic simulator [108] via the TraCI communication protocol [109]. The authors incorporate topological data into their channel model, using different models for different programmed scenarios as appropriate. A similar approach is applied in our previous work [90].

Version 6.0 of the NCTUns network simulator introduces significant capabilities for VANET research, in particular, modelling vehicular mobility and vehicular application interfaces [101], [110]. NCTUns offers three channel models: a simple static transmission range channel model, two-ray ground, and a Rayleigh fading model. NCTUns has been used with a static transmission range model in two of the reviewed papers [53], [54].

GloMoSim and its descendant simulators QualNet and Exata [98] provide a similar suite of channel models to NCTUns and ns-2, including free-space, two-ray and several terrain-based path loss models with log-normal shadowing and Rayleigh and Rician fading. The use of a static communication range in [45] implies the use of one of the deterministic path-loss models.

2) Custom Channel Models: The authors of [60] use a custom simulator described in [111]. The channel model is based on a static transmission range model with consideration of the effects of reflection and diffraction around buildings. This appears to resemble the approach taken in CORNER [104], [106]; however, few specific details on the model parameters are provided in the paper as the model is not the focus of the authors' work.

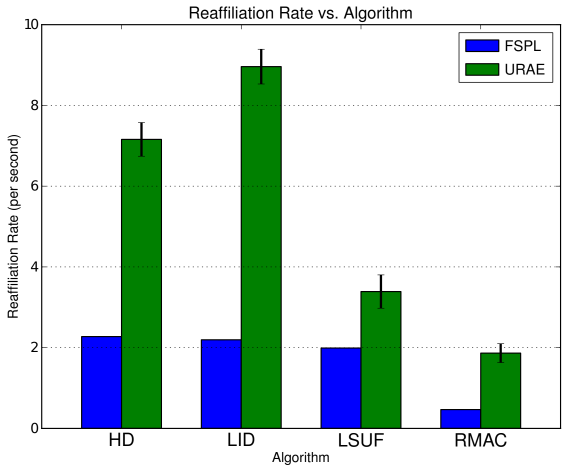

Fig. 9: Reaffiliation rate of surveyed algorithms under free space path loss and the experimentally-derived URC channel model. The simplistic model gives an overly optimistic view of performance, and hides disparities between relative performance of clustering methods which are seen when a more realistic model is used.

3) Unspecified Channel Model Citations: It must be noted that in many of the publications reviewed, no specific information on the choice of channel model is provided; in fact, the second-most common "channel model" is *unspecified*. It is therefore difficult to determine the generality of the simulation results presented in these papers. However, of the reviewed papers lacking information on channel model configuration, ten identify the simulation framework in use, which provides some insight into which models which may have been used.

4) Consequences of Simplified vs. Realistic VANET Channel Modelling: Our previously published simulation survey included a comparison of clustering performance under both a simple channel model – free space path loss – and a realistic urban vehicular channel model, URC, which was experimentally derived and incorporates fading and vehicular shadowing [90]. Figure 9 demonstrates how a simplistic channel model can give unduly optimistic indications of clustering performance and performance variability, both between clustering methods and between runs with the same method.

C. Comparative Studies

Most authors proposing novel clustering algorithms seek to illustrate performance advantages of the proposed algorithm using simulations in which a range of metrics are compared against one or more well-known alternative clustering algorithms. Often the choice of algorithm against which performance is to be compared is strongly influenced by the free availability of source code for simulations, or the existence of sufficient detail in the corresponding publication(s) to allow for straightforward implementation in a particular simulation environment.

Most of the papers cited in this survey present this type of performance comparison. Among those that don't offer any performance comparison at all – indicated in Table VIII as comparing to *None* – several algorithms aimed at very specific applications were compared to non-cluster-based alternatives for those particular applications instead, e.g. a cluster-based routing algorithm compared to a traditional MANET routing protocol.

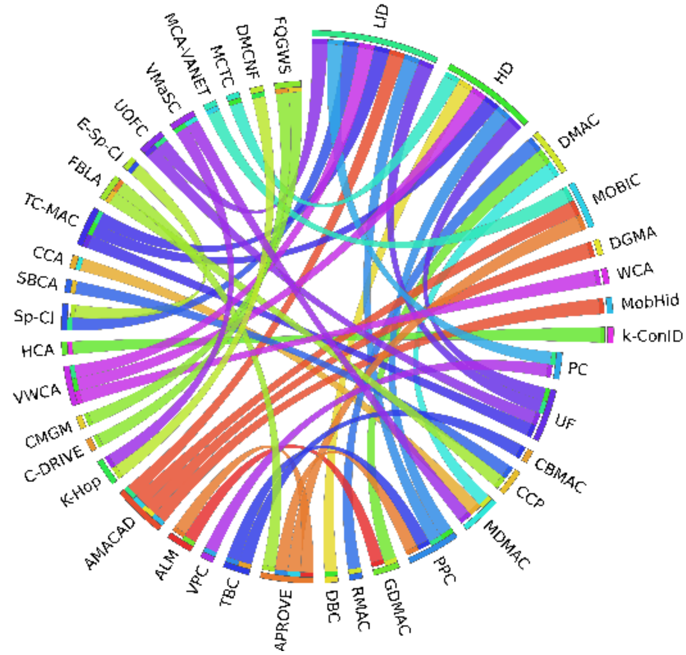

Fig. 10: Cluster comparisons. Algorithms are sorted clockwise in order of the date of publication. The colour of the ribbon matches the colour of the algorithm that is being proposed, and the ribbon leads to the algorithm to which it was compared. MANET algorithms jut out from the edge of the circle. Diagram created with Circos [112].

Figure 10 illustrates the relationships between the algorithms evaluated in this survey. The algorithms are ordered clockwise in order of the date of publication. The most obvious feature of this comparison is that many of the surveyed clustering methods are compared to MANET (rather than other VANET) approaches to equivalent problems – for example, comparing clustering VANET routing protocols with well-known non-cluster-based MANET routing protocols. When VANET technology emerged as a research discipline in its own right, it made sense to use these protocols for comparison; however, MANET clustering algorithms are often unsuitable for VANET scenarios – particularly urban scenarios – due to the constrained node-to-node connectivity and high node velocities. Regardless, despite the emergence of new clustering approaches specifically designed to address the challenges of clustering in VANETs, performance comparisons are frequently still made with MANET algorithms.

VI. Discussion and Conclusion

Several previous surveys and taxonomies of clustering in VANETs, such as [1], [2], have identified the need to exploit more of the unique aspects of the VANET environment to optimally solve the problem of cluster formation and maintenance in VANETs. The suggested directions include wider application of machine learning techniques, use of bidirectional vehicular traffic flows in the cluster formation process, and increased involvement from stationary road-side units. While these opportunities are significant, meaningful progress in advancing the state of the art in VANET clustering technique is hampered by a number of significant and fundamental shortcomings in the existing literature, which are summarised below, and which need to be adequately addressed if robust and reliable VANETs are to move beyond simulations and into large scale practical deployment.

A. The need for more VANET-specificity in VANET clustering algorithms

As shown in Tables II, III, IV, and V, many VANET clustering algorithms take a very similar approach to the clustering problem, and may differ only in their cluster head election metrics. Gateway election metrics and cluster head hand-off strategies are also identical in some cases, often due to the MANET ancestry of a given clustering method. Newer strategies, which fully exploit the unique mobility patterns, channel behaviour, energy capacity, and available processing power in VANETs, could potentially offer performance far beyond that achievable with more conservative approaches, particularly under realistic urban VANET channel conditions.

Better use of vehicular behaviour prediction, using road structure data, and road-side unit assistance, also seems likely to help with clustering. There are clear opportunities to utilise a wider range of metrics to assist with routing, particularly those which exploit the unique characteristics of the VANET environment. Robustness and stability of clusters could be improved with something as simple as modulating the weights applied to cluster head election metrics based on the number of lanes on the road on which the vehicle is currently situated, previously demonstrated in [90]. Instantaneous driver intention, as indicated by lane changes detected via indicator signals or accelerometers, signals important information which is relevant to cluster head eligibility, such as indicating that a vehicle is about to leave a highway. Further studies into the relation between such manoeuvres and cluster head suitability are warranted.

B. Channel Modelling Disconnect

There is an apparent disconnect between new findings in VANET channel research with respect to environmental dependence and non-stationarity of channel parameters, and the validation methodology used in the vast majority of clustering research, which tend to make very optimistic and unrealistic assumptions about signal propagation between nodes, or which fail to even specify how the channel is modelled. Simulations have demonstrated the critical link between channel model and clustering performance – ignoring this link significantly reduces the real-world relevance of simulation studies into clustering performance. For clustering techniques to be predictable and reliable in practice, it is essential that researchers move to channel models which properly reflect the reality of the complex signal propagation environment in which they must operate.

Accurate channel modelling, such as the model developed in [90], can reveal problems with certain design choices and enable researchers to improve the robustness of proposed algorithms. This will drive the development of innovative approaches for VANET protocol design and allow researchers to investigate new applications of the technology with greater confidence in the validity of their results.

C. Benchmarking and Validation

Any algorithm designed for VANETs should be expected out-perform a MANET algorithm in a vehicular scenario. While it is reasonable to use older but well-understood approaches like LID/HD and MOBIC as a basis for performance comparisons, one can only make this statement if the simulation scenarios are identical between publications, or if the older approaches are re-simulated under the same road and network conditions as the proposed VANET clustering scheme. There is also considerable variability in the choice and even nomenclature of performance metrics between publications, even for algorithms designed for the same application.

VANET research would benefit significantly from a standardised validation methodology, including a universally accepted set of performance metrics. Particular care should be taken to provide adequate details of the channel model, including implementation details and specific operational parameters, to allow independent replication and validation of published results. Additionally, source code availability would enable fellow researchers to quickly benchmark new proposals against previously published algorithms, and allow more robust comparison between protocols. It is noteworthy that many of the more well-cited authors in the field have provided open source implementations of their algorithms, demonstrating the mutual benefits of this practice.

D. A Recommended Approach to Future VANET Research

Mobile ad hoc networks in general and VANETs in particular require participants to harmonise their communication and aim to ensure fair distribution of channel access. Unlike the wider Internet, which is a general purpose carriage service that generally assumes relatively stable network topology, a VANET is a constantly changing dynamic network, with many of the applications served often having differing or even conflicting service requirements. Indeed, Kwon et al. noted that a general-purpose clustering algorithm with its own control frames may interfere with the very objective it was intended to achieve [45]. In the literature, much emphasis is placed on sending data from source to destination without concern for the *objective* of the network. In this light, the vision of a VANET that forms topology-homeostatic clusters, routes data reliably and securely, prevents collisions safely, delivers traffic data in time for the driver to respond, and executes comfort and infotainment applications becomes difficult to realise.

For most researchers, VANETs, unlike the Internet, have highly specific applications. Thus, protocol design methodology should begin with the intended application. From there, the known and accepted obstacles to vehicular communication should be considered and analysed. The protocol can then be designed around these obstacles; alternatively, the obstacles can be innovatively used to the protocol's advantage. An example of such advantageous use would be employing the broadcast nature of wireless to send data to multiple receivers with a single transmission as is used in the passive clustering approaches, or utilisation of coding, frequency, spatial, or cooperative diversity. Finally, the protocol should then be evaluated with an accurate channel model that accounts for as many of the known propagation phenomena as possible, with comparison to the most recent competing protocols. An improvement in methodology and analysis of the VANET problem will accelerate the roll-out of this technology and greatly increase the benefits it is hoped to provide.

REFERENCES

[1] S. Vodopivec, J. Bester, and A. Kos, "A survey on clustering algorithms for vehicular ad-hoc networks," in *Telecommunications and Signal Processing (TSP), 2012 35th International Conference on*, July 2012, pp. 52–56.

[2] R. S. Bali, N. Kumar, and J. J. Rodrigues, "Clustering in vehicular ad hoc networks: Taxonomy, challenges and solutions," *Vehicular Communications*, vol. 1, no. 3, pp. 134 – 152, 2014.

[3] Y. L. Morgan, "Notes on DSRC & WAVE standards suite: Its architecture, design, and characteristics," *IEEE Commun. Surv. Tutorials*, vol. 12, no. 4, pp. 504–518, 2010. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5462975

[4] "IEEE Standard for Information technology- 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 Environments," *IEEE Std 802.11p-2010 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, IEEE Std 802.11n-2009, and IEEE Std 802.11w-2009)*, pp. 1–51, Jul. 2010.

[5] I. C. S. E. T. Committee, "Vehicular networks and telematics applications sub-committee," 2014. [Online]. Available: http://www.danielwireless.com/vnta/

[6] I. S. Organization, "ISO/TC 204 Intelligent transport systems," 1992. [Online]. Available: http://www.iso.org/iso/iso_technical_committee?commid=54706

[7] E. T. S. Institute, "Intelligent transport systems," 2016. [Online]. Available: http://www.etsi.org/technologies-clusters/technologies/intelligent-transport

[8] I. V. T. Society, "Intelligent transportation systems committee (vt/its)," 2016. [Online]. Available: https://standards.ieee.org/develop/wg/1609_WG.html

[9] O. Eriksson and K. Axelsson, "Its systems architectures-from vision to reality," in *Proceedings of the 7th World Congress on Intelligent Systems*, 2000, pp. 1–8.

[10] "Cooperative vehicle-infrastructure systems," Accessed on 13-05-2016. [Online]. Available: http://www.transport-research.info/project/cooperative-vehicle-infrastructure-systems

[11] G. Toulminet, J. Boussuge, and C. Laurgeau, "Comparative synthesis of the 3 main european projects dealing with cooperative systems (cvis, safespot and coopers) and description of coopers demonstration site 4," in *2008 11th International IEEE Conference on Intelligent Transportation Systems*, Oct 2008, pp. 809–814.

[12] "Safespot – smart vehicles on smart roads," Accessed on 14-05-2016. [Online]. Available: http://www.safespot-eu.org/

[13] "NSF Project: Analytic Modelling and Enhancement of Vehicular Ad Hoc Networks for Safety Critical Applications," Accessed on 13-05-2016. [Online]. Available: http://vanet.oru.edu/index.html

[14] M. Alsabaan, W. Alasmary, A. Albasir, and K. Naik, "Vehicular networks for a greener environment: A survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1372–1388, Third 2013.

[15] A. Madrigal, "The trick that makes google's self-driving cars work," *The Atlantic*, May 2014. [Online]. Available: http://www.theatlantic.com/technology/archive/2014/05/all-the-world-a-track-the-trick-that-makes-googles-self-driving-cars-work/370871/

[16] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil, "Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions," *IEEE Communications Surveys Tutorials*, vol. 13, no. 4, pp. 584–616, Fourth 2011.

[17] S. Djahel, R. Doolan, G. M. Muntean, and J. Murphy, "A communications-oriented perspective on traffic management systems for smart cities: Challenges and innovative approaches," *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 125–151, Firstquarter 2015.

[18] M. Gerla and J. Tzu-Chieh Tsai, "Multicluster, mobile, multimedia radio network," *Wireless Networks*, vol. 1, no. 3, pp. 255–265, 1995.

[19] A. Ephremides, J. Wieselthier, and D. Baker, "A design concept for reliable mobile radio networks with frequency hopping signaling," *Proceedings of the IEEE*, vol. 75, p. 5673, 1987.

[20] P. Basu, N. Khan, and T. D. C. Little, "A mobility based metric for clustering in mobile ad hoc networks," in *Distributed Computing Systems Workshop, 2001 International Conference on*, 2001, pp. 413–418.

[21] Y. Zhang and J. M. Ng, "A distributed group mobility adaptive clustering algorithm for mobile ad hoc networks," in *Communications, 2008. ICC '08. IEEE International Conference on*, May 2008, pp. 3161–3165.

[22] C. Konstantopoulos, D. Gavalas, and G. Pantziou, "Clustering in mobile ad hoc networks through neighborhood stability-based mobility prediction," *Computer Networks*, vol. 52, no. 9, pp. 1797 – 1824, 2008.

[23] L. Maglaras and D. Katsaros, "Enhanced spring clustering in vanets with obstruction considerations," in *Vehicular Technology Conference (VTC Spring), 2013 IEEE 77th*, June 2013, pp. 1–6.

[24] J. Luo, X. Gu, T. Zhao, and W. Yan, "Mi-vanet: A new mobile infrastructure based vanet architecture for urban environment," in *Vehicular Technology Conference Fall (VTC 2010-Fall), 2010 IEEE 72nd*, Sept 2010, pp. 1–5.

[25] Y. Chen, M. Fang, S. Shi, W. Guo, and X. Zheng, "Distributed multi-hop clustering algorithm for vanets based on neighborhood follow," *EURASIP Journal on Wireless Communications and Networking*, vol. 1, pp. 1–12, April 2015. [Online]. Available: http://dx.doi.org/10.1186/s13638-015-0327-0

[26] S. Basagni, "Distributed clustering for ad hoc networks," in *Proceedings of the 1999 International Symposium on Parallel Architectures, Algorithms and Networks*, ser. ISPAN '99. Washington, DC, USA: IEEE Computer Society, 1999, pp. 310–.

[27] R. Ghosh and S. Basagni, "Mitigating the impact of node mobility on ad hoc clustering," *Wirel. Commun. Mob. Comput.*, vol. 8, no. 3, pp. 295–308, Mar. 2008.

[28] G. Wolny, "Modified dmac clustering algorithm for vanets," in *Systems and Networks Communications, 2008. ICSNC '08. 3rd International Conference on*, 2008, pp. 268–273.

[29] S. Kuklinski and G. Wolny, "Density based clustering algorithm for VANETs." IEEE, 2009, pp. 1–6.

[30] S. Kuklinski and G. Wolny, "Density based clustering algorithm for vehicular ad-hoc networks," vol. 4, pp. 149–157, 2009.

[31] W. Li, A. Tizghadam, and A. Leon-Garcia, "Robust clustering for connected vehicles using local network criticality," in *Communications (ICC), 2012 IEEE International Conference on*, June 2012, pp. 7157–7161.

[32] Z. Zhang, A. Boukerche, and R. Pazzi, "A novel multi-hop clustering scheme for vehicular ad-hoc networks," in *Proceedings of the 9th ACM International Symposium on Mobility Management and Wireless Access*, ser. MobiWac '11. New York, NY, USA: ACM, 2011, pp. 19–26.

[33] E. Souza, I. Nikolaidis, and P. Gburzynski, "A new aggregate local mobility (alm) clustering algorithm for vanets," in *Communications (ICC), 2010 IEEE International Conference on*, May 2010, pp. 1–5.

[34] L. Maglaras and D. Katsaros, "Distributed clustering in vehicular networks," in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2012 IEEE 8th International Conference on*, Oct 2012, pp. 593–599.

[35] Z. Wang, L. Liu, M. Zhou, and N. Ansari, "A position-based clustering technique for ad hoc intervehicle communication," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 38, no. 2, pp. 201–208, March 2008.

[36] P. Fan, J. Haran, J. Dillenburg, and P. Nelson, "Cluster-based framework in vehicular ad-hoc networks," in *Ad-Hoc, Mobile, and Wireless*

Networks, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2005, vol. 3738, pp. 32–42.

[37] Z. Rawshdeh and S. Mahmud, "Toward strongley connected clustering structure in vehicular ad hoc networks," in *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*, Sept 2009, pp. 1–5.

[38] Z. Y. Rawashdeh and S. Mahmud, "A novel algorithm to form stable clusters in vehicular ad hoc networks on highways," *EURASIP J. Wireless Comm. and Networking*, vol. 1, pp. 1–13, Jan 2012.

[39] K. Hafeez, L. Zhao, Z. Liao, and B.-W. Ma, "A fuzzy-logic-based cluster head selection algorithm in vanets," in *Communications (ICC), 2012 IEEE International Conference on*, June 2012, pp. 203–207.

[40] I. Tal and G.-M. Muntean, "User-oriented fuzzy logic-based clustering scheme for vehicular ad-hoc networks," in *Vehicular Technology Conference (VTC Spring), 2013 IEEE 77th*, June 2013, pp. 1–5.

[41] F. Chiti, R. Fantacci, and G. Rigazzi, "A mobility driven joint clustering and relay selection for ieee 802.11p/wave vehicular networks," in *IEEE International Conference on Communications (ICC) 2014*, 2014.

[42] J.-H. Kwon, C. Kwon, and E.-J. Kim, "Neighbor mobility-based clustering scheme for vehicular ad hoc networks," in *Platform Technology and Service (PlatCon), 2015 International Conference on*, Jan 2015, pp. 31–32.

[43] Y. Gunter, B. Wiegel, and H. P. Grossmann, "Medium access concept for VANETs based on clustering," in *2007 IEEE 66th Vehicular Technology Conference*, Baltimore, MD, USA, Sep. 2007, pp. 2189–2193.

[44] R. Goonewardene, F. Ali, and E. Stipidis, "Robust mobility adaptive clustering scheme with support for geographic routing for vehicular ad hoc networks," *IET Intell. Transp. Syst.*, vol. 3, no. 2, p. 148, 2009.

[45] T. Kwon, M. Gerla, V. Varma, M. Barton, and T. Hsing, "Efficient flooding with passive clustering-an overhead-free selective forward mechanism for ad hoc/sensor networks," *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1210–1220, Aug 2003.

[46] S.-S. Wang and Y.-S. Lin, "Performance evaluation of passive clustering based techniques for inter-vehicle communications," in *Wireless and Optical Communications Conference (WOCC), 2010 19th Annual*, May 2010, pp. 1–5.

[47] Ö. Kayiş and T. Acarman, "Clustering formation for inter-vehicle communication," in *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, Sept 2007, pp. 636–641.

[48] R. Sahoo, R. Panda, D. Behera, and M. Naskar, "A trust based clustering with ant colony routing in vanet," in *Computing Communication Networking Technologies (ICCCNT), 2012 Third International Conference on*, July 2012, pp. 1–8.

[49] L. Bononi and M. di Felice, "A cross layered MAC and clustering scheme for efficient broadcast in VANETs," in *2007 IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems*, Pisa, Italy, Oct. 2007, pp. 1–8.

[50] S.-T. Cheng, G.-J. Horng, and C.-L. Chou, "Using cellular automata to form car society in vehicular ad hoc networks," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, no. 4, pp. 1374–1384, Dec 2011.

[51] S. Vodopivec, J. Beter, and A. Kos, "A multihoming clustering algorithm for vehicular ad hoc networks," *International Journal of Distributed Sensor Networks*, vol. 2014, p. 8, 2014.

[52] N. Maslekar, M. Boussedjra, J. Mouzna, and L. Houda, "Direction based clustering algorithm for data dissemination in vehicular networks," in *Vehicular Networking Conference (VNC), 2009 IEEE*, Oct 2009, pp. 1–6.

[53] N. Maslekar, M. Boussedjra, J. Mouzna, and H. Labiod, "C-DRIVE: clustering based on direction in vehicular environment," in *2011 4th IFIP International Conference on New Technologies, Mobility and Security*, Paris, Feb. 2011, pp. 1–5.

[54] N. Maslekar, M. Boussedjra, J. Mouzna, and H. Labiod, "A stable clustering algorithm for efficiency applications in vanets," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International*, July 2011, pp. 1188–1193.

[55] M. Morales, C. seon Hong, and Y. C. Bang, "An adaptable mobility-aware clustering algorithm in vehicular networks," in *Network Operations and Management Symposium (APNOMS), 2011 13th Asia-Pacific*, Sept 2011, pp. 1–6.

[56] M. Morales, E. J. Cho, C. seon Hong, and S. Lee, "An adaptable mobility-aware clustering algorithm in vehicular networks," *Journal of Computing Science and Engineering*, vol. 6, pp. 227–242, Sept 2012.

[57] P. Fan, "Improving broadcasting performance by clustering with stability for inter-vehicle communication," in *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, April 2007, pp. 2491–2495.

[58] C. Shea, B. Hassanabadi, and S. Valaee, "Mobility-based clustering in vanets using affinity propagation," in *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, Nov 2009, pp. 1–6.

[59] A. Ahizoune, A. Hafid, and R. Ben Ali, "A contention-free broadcast protocol for periodic safety messages in vehicular ad-hoc networks," in *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*, Oct 2010, pp. 48–55.

[60] Y. Gunter, B. Wiegel, and H. P. Grossmann, "Cluster-based medium access scheme for VANETs," in *2007 IEEE Intelligent Transportation Systems Conference*, Bellevue, WA, USA, Sep. 2007, pp. 343–348.

[61] E. Dror, C. Avin, and Z. Lotker, "Fast randomized algorithm for hierarchical clustering in vehicular ad-hoc networks," in *Ad Hoc Networking Workshop (Med-Hoc-Net), 2011 The 10th IFIP Annual Mediterranean*, June 2011, pp. 1–8.

[62] E. Dror, C. Avin, and Z. Lotker, "Fast randomized algorithm for 2-hops clustering in vehicular ad-hoc networks," *Ad Hoc Networks*, vol. 11, no. 7, pp. 2002 – 2015, 2013.

[63] M. Almalag and M. Weigle, "Using traffic flow for cluster formation in vehicular ad-hoc networks," in *Local Computer Networks (LCN), 2010 IEEE 35th Conference on*, 2010, pp. 631–636.

[64] M. Almalag, S. Olariu, and M. Weigle, "Tdma cluster-based mac for vanets (tc-mac)," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2012 IEEE International Symposium on a*, June 2012, pp. 1–6.

[65] M. Almalag, S. Olariu, M. Weigle, and S. El-Tawab, "Peer-to-peer file sharing in vanets using tc-mac," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*, March 2013, pp. 84–89.

[66] "VWCA: An efficient clustering algorithm in vehicular ad hoc networks," *Journal of Network and Computer Applications*, vol. 34, no. 1, p. 207.

[67] T. Gazdar, A. Benslimane, and A. Belghith, "Secure clustering scheme based keys management in vanets," in *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, May 2011, pp. 1–5.

[68] N. Kumar, N. Chilamkurti, and J. Park, "Alca: agent learning-based clustering algorithm in vehicular ad hoc networks," *Personal and Ubiquitous Computing*, vol. 17, no. 8, pp. 1683–1692, 2013.

[69] H. Su and X. Zhang, "Clustering-based multichannel mac protocols for qos provisionings over vehicular ad hoc networks," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3309–3323, Nov 2007.

[70] X. Zhang, H. Su, and H.-H. Chen, "Cluster-based multi-channel communications protocols in vehicle ad hoc networks," *Wireless Communications, IEEE*, vol. 13, no. 5, pp. 44–51, October 2006.

[71] A. Ahizoune and A. Hafid, "A new stability based clustering algorithm (sbca) for vanets," in *Local Computer Networks Workshops (LCN Workshops), 2012 IEEE 37th Conference on*, Oct 2012, pp. 843–847.

[72] F. Yang, Y. Tang, and L. Huang, "A multi-channel cooperative clustering-based mac protocol for vanets," in *Wireless Telecommunications Symposium (WTS), 2014*, April 2014, pp. 1–5.

[73] T. Taleb, A. Benslimane, and K. Ben Letaief, "Toward an effective risk-conscious and collaborative vehicular collision avoidance system," *IEEE Trans. Veh. Technol.*, vol. 59, no. 3, pp. 1474–1486, Mar. 2010. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5382585

[74] L. Zhang and H. El-Sayed, "A novel cluster-based protocol for topology discovery in vehicular ad hoc network," *Procedia Computer Science*, vol. 10, pp. 525 – 534, Aug 2012.

[75] T. Taleb and A. Benslimane, "Design guidelines for a network architecture integrating vanet with 3g amp; beyond networks," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, Dec 2010, pp. 1–5.

[76] A. Benslimane, T. Taleb, and R. Sivaraj, "Dynamic clustering-based adaptive mobile gateway management in integrated VANET & 3G heterogeneous wireless networks," *IEEE J. Select. Areas Commun.*, vol. 29, no. 3, pp. 559–570, Mar. 2011.

[77] S. Ucar, S. Ergen, and O. Ozkasap, "Vmasc: Vehicular multi-hop algorithm for stable clustering in vehicular ad hoc networks," in *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, April 2013, pp. 2381–2386.

[78] S. Ucar, S. Coleri Ergen, and O. Ozkasap, "Multi-hop cluster based ieee 802.11p and lte hybrid architecture for vanet safety message dissemination," *Vehicular Technology, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2015.

[79] G. El Mouna Zhioua, N. Tabbane, H. Labiod, and S. Tabbane, "A fuzzy multi-metric qos-balancing gateway selection algorithm in a clustered

vanet to lte advanced hybrid cellular network," *Vehicular Technology, IEEE Transactions on*, vol. 64, no. 2, pp. 804–817, Feb 2015.

[80] I. Tal and G.-M. Muntean, "Towards smarter cities and roads: A survey of clustering algorithms in vanets," *Convergence of Broadband, Broadcast, and Cellular Network Technologies*, pp. 16–50, 2014.

[81] W. Liu, C. Chiang, H. Wu, and C. Gerla, "Routing in Clustered Multihop Mobile Wireless Networks with Fading Channel," in *Proc. IEEE SICON'97*, Apr. 1997, pp. 197–211.

[82] R. A. Santos, R. Edwards, and N. L. Seed, "Inter vehicular data exchange between fast moving road traffic using an ad-hoc cluster-based location routing algorithm and 802.11b direct sequence spread spectrum radio," in *PostGraduate Networking Conference (2003)*, April 2003.

[83] R. A. Santos, R. Edwards, and A. Edwards, "Cluster-based location routing algorithm for inter-vehicle communication," in *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, vol. 2, Sept 2004, pp. 914–918 Vol. 2.

[84] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," in *Proceedings WMCSA'99. Second IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, LA, USA, 1999, pp. 90–100.

[85] J. Blum, A. Eskandarian, and L. Hoffman, "Challenges of intervehicle ad hoc networks," *IEEE Trans. Intell. Transport. Syst.*, vol. 5, no. 4, pp. 347–351, Dec. 2004.

[86] C. Cooper, A. Mukunthan, M. Ros, D. Franklin, and M. Abolhasan, "Dynamic environmental fading in urban vanets," in *IEEE International Conference on Communications (ICC) 2014*, 2014.

[87] A. Mukunthan, C. Cooper, D. Franklin, M. Abolhasan, F. Safaei, and M. Ros, "Experimental validation of the corner propagation model based on signal power measurements in a vehicular environment," in *IEEE WCNC 2013*. Shanghai: IEEE WCNC, 2013.

[88] L. Bernado, T. Zemen, F. Tufvesson, A. Molisch, and C. Mecklenbrauker, "The (in-) validity of the wssus assumption in vehicular radio channels," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on*, Sept 2012, pp. 1757–1762.

[89] B. Hassanabadi, C. Shea, L. Zhang, and S. Valaee, "Clustering in vehicular ad hoc networks using affinity propagation," *Ad Hoc Networks*, vol. 13, Part B, no. 0, pp. 535 – 548, 2014.

[90] C. Cooper, M. Ros, F. Safaei, D. Franklin, and M. Abolhasan, "Simulation of contrasting clustering paradigms under an experimentally-derived channel model," in *IEEE Vehicular Technology Conference (VTC) Fall 2014*, 2014.

[91] C. Cooper and A. Mukunthan, "Urban radio attenuation emulator," 2013. [Online]. Available: http://euclid.elec.uow.edu.au/

[92] "The Network Simulator NS-2," Accessed on 02-11-2014. [Online]. Available: http://www.isi.edu/nsnam/ns/

[93] R. Barr, Z. J. Hass, and R. van Renesse, "JiST/SWANS: Java in Simulation Time / Scalable Wireless Ad hoc Network Simulator," http://jist.ece.cornell.edu/. [Online]. Available: http://jist.ece.cornell.edu/

[94] "The Network Simulator NS-3," Accessed on 12-11-2014. [Online]. Available: http://www.isi.edu/nsnam/ns/

[95] MATLAB, *version 7.10.0 (R2010a)*. Natick, Massachusetts: The MathWorks Inc., 2010.

[96] A. Varga, "OMNeT++," 2009. [Online]. Available: http://www.omnetpp.org/

[97] C. Sommer, "Veins (Vehicles in network simulation)," 2010. [Online]. Available: http://veins.car2x.org/

[98] X. Zeng, R. Bagrodia, and M. Gerla, "Glomosim: A library for parallel simulation of large-scale wireless networks," *SIGSIM Simul. Dig.*, vol. 28, no. 1, pp. 154–161, Jul. 1998.

[99] M. Treiber, "Traffic Simulator 3.0." [Online]. Available: {http://vwisb7.vkw.tu-dresden.de/~treiber}

[100] J. Härri, F. Filali, C. Bonnet, and M. Fiore, "Vanetmobisim: Generating realistic mobility patterns for vanets," in *Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks*, ser. VANET '06. New York, NY, USA: ACM, 2006, pp. 96–97. [Online]. Available: http://doi.acm.org/10.1145/1161064.1161084

[101] *NCTUns 4.0: An Integrated Simulation Platform for Vehicular Traffic, Communication, and Network Researches*, 2007. [Online]. Available: http://dx.doi.org/10.1109/vetecf.2007.437

[102] P. Gburzynski and I. Nikolaidis, "Wireless network simulation extensions in side/smurph," in *Simulation Conference, 2006. WSC 06. Proceedings of the Winter*, Dec 2006, pp. 2225–2233.

[103] "OPNET Network Performance Management," 2014, Accessed on 12-11-2014. [Online]. Available: http://www.riverbed.com/products/performance-management-control/opnet.html?redirect=opnet

[104] E. Giordano, R. Frank, G. Pau, and M. Gerla, "CORNER: a radio propagation model for VANETs in urban scenarios," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1280–1294, Jul. 2011.

[105] P.-J. Wang, C.-M. Li, and H.-J. Li, "Influence of the shadowing on the information transmission distance in inter-vehicle communications," in *IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*. IEEE, 2009, pp. 3015–3019.

[106] E. Giordano, R. Frank, G. Pau, and M. Gerla, "CORNER: a realistic urban propagation model for VANET," in *Seventh International Conference on Wireless On-demand Network Systems and Services (WONS)*. IEEE, Feb. 2010, pp. 57–60.

[107] A. Viklund, "Mixim project," 2011. [Online]. Available: http://mixim.sourceforge.net/

[108] D. Krajzewicz, E. Nicolay, and M. Behrisch, "SUMO - simulation of urban mobility," 2011. [Online]. Available: http://sumo.sourceforge.net/

[109] A. Wegener, M. Piórkowski, M. Raya, H. Hellbruck, S. Fischer, and J. Hubaux, "TraCI: an interface for coupling road traffic and network simulators," in *Proceedings of the 11th communications and networking simulation symposium on - CNS '08*, Ottawa, Canada, 2008, p. 155.

[110] S.-Y. Wang and Y.-B. Lin, "Nctuns network simulation and emulation for wireless resource management," *Wireless Communications and Mobile Computing*, vol. 5, no. 8, pp. 899–916, 2005.

[111] Y. Hormann, H. Grossmann, W. H. Khalifa, M. Salah, and O. Karam, "Simulator for inter-vehicle communication based on traffic modeling," in *Intelligent Vehicles Symposium, 2004 IEEE*, June 2004, pp. 99–104.

[112] M. I. Krzywinski, J. E. Schein, I. Birol, J. Connors, R. Gascoyne, D. Horsman, S. J. Jones, and M. A. Marra, "Circos: An information aesthetic for comparative genomics," *Genome Research*, 2009. [Online]. Available: http://genome.cshlp.org/content/early/2009/06/15/gr.092759.109.abstract

Dr. Craig Stephen Cooper earned his Bachelors Degree in Computer Engineering at the University of Wollongong in 2010, and completed his PhD in Telecommunications in 2015. His research involved vehicular communication and clustering. He has also worked at CSIRO in their wireless sensor network division. Outside of his research, his interests include game and multimedia programming, linguistics, and astronomy.

Dr. Daniel Franklin completed his PhD in Electrical Engineering Enhancements to Channel Models, DMT Modulation and Coding for Channels Subject to Impulsive Noise, at the University of Wollongong in 2007. He also holds a Bachelor of Engineering (electrical) Honours I, University of Wollongong, 1995-1998 and has been a member of the Institution of Electrical and Electronics Engineers since 1998. He is currently a Senior Lecturer in the Faculty of Engineering and Information Technology at the University of Technology, Sydney. His research and commercial interests include wireless networks (routing protocols, mesh networking, modulation and coding), medical radiation physics (solid-state detectors for PET, gamma-ray dosimetry), embedded and real-time systems, and analog and digital electronics.

Dr. Montserrat Ros received the B.E.(Hons1)/B.Sc. double degree with majors in Computer Systems Engineering and Mathematics (2000), and the Ph.D. degree in Computer Engineering (2007) from the University of Queensland (UQ) in Brisbane, QLD, Australia. She is currently a Senior Lecturer at the University of Wollongong and has won two UOW Vice-Chancellors awards for Teaching and Community Engagement.

Farzad Safaei graduated from the University of Western Australia with the degree of Bachelor of Engineering (Electronics) and obtained his PhD in Telecommunications Engineering from Monash University, Australia. Currently, he is the Professor of Telecommunications Engineering and Managing Director of ICT Research Institute at the University of Wollongong. Before joining the University of Wollongong, he was the Manager of Internetworking Architecture and Services Section in Telstra Research Laboratories.

A/Prof. Mehran Abolhasan completed his B.E in Computer Engineering and PhD in Telecommunications on 1999 and 2003 respectively at the University of Wollongong. From July 2003, he joined the Smart Internet Technology CRC and Office of Information and Communication Technology within the Department of Commerce in NSW, where he proposed a major status report outlining strategies and new projects to improve the communications infrastructure between the NSW Emergency Services Organisations. In July 2004, he joined the Desert knowledge CRC and Telecommunication and IT Research Institute to work on a joint project called the Sparse Ad hoc network for Deserts project (Also known as the SAND project). During 2004 to 2007. A/Prof. Abolhasan led a team of researchers at TITR to develop prototype networking devices for rural and remote communication scenarios. Furthermore, he led the deployment of a number of test-beds and field studies in that period. In 2008, A/Prof. Abolhasan accepted the position of Director of Emerging Networks and Applications Lab (ENAL) at the ICTR institute. During his time as the Director of ENAL, A/Prof. Abolhasan won a number of major research project grants including an ARC DP project and a number of CRC and other government and industry-based grants. He also worked closely with the Director of ICTR in developing future research directions in the area wireless communications. In March 2010, A/Prof. Abolhasan accepted the position of Senior Lecturer at the School of Computing and Communications within the faculty of Engineering and IT (FEIT) at the University of Technology Sydney , where he is now an Associate Professor. A/Prof. Abolhasan has authored over 90 international publications and has won over one million dollars in research funding. His Current research Interests are in Wireless Mesh, Wireless Body Area Networks, 4th Generation Cooperative Networks and Sensor networks. He is currently a Senior Member of IEEE.

TABLE VIII: Surveyed algorithms and their validation methodologies. "N/S" stands for "Not Simulated"

Algorithm	Simulator	Channel Model	Comparison	Based upon
DMAC	N/S	N/S	N/S	LID/HD
GDMAC	ns-2	Static R_{TX}	DMAC	DMAC
MDMAC	JiST/SWANS++	Log-Normal	DMAC	DMAC
DBC	JiST/SWANS++	Log-Normal	HD	MDMAC
CCA	Unspecified	Unspecified	MDMAC	MDMAC
CBLR	OPNET	Unspecified	None	Original
CBMAC	Custom	Custom	None	CBLR
RMAC	ns-2	Static R_{TX}	DMAC	Original
HCA	VEINS	Static R_{TX}	k-ConID	Original
C-DRIVE	NCTUns	Static R_{TX}	None	Original
K-hop	ns-2	Two-Ray	None	MOBIC
ALM	SIDE/SMURPH	FSPL	GDMAC	MOBIC
C-RACCA	ns-2	Two-Ray	None	Original
CMGM	ns-2	Two-Ray	None	C-RACCA
MI-VANET	ns-2	Static R_{TX}	None	Original
Sp-Cl	Unspecified	Unspecified	LID	Original
E-Sp-Cl	Custom	FSPL+Shadowing	Sp-Cl	Sp-Cl
PPC	ns-2	Static R_{TX}	LID, HD	Original
AMACAD	Custom	Static R_{TX}	LID, MOBIC, DGMA, MobHid	Original
UF	Traffic Simulation 3.0	Unspecified	LID, HD	LID/HD
CSBP	JiST/SWANS++	Unspecified	None	UF
TC-MAC	ns-3	Static R_{TX}	LID, HD, UF	UF
APROVE	ns-2	Static R_{TX}	MOBIC, ALM, PPC	Original
CCP	Unspecified	Unspecified	None	Original
SBCA	ns-2	Static R_{TX}	CCP	CCP
TBC	Custom	Unspecified	CBMAC, PPC	Original
PC	GloMoSim	Static R_{TX}	LID	Original
VPC	ns-2	Static R_{TX}	PC	PC
CF-IVC	N/S	N/S	N/S	PC
FBLA	ns-2	Nakagami	APROVE, CCP	Original
UOFC	ns-2	Unspecified	LID, UF	UF
VWCA	MATLAB	Static R_{TX}	LID, HD, WCA	WCA
TACR	Unspecified	Unspecified	None	Original
DBA-MAC	ns-2	Static R_{TX}	None	Original
CPTD	ns-2	Static R_{TX}	None	Original
MCC-MAC	MATLAB	Unspecified	None	Original
MCTC	MATLAB	Unspecified	HD	Original
CBPKI	VEINS	Static R_{TX}	None	Original
CAC	Unspecified	Unspecified	None	Original
ALCA	VANET MobiSim	Static R_{TX}	None	Original
MCA-VANET	ns-3	FSPL+fading	MOBIC	Original
NMCS	N/S	N/S	N/S	Original
FQGWS	MATLAB	Static T_{TX}	CMGM, C-DRIVE	Original
DMCNF	ns-2	Unspecified	K-Hop	Original
VMaSC	ns-3	Static R_{TX}	K-Hop, MDMAC	Original