SAE TECHNICAL PAPER SERIES

2014-36-0126 E

New trends in Automotive Embedded System Design

ANTONIO CESAR GALHARDI

SAE BRASIL

AFFILIATED TO SAE INTERNATIONAL

Congresso 2014 SAE BRASIL

23rd SAE BRASIL International Congress and Display
São Paulo, Brasil
October, 30th to 2nd

AV. PAULISTA, 2073 - HORSA II - CJ. 1003 - CEP 01311-940 - SÃO PAULO – SP

ISSN 0148-7191

2014-36-0126

New Trends in Automotive Embedded System Design

Antonio César Galhardi
Centro Estadual de Educação Tecnológica Paula Souza

Abstract

Today, though still relatively stable, the roles of carmakers and their suppliers are undergoing a period of stress caused by the increased importance and added value of electronics. Most of today's embedded systems are required to work where the characteristics of the computational load cannot always be predicted in advance. The embedded systems are, by nature, inherently real-time. Moreover, most of embedded systems work under several resource constraints, due to space, weight, energy, and cost limitations imposed by the specific application. Still timely responses to events have to be provided within precise timing constraints in order to guarantee a desired level of performance. Novel methods and tools for system-level analysis and modeling are needed not only for predictability and composability when partitioning end-to-end functions at design time (and later, at system integration time), but also for providing guidance and support to the designer in the very early stage where the electronics and software architectures of product lines are evaluated and selected. The critical architecture-evaluation and selection design-process phase affects profoundly a product line's cost, performance, and quality. The objective of this document is to present the characteristics of modern embedded applications, the problems of the current approaches and discusses the new trends.

Introduction

As computers became smaller in size and the advances in Information and Communication Technology - ICT, had been made, a range of new applications involving automotive embedded computing can emerge. These applications may be enabled by the arrival of general-purpose computing platforms in vehicles. Many of these applications will involve the collection, processing and distribution of data sampled by sensors on large numbers of vehicles.

Over recent decades, the computer technology has impacted on many areas of society. These innovations have brought increases in efficiency, productivity, safety and many other benefits. The automotive vehicles has enjoyed many aspects of this revolution, with innovations ranging from electronic stability control until the urban traffic control systems and electronic toll collection systems on the road network. The recent trends in Information and Communication Technology mean that a further range of applications will become feasible in the near future. In particular, applications involving communication amongst large numbers of vehicles across a wide geographic scale are likely to be implemented. Then, the main focus of this paper is to show how these trends can be applied in the future, specially the Vanets, I mean: involving communication amongst large numbers of vehicles on the roads.

Theory

The term ubiquitous computing [1] was coined as an area of Computer Science research in which computers are deployed pervasively throughout everyday environments. Various related terms have been employed [2]: sentient computing, pervasive computing, ambient computing, and more. This indicates the goal of computers is shrinking in size and becoming embedded into the environment. As this vision is increasingly real, the environment becomes populated with intelligent devices: smart buildings, smart desks, and other ,"smart" nouns become a reality.

In this field born the term: "Intelligent Transport Systems" – ITS that is a term applied to advances in this area, covering a variety of fields, including integrated transportation systems, urban traffic control systems, and driver assistance systems.

With the intend of ubiquitous computing vision be attained, all vehicles, all traffic lights, all road lanes and all road signs eventually will become addressable nodes on the Internet.

Traditionally, computing has been primarily applied to vehicular technology for safety related purposes. After these applications, the next beneficiaries of computer technology in the vehicle industry have been entertainment-oriented applications. Beyond these areas, there have been very few examples of computing in vehicles. However, as on-board processing and communications become cheaper and less invasive, more applications will become possible.

At this point, it is necessary to differentiate the automotive embedded computing of the automotive electronics. Although quite similar, the main difference is that the first one focuses on the software, while the second focuses on the hardware. The onboard electronics of course, was first introduced in vehicles, with a number of sensors and devices, later when there was a need or opportunity and take charge of the vehicle, regardless of the driver, for various applications, it was borne the term embedded computing.

Traditionally, computing has been primarily applied to vehicular technology for safety related purposes. After these applications, the next beneficiaries of computer technology in the vehicle industry have been entertainment-oriented applications. Beyond these areas, there have been very few examples of computing in vehicles. However, as on-board processing and communications become cheaper and less invasive, more applications will become possible.

A modern vehicle contains many microprocessors governing various aspects of the vehicle's operation, from the anti-lock braking system through to the CD player unit. A vehicle's engine management unit (EMU) is responsible for monitoring data from sensors attached to the components of the engine and controlling actuators such as that which controls the fuel pump.

Some systems installed by the manufacturer of the vehicle use general purpose CPUs running commodity operating systems. For example, BMW's iDrive system, a software interface to allow the driver to interact with the vehicle, uses the VxWorks real-time operating system.

There are many instances of after-market, third-party devices being used in vehicles.

Off-the-shelf navigation units are becoming increasingly popular, employ similar technology. Pay-as-you-drive insurance schemes [3], pioneered by Norwich Union in the UK, involve a "black box" inside the vehicle that records parameters relating to the vehicle's whereabouts and communicate over a cellular network to report back to the insurance company so that a premium can be computed. Also, several proposed schemes for implementing road-user charging such as the HGV tolling scheme in Germany [4] involve an in-vehicle device which communicates with roadside interrogators.

Rather than all of these systems, and many others, being installed in separate black boxes, each with their own CPUs, it is conceivable that, in the future, vehicle manufacturers could construct vehicles fitted with general-purpose CPUs that are shared by all such applications.

Due to the highly competitive nature of the vehicle market, manufacturers are keen to keep both the fuel consumption and the mass of its vehicles as low as possible. Therefore, if general-purpose CPUs are to be provided by manufacturers, it must be the case that they bring utility to the driver which outweighs any increase in the vehicle's fuel consumption or mass. In some cases, the overall mass may decrease if a CPU is shared between applications. Either way, in the future, embedded computing in vehicles will become increasingly feasible if Moore's Law [5] continues to hold.

At present, vehicle manufacturers produces proprietary applications which make use of built-in hardware. Other vendors supply after-market products that can be added to the vehicle. However, it is rare for these products to integrate seamlessly with each other and with the vehicle's own computer software and hardware.

As a result, applications that could share common hardware tend to be deployed in isolation from each other. This inevitably causes the need for lots of separate wires and circuit boards, which is expensive. Instead, it is preferable for applications to share common hardware to avoid unnecessary redundancy. For example, GPS units are required by a range of in-vehicle devices: navigation units, black-box devices used in pay-as-you-drive insurance, and certain implementations of road-user charging. Provided that the vehicle can supply a GPS unit of sufficient accuracy to satisfy all these applications' demands, using this single device would be a preferable solution.

Modern vehicles contain a vast array of sensors. Some monitor various aspects of the engine's operation, such as thermometers and fuel flow rate sensors. Others participate in driver-level applications, such as sensors which detect whether a door is closed, and rear-mounted range-finding sensors which detect the distance to the nearest object behind the vehicle.

Sensors may vary widely in terms of the frequency of sampling and the amount of data they produce. For example, a thermometer may output a single byte and be sampled every few seconds; a video camera to assist with reversing may output several megabytes per second.

In the past, wiring was added to vehicles for each new sensor added. However, as vehicles gained increasing numbers of sensors, this added considerable weight, consumed a significant amount of space and made adherence to reliability standards difficult [6].

This led to the adoption of a bus-based approach, with wiring shared between sensors. In the mid-1980s, the controller area network (CAN) was developed, derivatives of which are still in widespread use in some modern vehicles [6]. The vehicles may contain multiple can buses: perhaps a low-speed bus for the comfort electronics and a high-speed bus for real-time systems involving engine management.

It is conceivable that in future vehicles will contain general-purpose sensing and computing platforms to avoid this situation. This may be spurred by an increasing demand for a variety of applications involving in-vehicle computing.

A variety of applications, some of which are already emerging [7]:

- Entertainment applications will expand beyond listening to music and watching films. The AMI-C (Automotive Multimedia Interface Collaboration) has published standards for automotive interfaces, which enable a wide variety of applications that are primarily entertainment-oriented [8].
- Mobile commerce applications involve purchasing products and performing other business transactions whilst on the move [9].
- Location-based services may include local weather or traffic data, or advertisements from local businesses.
- Remote operation applications involve the operation of devices in a remote location such as switching on the home heating in advance of arriving at a destination.
- Asset tracking, a delivery or haulage vehicle can report its position and its current inventory.

- Congestion information. An in-vehicle navigation unit could receive a broadcast of current traffic conditions determined for example by aggregating sensor data gathered from in-road vehicle sensors. Further, the navigation unit could proactively query a remote congestion information service regarding the roads that it considers to incorporate into a route. Such a service could collect data from vehicles [10], and after processed to provide accurate estimates of journey times on particular roads.
- Real-time weather. Similarly, vehicles could download current weather observations and forecasts for the local region. In addition, the network could gather data from vehicles which contain meteorological sensors and redistribute the aggregated data back to interested vehicles.
- Road hazard detection. Potential hazards on the roads could be detected if data from many vehicles' braking systems were gathered [11]. When a substantial number of vehicles are found to brake sharply at a particular location, it could be marked as a potential hazard.
- Map generation. Similarly, the location histories of large numbers of vehicles could be combined and aggregated to provide updates to digital road maps in real-time. For example, new roads opening or road closures could be detected and the information delivered back to the vehicles [12].
- Slot booking [13]. Systems which co-ordinate the efficient flow of traffic through with a known Internet host.
- Fleet management. Organizations owning a number of vehicles need to be able to manage them centrally, perhaps so that their future movements can be planned and their routes optimized.
- Gaming. Occupants of vehicles can play games [14] with occupants of other vehicles or non-mobile participants.
- Road user charging. Many suggested implementations of electronic toll collection or congestion charging scan and dynamic pricing require the in-vehicle unit to keep track of current prices and pay on behalf of the driver. Alternatively, a peer-to-peer implementation of congestion charging involving communication between vehicles that preserves the privacy of the users has been proposed by Harle and Beresford [15].
- Intersection collision avoidance. This class of safety applications involves the self-organizing co-ordination of the movement of traffic, such as negotiation between vehicles approaching a road junction [16].
- Accident notification. In-vehicle systems that detect collisions and automatically notify the emergency services of the location and the nature of the collision enable faster responses.
- Journey scheduling. A personal device could proactively suggest not only routes to travel to a destination, but also the modes of transport to use. These recommendations could be derived from the aggregated information it receives from a journey-time and timetable service.

Some of the applications described before involve the co-ordination of computation to process sensor data amongst large numbers of vehicles. This can make the design and implementation of these applications challenging. Specifically, there are four particular challenges facing application designers:

1. Vehicles need a means of wireless communication with other vehicles and with the Internet.
2. The mobility of vehicles means that the characteristics of the communication links will vary over time. Applications must be able to adapt to changes in connectivity.
3. Applications will have goals that relate to the entire network rather than to specific individual vehicles. This means that the set of participating vehicles will not necessarily be known before the application is executed. Applications must therefore be able to adapt to whichever computational resources are available to execute it at run-time.
4. A network including computers on-board vehicles will be heterogeneous. Participating computers will differ in terms of their levels of computational, storage, energy and communication resources. Applications should make the most efficient use possible of these resources.

In order to share data (processed or unprocessed) with other vehicles, and to obtain data from other nodes on the Internet (which may themselves be vehicles), extra-vehicular wireless communications are required.

Historically, the only form of extra-vehicle communication in the electro-magnetic spectrum was the receipt of broadcast radio signals via an analogue tuner. Later, this system was augmented by small amounts of digital information being broadcast on public radio channels via the Radio Data System (RDS).

Using near-ubiquitous cellular networks such as GSM, GPRS and UMTS, vehicles are able to connect to the Internet in the same way as mobile telephones. Recently, with systems such as General Motors' OnStar system, two-way communication to and from vehicles has become available. OnStar provides a remote door-unlocking facility and an automatic crash detection system which notifies the emergency services of the vehicle's location.

A vehicular network can be centralized or decentralized. A centralized network uses a ubiquitous wireless communication technology such as GSM or UMTS, and vehicles have globally unique identifiers. On the other hand, a decentralized network uses local communication between nearby nodes. As vehicles move around, they may move and come out of range of other nodes. Hence this variety of network is largely ad-hoc, and its topology is highly variable and may be somewhat unpredictable.

Vehicular networks have particular characteristics that distinguish them from other classes of network:

- Despite the fast speeds of vehicles, their movements are constrained and their mobility is hence somewhat predictable. Vehicles can only use roads, which occupy a comparatively small proportion of the surface of the Earth.

- Due to the nodes' mobility, energy resources are necessarily constrained. However, energy is in more plentiful supply than in battery-powered devices. Vehicles have a plentiful supply of fuel, and the consumption of energy due to computing and communications equipment is likely to be significantly lower than that consumed as a result of the vehicle's other operations.
- Vehicles may be assumed an accurate knowledge of their position [17], by using a global, outdoor positioning system such as GPS, and may also possess a road map of the local area [18].

The term "mobile ad-hoc network", or Manet, is used to describe networks involving mobile nodes which do not rely on the availability of a ubiquitous wireless network to communicate. Manets are usually characterized by nodes having limited computational resources (processor and storage). The vehicular ad-hoc network, or Vanet, is a particular class of Manet in which the mobile nodes are vehicles travelling on roads.

The characteristics of vehicular networks outlined before have important implications for Vanets:

- Networks are highly dynamic due to the mobility of vehicles. Vehicles passing each other on a motorway may only be within communication range for a matter of seconds, so communication links between vehicles are frequently established and broken.
- The traffic density on roads varies significantly throughout the day, perhaps by a few orders of magnitude. Hence, at off-peak times, vehicles may be disconnected from other vehicles if none are within range [19].

Blum et al [20] have attempted to quantify the distinctions in mobility in Vanets, compared to general Manets, through simulation. They have established that vehicular networks experience very rapid changes in topology due to the high relative speed of vehicles.

Furthermore, even links between vehicles travelling in the same direction are short-lived: for transmissions of 500 ft. range, the links last about one minute on average. In addition, the inter-vehicular networks were found to be subject to frequent fragmentation, in which chunks of the network become isolated from each other.

These factors make the implementation of useful Vanets difficult [17]. Due to the highly dynamic mobility of vehicles, there is no guarantee that the vehicles nearby in one instant will be nearby in the next. Hence, reliably routing a message through a network is a challenge. Vehicles passing on motorways can pass at up to 140 mph, whilst the density of vehicles may vary from as little as one vehicle per kilometer of road to five hundred vehicles per kilometer. In low density situations, a wide transmission range is desirable but in high density situations it would cause too much contention.

The Point-to-point communication is required in applications which involve collaboration between vehicles which are not necessarily collocated. There have been many protocols suggested for routing messages between nodes in Manets. They can be classified into three categories: proactive, reactive and position-based protocols [21]:

- Proactive protocols employ classical routing strategies such as distance-vector routing or link-state routing. These protocols are unsatisfactory for use in Vanets since they maintain state about paths, expecting this information to stay fairly constant over time, which in a vehicular network cannot be assumed.
- Reactive protocols create new routes for each message sent, so do not need to store state about the paths which are not currently in use. Dynamic Source Routing (DSR) [22] is a source routing protocol, in which a message's sender specifies the entire path to the destination in the message's header. In contrast, the ad-hoc on-demand distance vector algorithm (AODV) [23] is an example of destination routing, which adopts a hop-by-hop approach. Intermediate nodes use a local look-up table to determine which node to forward a message to. In a vehicular network, the destination routing approach is likely to fare better than source routing as the dynamic nature of communication links may render a specified path invalid before the message has reached its destination [24].
- Position-based protocols assume that nodes have knowledge of their location, which is periodically broadcast to neighboring nodes and registered with a centralized location service [25]. Routing can then be stateless, performed solely based upon the positional displacement of neighbors relative to the destination node whose location was looked up in the location service. A well-known example of a position based protocol is Greedy Perimeter Stateless Routing (GPSR) [26]. This protocol attempts to move a message towards its destination. When greedy routing can get the message no further, it is routed around the perimeter of the region between that point and the destination.

Most routing protocols are commonly evaluated against a random-waypoint model. However, this is inappropriate for Vanets where mobility is far more tightly constrained. Füßler et al. [21] have instead compared DSR and GPSR by simulating the movements of vehicles in a traffic simulator. Their results show that the position-based approach performs best for communications spanning more than a few hops.

One of the assumptions about Vanets mentioned above is that vehicles may be assumed to know their locations and the local road topology; this makes position-based protocols appear most promising for routing in Vanets. This has given rise to a number of variants of position-based routing protocols tailored specifically to the characteristics of vehicular networks.

Various researches into vehicle-to-roadside and vehicle-to-vehicle communication have been undertaken, although there are few examples of deployments of these systems of any significant scale. Many of the major vehicle manufacturers are prototyping vehicles equipped with Wi-Fi (802.11a/b/g) and dsrc (dedicated short-range communications). These technologies are expected to feature on production vehicles within a few years.

DSRC is a 5.9GHz radio band which has been reserved specifically for inter-vehicular communications. This goes hand-in-hand with the IEEE's 802.11p standard for Wireless Access in Vehicular Environments (wave). This technology is deemed suitable for use for safety-critical applications [27].

The Drive-Thru Internet project has shown that high-speed Internet access to vehicles via roadside 802.11b access points is feasible [28]. A similar system called Mocca has attained similar achievements [29]. However, Bergamo et al [30] point out that performance falls off rapidly in the absence of line-of-sight communication.

Murphy et al. have tested the feasibility of using Bluetooth to transfer data between vehicles and to stationary access points [31]. Their findings are that vehicles travelling at 100 km/h can communicate with the access point for 18 seconds. They have also proposed modifications to the Bluetooth protocol which improves its suitability for this kind of application.

A system employing millimeter-wave radio has been suggested by Kim et al [32]. However, at such high frequencies, propagation distances are very low compared to other radio technologies, so roadside base stations would be required to be of the order of every 100metres. As well as the cost of deployment of such a system, there is the technical challenge of efficiently dealing with the frequent hand-overs that a fast-moving vehicle would require.

Ribeiro [33] has compared the use of Wi-Fi, Wimax, mbwa and 3G communication technologies for their suitability to create a wireless vehicular network which includes base-stations. The most significant difference between Wi-Fi and Wimax was found to be the size of the coverage area, which is an order of magnitude greater for Wimax, since Wi-Fi was originally designed merely for indoor use. Furthermore, Wi-Fi base stations are more restricted in terms of the number of users which can be supported at a given time. Mbwa was found to be very similar to Wimax, although it was specifically designed for mobile use. Ribeiro [33] suggests that Wimax will eventually supersede 3G cellular technology.

Final Thoughts

Technological advances in computing and communications mean that a new range of applications involving computing in vehicles is likely to become feasible. In the future, the implementation of these applications may be spurred by the existence of general purpose computing platforms in vehicles. Such platforms may be provided by vehicle manufacturers in order to avoid redundancy caused by isolated applications or devices which could share common software or hardware. Alternatively, vendors of after-market in-vehicle devices could collaborate to provide interfaces between their devices to allow them to interact. In either scenario, many new applications become possible. Of these applications, some of the most promising yet most challenging to implement involve the collection, processing and distribution of sensor data originating on vehicles.

In these applications, vehicles become sensor platforms and nodes in a large-scale wireless sensor network. Furthermore, different sensors require a variety of treatments in terms of sampling, communications and storage. Perhaps the largest challenge involved in implementing applications which make use of vehicular sensor data is to design them to operate in a scalable fashion. Applications of this kind are usually most efficacious when they use data from many vehicles. For example, if the vehicles operate in different geographic regions then the output becomes more complete.

Furthermore, using more vehicles means that more data is available, so that random errors become less significant, potentially leading to a higher accuracy of the output. Hence, the applications must be able to simultaneously handle large numbers, perhaps millions, of vehicles in order to maximize utility.

An application which makes use of vehicular sensor data is that of automatic road map generation involves the use of algorithms from image processing and graph theory to process sets of location traces into a map of the road network. Generating road maps in this way has some advantages over traditional map-making techniques. For example, the opening of new roads and the closure of existing roads can be reflected in the map in real-time.

Whilst this application involves the processing of one particular type of sensor data, location histories, different applications will make use of other types of data. These applications will also infer higher-level information from low-level raw data. More complex applications, such as an urban traffic management system which optimizes the movement of vehicles according to journey time, levels of pollution and congestion, could be implemented along similar lines.

The requirement for the scalability of these applications leads to the question of how to arrange the collection and processing of vehicular sensor data in the most resource-efficient manner. Until now the right answer isn't clear. One option is a centralized approach in which all of the vehicles transmit their data to a fixed server which performs the processing. However, this suffers from the problems of a communication bottleneck and the presence of a single point of failure. Moreover, it may be difficult for vehicles to communicate their data to a central server cheaply and in a timely fashion. Instead, a distributed systems approach involving less communication may be preferable.

The traditional approach to programming wireless sensor networks expects the programmer to decide which processors should perform what computation. Furthermore, the particular computational resources and communication infrastructure available in the network may not be known until run-time. There are several proposals for improved techniques for programming wireless sensor networks. However, these approaches are either too inflexible or do not sufficiently abstract away from knowledge of the network in which the application is to execute, so are unsuitable.

A country or even planet-sized network will contain vast quantities of computational resource, constituted from many individual devices with varying characteristics, and there will be many means of communication from one node to another. Judicious selection of processors and communication links to use is crucial in this kind of network because processing may be very costly on some processors, and transferring data may be very costly on some communication links. This is a particular instance of the general problem of making efficient use of available resources.

References

1. Mark Weiser. The computer for the twenty-first century. *Scientific American*, 265(3):94–104, September 1991.
2. Andy Hopper. Sentient computing (abridged and updated version of the Royal Society Clifford Paterson Lecture, 1999). In *Computer Systems: Theory, Technology, and Applications: A Tribute to Roger Needham*, Monographs in Computer Science, pages 125–131. Springer-Verlag, December 2003.
3. Todd Litman. Distance-based vehicle insurance as a TDM strategy. Available online at http://www.vtpi.org/dbvi.pdf, December 2004
4. Silke Jung. HGV tolls in Germany: Innovative, environmentally friendly and fair. *The IEE Road Transport Symposium*, pages 5/1–7, December 2005.
5. Gordon E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8):114–117, April 1965.
6. Gabriel Leen and Donal Heffernan. Expanding automotive electronic systems. *IEEE Computer*, 35(1):88–93, January 2002.
7. David N. Cottingham and Jonathan J. Davies. A vision for wireless access on the road network. In *Proceedings of the 4th International Workshop on Intelligent Transportation (WIT 2007)*, pages 25–30, Hamburg, Germany, March 2007.
8. AMI-C. AMI-C use cases. Available from http://www.ami-c.org/, January 2003.
9. Upkar Varshney. Vehicular mobile commerce. *IEEE Computer*, 37(12):116–118, December 2004.
10. Peter Day, Jianpang Wu, and Neil Poulton. Beyond real time. *ITS International*, 12(6):55–56, Nov–Dec 2006.
11. Marco Gruteser and Dirk Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proceedings of 1st ACM/USENIX International Conference on Mobile Systems, Applications and Services (MobiSys)*, pages 31–42, San Francisco, CA, USA, May 2003. ACM Press.
12. Jonathan J. Davies, Alastair R. Beresford, and Andy Hopper. Scalable, distributed, real-time map generation. *IEEE Pervasive Computing*, 5(4):47–54, Oct–Dec 2006.
13. Markos Papageorgiou and Apostolos Kotsialos. Freeway ramp metering: An overview. *IEEE Transactions on Intelligent Transportation Systems*, 3(4):271–281, December 2002.
14. Kieran Mansley, David Scott, Alastair Tse, and Anil Madhavapeddy. Feedback, latency, accuracy: Exploring tradeoffs in location-aware gaming. In *SIGCOMM 2004 Workshops: Proceedings of ACM SIGCOMM 2004 workshops on NetGames'04*, pages 93–97, Portland, Oregon, USA, August 2004. ACM Press.
15. Robert Harle and Alastair Beresford. Keeping big brother off the road. *IEE Review*, 51(10):34–37, October 2005.
16. Florian Dötzer, Florian Kohlmayer, Timo Kosch, and Markus Strassberger. Secure communication for intersection assistance. In *Proceedings of the 2nd International Workshop on Intelligent Transportation (WIT 2005)*, Hamburg, Germany, March 2005.
17. Maziar Nekovee. Sensor networks on the road: The promises and challenges of vehicular ad hoc networks and grids. In *Workshop on Ubiquitous Computing and e-Research*, National e-Science Centre, Edinburgh, Scotland, May 2005.
18. Genping Liu, Bu-Sung Lee, Boon-Chong Seet, Chuan-Heng Foh, Kai-Juan Wong, and Keok-Kee Lee. A routing strategy for metropolis vehicular communications. In *Proceedings of the 18th International Conference on Information Networking*, pages 134–143. Springer, February 2004.
19. Tamer Nadeem, Sasan Dashtinezhad, Chunyuan Liao, and Liviu Iftode. TrafficView: Traffic data dissemination using car-to-car communication. *Mobile Computing and Communications Review*, 8(3):6–19, July 2004.
20. Jeremy J. Blum, Azim Eskandarian, and Lance J. Hoffman. Challenges of intervehicle *Ad Hoc* networks. *IEEE Transactions on Intelligent Transportation Systems*, 5(4):347–351, December 2004.
21. Holger Füßler, Martin Mauve, Hannes Hartenstein, Michael K¨asemann, and Dieter Vollmer. A comparison of routing strategies for vehicular ad hoc networks. Technical Report TR-02-003, Department of Computer Science, University of Mannheim, July 2002.
22. David B. Johnson and David A. Maltz. *Dynamic Source Routing in Ad Hoc Wireless Networks*, volume 353, pages 153–181. Springer US, 1996.
23. Charles E. Perkins and Elizabeth M. Royer. Ad-hoc on-demand distance vector routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA '99)*, pages 90–100, New Orleans, LA, USA, February 1999.
24. Hao Wu, Richard Fujimoto, Randall Guensler, and Michael Hunter. MDDV: A mobility-centric data dissemination algorithm for vehicular networks. In *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks (VANET'04)*, pages 47–56, Philadelphia, PA, USA, October 2004.
25. Hannes Hartenstein, Bernd Bochow, Andr´e Ebner, Matthias Lott, Markus Radimirsch, and Dieter Vollmer. Position-aware ad hoc wireless networks for intervehicle communications: the Fleetnet project. In *MobiHoc '01: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, pages 259–262, Long Beach, CA, USA, 2001. ACM Press.

26. Brad Karp and H. T. Kung. GPSR: Greedy perimeter stateless routing for wireless networks. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pages 243–254, Boston, MA, USA, 2000.
27. Jijun Yin, Tamer ElBatt, Gavin Yeung, Bo Ryu, Stephen Habermas, Hariharan Krishnan, and Timothy Talty. Performance evaluation of safety applications over dsrc vehicular ad hoc networks. In *VANET '04: Proceedings of the first ACM workshop on Vehicular ad hoc networks*, pages 1–9, Philadelphia, PA, USA, 2004. ACM Press.
28. Jörg Ott and Dirk Kutscher. Drive-Thru Internet: IEEE 802.11b for "automobile" users. In *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communication Societies (INFOCOM 2004)*, volume 1, March 2004.
29. Marc Bechler, Walter J. Franz, and Lars Wolf. Mobile internet access in FleetNet. In *13. Fachtagung Kommunikation in Verteilten Systemen: Kurzbeitr¨age, Praxisberichte und Workshop E-Learning*, pages 107–118, Leipzig, Germany, April 2003.
30. Pierpaolo Bergamo, Daniela Maniezzo, Kung Yao, Matteo Cesana, Giovanni Pau, Mario Gerla, and Don Whiteman. IEEE802.11 wireless network under aggressive mobility scenarios. In *International Telemetry Conference ITC/USA2003*, Las Vegas, NV, USA, October 2003.
31. Patrick Murphy, Erik Welsh, and J. Patrick Frantz. Using Bluetooth for short-term ad hoc connections between moving vehicles: A feasibility study. In *Proceedings of the 55th IEEE Vehicular Technology Conference (VTC Spring 2002)*, volume 1, pages 414–418, 2002.
32. Hong Bong Kim, Marc Emmelmann, Berthold Rathke, and Adam Wolisz. A radio over fiber network architecture for road vehicle communication systems. In *Proceedings of the 61st IEEE Vehicular Technology Conference (VTC Spring 2005)*, volume 5, pages 2920–2924, May 2005.
33. Connie Ribeiro. Bringing wireless access to the automobile: A comparison of Wi-Fi, WiMAX, MBWA and 3G. In *Proceedings of the 21st Annual Rensselaer at Hartford Computer Science Conference*, April 2005.

Definitions/Abbreviations

ITS – Intelligent Transport Systems

CD – Compact Disk

ICT – Information and Communication Technology

GPS – Global Positioning System

EMU – Engine Management Unit

CPU – Central Processing Unit

VxWorks – A real-time operating system similar to UNIX, produced by Wind River Systems

CAN – Controller Area Network

AMI-C – Automotive Multimedia Interface Collaboration

RDS – Radio Data System

GSM – Global System for Mobile Communications

GPRS – General Packet Radio Service

UMTS – Universal Mobile Telecommunications System

DSR – Dynamic Source Routing

AODV – On-demand Distance Vector Algorithm

WiFi – A trademark of the Wi-Fi Alliance, a class of Local Area Network

IEEE – Institute of Electrical and Electronics Engineers