Mobile Information Systems 6 (2010) 213–227
DOI 10.3233/MIS-2010-0100
IOS Press

Implementation of CAVENET and its usage for performance evaluation of AODV, OLSR and DYMO protocols in vehicular networks

Evjola Spaho[a,*], Leonard Barolli[b], Gjergji Mino[a], Fatos Xhafa[c], Vladi Kolici[d] and Rozeta Miho[d]
[a]*Graduate School of Engineering, Fukuoka Institute of Technology* (*FIT*), *3-30-1 Wajiro-Higashi, Higashi-Ku, Fukuoka 811-0295, Japan*
[b]*Department of Information and Communication Engineering, Fukuoka Institute of Technology* (*FIT*), *3-30-1 Wajiro-Higashi, Higashi-Ku, Fukuoka 811-0295, Japan*
[c]*Department of Languages and Informatics Systems, Polytechnic University of Catalonia, Jordi Girona 1-3, 08034 Barcelona, Spain*
[d]*Department of Electronic and Telecommunication, Polytechnic University of Tirana, Mother Teresa Square, Nr.4, Tirana, Albania*

Abstract. Vehicle Ad-hoc Network (VANET) is a kind of Mobile Ad-hoc Network (MANET) that establishes wireless connection between cars. In VANETs and MANETs, the topology of the network changes very often, therefore implementation of efficient routing protocols is very important problem. In MANETs, the Random Waypoint (RW) model is used as a simulation model for generating node mobility pattern. On the other hand, in VANETs, the mobility patterns of nodes is restricted along the roads, and is affected by the movement of neighbour nodes. In this paper, we present a simulation system for VANET called CAVENET (Cellular Automaton based VEhicular NETwork). In CAVENET, the mobility patterns of nodes are generated by an 1-dimensional cellular automata. We improved CAVENET and implemented some routing protocols. We investigated the performance of the implemented routing protocols by CAVENET. The simulation results have shown that DYMO protocol has better performance than AODV and OLSR protocols.

1. Introduction

During recent years, there has been an unprecedented growth in wireless networks. This can be attributed to high demand for wireless multimedia services such as data, voice, video, and the development of new wireless standards. There are lots of other driving factors that have led to the rapid and continuous change of the wireless networks worldwide. Mobility is a major driver for mobile networks because mobile users continue to demand access remotely anywhere and anytime. The ever growing need for mobile Internet access, interactive services, training, and entertainment; the need for a single standard for seamless roaming; interoperability across networks; and upward integration of earlier wireless network technologies are also driving factors for new developments in wireless networks. Other driving factors

*Corresponding author. E-mail: evjolaspaho@hotmail.com.

are improvements in RF performance that are attributable to improved antennas, reduction in sources of interference, and the ability to support multiple frequency bands. In recent years, wireless networks are continuing to attract attention for their potential use in several fields such as ad-hoc networks, sensor networks, mesh networks, and vehicular networks [5,9,13,14,17,18,25–27].

Vehicular communication is seen as a key technology for improving road safety and comfort through Intelligent Transportation Systems (ITS). There are many possible application of wireless technologies for vehicular environment [10].

Vehicular Ad Hoc Networks (VANETs) are an instance of ad-hoc networks, which are general-purpose distributed wireless networks interconnected without the need of any centralized infrastructure. VANETs are expected to be massively deployed in upcoming vehicles, because their use can improve the safety of driving and makes new forms of inter-vehicle communications possible as well. Given a mobility model of vehicles, usually a simulator is used to test networking protocols. In this regard, we present a lightweight simulator which can be used to understand the properties of the mobility models of vehicular traffic and their impact on the performance of VANETs. We call this simulator Cellular Automaton based VEhicular NETwork (CAVENET), because its mobility model is built upon a 1-dimensional Cellular Automaton (CA).

The CAVENET separates the problem of mobility model from that of the protocol evaluation, which is performed by means of a network simulator. The properties of the mobility model, e.g. the average transient time towards the stationary state, can be analysed independently of the protocol simulation. Eventually, the movement patterns generated by the mobility model can be mapped into a trace file format suitable for the network simulator.

The Random Waypoint (RW) model has been the earliest mobility model for ad-hoc networks. Basically, in RW every node picks up a random destination and a random velocity at certain points called waypoints. This model has been extended in a number of ways in order to take into account more realistic movements. The simulation of such models has shown the problem of velocity decay, which posed some doubts about the length of the simulation time and the duration of the transient. The problem has been solved by several authors, in particular by Le Boudec [16], who used Palm distributions, and Noble [28]. However, all mobility models considered so far are Short Range Dependent (SRD). This means that every mobile chooses its velocity independently by the others. In the case of VANETs, this assumption is clearly not valid anymore, especially in the case of highway traffic. We show this fact by means of basic simulations performed with CAVENET. For instance, we show that the traffic model strongly affects the statistical structure of the average velocity.

In the particular case of deterministic traffic models, the average velocity is SRD and the transient state depends on the density of the vehicles. In general, the mobility model of VANETs for the simulated variable of interest (e.g. the average velocity) can be Long Range Dependent (LRD) in some cases. This fact poses some problems on how long the simulation should be and how many samples from the starting time should be discarded.

In literature, vehicular mobility models are usually classified as either macroscopic or microscopic. The macroscopic description models gross quantities of interest, such as vehicular density or mean velocity, treating vehicular traffic according to fluid dynamics, while the microscopic description considers each vehicle as a distinct entity, modelling its behaviour in a more precise, but computationally more expensive way. Yet, a micro-macro approach may be seen more as a broad classification schema than a formal description of the models' functionalities in each class [10].

In this work, we consider the vehicular mobility model as a microscopic model. Our simulator is based on 1-dimensional CA model. The CA is a discrete time model of the vehicular traffic. The first

version of CAVENET had some problems. For this reason, we improved the CAVENET, by changing the movement pattern of the vehicles from the straight line to a circle. We also implemented three routing protocols for Ad Hoc networks: Optimized Link State Routing (OLSR) [3], Ad-hoc On-demand Distance Vector (AODV) [20], Dynamic MANET On Demand (DYMO) [4,22], and investigated their performance for VANETs.

The rest of the paper is structured as follows. In Section 2, we discuss the related work for VANETs. In Section 3, we present CAVENET structure and description. In Section 4, we discuss the simulation results. Finally, the conclusions and future work are presented in Section 5.

2. Related work

In general, a simulator should have the following properties.

1. It should be open source, in order to let other users criticize the validity of the model and the implementation.
2. The code should be clear, in order to let others performing the task in 1.
3. The structure should be modular, in order to analyse single pieces of the simulation process.

In the recent years, a lot of simulators for VANETs have been emerging [10]. For example, the IMPORTANT framework has been one of the first attempt to understand the dependence between vehicular traffic and communication performance [2,23]. The authors analyzed the impact of the node mobility on the duration of communication paths. However, the author implemented the code in C, which is difficult to debug and extend without the support of a detailed documentation. Moreover, it seems that their Freeway model is not as realistic as the model we study here.

In [8], the authors present a simulator written in Java, which can generate mobility traces in several formats. The details of the implementation are not open. There are also other powerful traffic simulators, like TranSim [24], which makes use of a cellular automaton for simulating the interaction of vehicles. Unfortunately, the code is not conceived for network protocols simulation, and the software is commercially licensed. Also, SUMO is another powerful traffic simulator, intended for traffic planning and road design optimization. There is an attempt to interface SUMO with ns-2 [21]. However, in our opinion, it is very expensive to understand the SUMO language and also unnecessary, because the communications engineer needs only a parsimonious model, easy to extend and/or modify.

There are many other works which consider the possibility of using ad-hoc and MANET protocols for VANET scenarios. A car taking part in a MANET scenario could establish connections using the public hotspots while driving in the city. Also, the deployment of access points along highways in the near future seems feasible. Thus, it is important to investigate the application of MANET routing protocols for VANETs [6,11,12]. In [6], the authors present only the way of generating the vehicle movement pattern. They did not evaluate the performance of routing protocols. While, in [11], the authors used the simulation in [12] and present the performance evaluation for AODV and OLSR protocols. However, they use the uniform distribution for the generation of the node movement.

3. CAVENET structure and description

The mobility model for VANETs should take into account the following parameters.

• The number of lanes and their directions

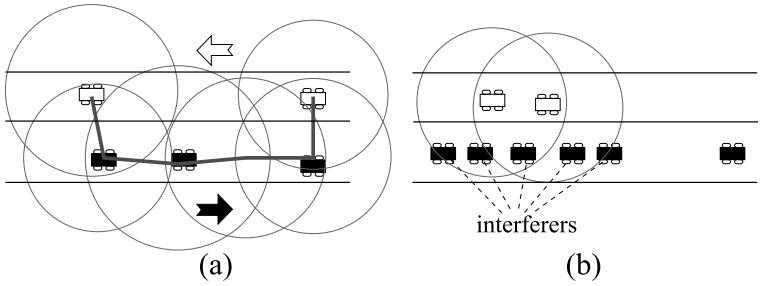

Fig. 1. Impact of multi-lanes: a) on the connectivity; b) on the interference.

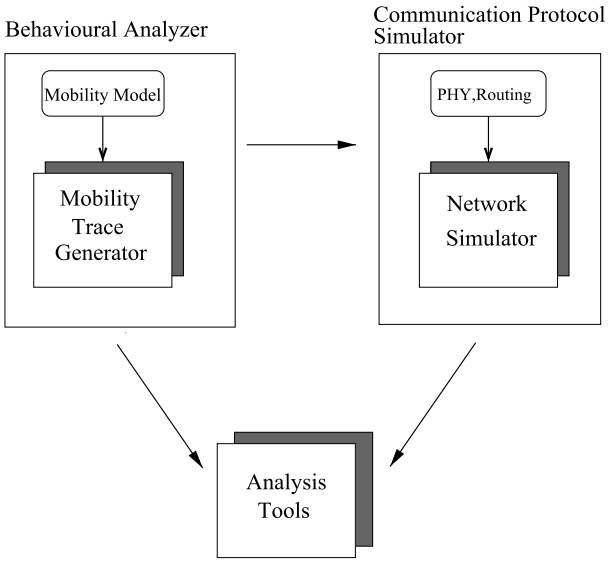

Fig. 2. Structure of CAVENET.

From the point of view of protocol operations, these parameters can affect the connectivity of the network. In particular, connectivity gaps on a lane can be filled by the presence of relay nodes on the other lanes, as shown in Fig. 1-a. On the other hand, the message penetration on a particular lane can be affected by the radio interference on the opposite lane, as in Fig. 1-b.

• The intersection of lanes

This parameter affect the traffic behaviour on the whole lane, because the crosspoint is the bottleneck for the lane.

Here, we take into account only the first parameter. With respect to the aforementioned properties, we propose to divide the simulator into two blocks, as shown in Fig. 2. The first one, which we call Behavioural Analyzer (BA) block, is concerned with the mobility model, and it should take into account the previous parameters in order to produce accurate mobility traces. The second one, which we name Communication Protocol Simulator (CPS), is the protocol simulator, and it is conceived to test the performance of communication protocols given a particular mobility trace. The BA block should be written in a high-level language, easy to understand and easy to extend. For the particular case of CAVENET, the matrix operations are needed. For this reason, we choose MATLAB. The BA block produces movement patterns which are formatted in a textual format compatible with the CPS's language.

Extending the BA block in order to export to other formats is straightforward. The CPS can be one of the many publicly available network simulators, as the well known ns-2 [15]. In principle, the two blocks could also be implemented in two separate machines, in order to speed up the simulation, as in [21].

3.1. Microscopic model

The core of our simulator is 1-dimensional CA model, which has been first studied by Nagel and Schreckenberg (NaS) [19] in a stochastic settings. The CA is a discrete time model of the vehicle traffic. It is governed by three simple rules. However, as for other CAs, these simple rules can well model and reproduce complex real systems. For this reason, the NaS model has gained a lot of attention during the last ten years.

The time is divided in discrete units Δt, so that $t_n = n\Delta t$. There are N vehicles. A lane k of the road at time $t_n, n \in \mathbb{N}$, is represented by a vector $\mathbf{L}_n^k$ of L sites. The lane is assigned an Nx1 velocity vector $\mathbf{v}_n^k = (v_{i,n}^k)_{i=1}^N$, where $v_{i,n} \in \mathbb{N}_{v_{\max}}$ is the velocity of the vehicle at time t_n and position i. If the ith site is occupied by a car, $L_{i,n} = v_{i,n}$. Otherwise, $L_{i,n} = -1$. We use the lane index only when it is explicitly required. Every cell or site of the lane has a length of s meters. By setting $v_{\max} = 135$ km/h and $\Delta t = 1$ s, we obtain $s = 7.5$ m. At every time step, the velocity v is changed according to the following rules.[1]

Deterministic, $p = 0$ **or** $p = 1, \forall i$

- 1. $v_{i,n+1} = \min(v_{i,n} + 1, v_{\max})$
- 2. $v_{i,n+1} = \min(v_{i,n}, L_{i+1,n} - L_{i,n} - 1)$
- 3. $\mathbf{L}_{n+1} = \mathbf{L}_n + \mathbf{v}_{n+1}$

Stochastic

- $2'$. $v_{n+1,i} = \max(0, v_{n,i} - 1)$, with probability p.

The vehicle density is $\rho = N/L$. This simple model can recreate the footprints of real traffic scenarios, such as the $1/f$ noise of the average velocity observed in real traffic. The dynamics of the systems are regulated by three important parameters, p, ρ and L. For example, if $p = 0$ the average velocity is SRD, otherwise the system present LRD.[2]

3.2. Improvement of CAVENET

In the first version of CAVENET, the vehicles were moving in a horizontal line. When a vehicle was at the end of line, in order to continue the simulation we shifted the vehicle at the beginning of line. But,

[1] We assume parallel update only, i.e. the rules are applied in parallel to every vehicle on the lane.

[2] A stochastic process $\{X_n\}_{n=1}^{n=+\infty}$ is SRD if the autocorrelation is summable:

$$\sum_{k=1}^{+\infty} r(k) < +\infty\,,$$

where $r(k) = E[(X_n - \overline{X})(X_{n+1} - \overline{X})]/\sigma^2$. Otherwise, if $r(k)$ is not summable, the process is LRD. This means that very distant samples are not statistically independent, contrary to processes without memory, as the Poisson process which is an SRD process.

this caused a delay and the vehicles at the beginning and at the end of the line could not communicate with each other. For this reason, we improved the CAVENET, by changing the movement pattern of the vehicles from the straight line to a circle. We also implemented three routing protocols: OLSR, AODV, DYMO and investigated their performance for VANETs.

3.2.1. OLSR

The OLSR protocol is a pro-active routing protocol, which builds up a route for data transmission by maintaining a routing table inside every node of the network. The routing table is computed upon the knowledge of topology information, which is exchanged by means of Topology Control (TC) packets.

OLSR makes use of *HELLO* messages to find its one hop neighbours and its two hop neighbours through their responses. The sender can then select its Multi Point Relays (MPR) based on the one hop node which offer the best routes to the two hop nodes. By this way, the amount of control traffic can be reduced. Each node has also an MPR selector set which enumerates nodes that have selected it as an MPR node. OLSR uses TC messages along with MPR forwarding to disseminate neighbour information throughout the network. Host Network Address (HNA) messages are used by OLSR to disseminate network route advertisements in the same way TC messages advertise host routes.

OLSRv2 is currently being developed at IETF. It maintains many of the key features of the original protocol including MPR selection and dissemination. Key differences are the flexibility and modular design using shared components such as packet format packetbb and neighbourhood discovery protocol.

Recently *olsrd* has been equipped with the LQ extension, which is a shortest-path algorithm with the average of the packet error rate as metric. This metric is commonly called ETX, which is defined as $\mathrm{ETX}(i) = 1/(NI(i) \times LQI(i))$. Given a sampling window W, $\mathrm{NI}(i)$ is the packet arrival rate seen by a node on the i-th link during W. Similarly, $\mathrm{LQI}(i)$ is the estimation of the packet arrival rate seen by the neighbour node which uses the i-th link. When the link has a low packet error rate, the ETX metric is higher.

3.2.2. AODV

The AODV is an improvement of DSDV to on-demand scheme. It minimize the broadcast packet by creating route only when needed. Every node in network maintains the route information table and participate in routing table exchange. When source node wants to send data to the destination node, it first initiates route discovery process. In this process, source node broadcasts Route Request (RREQ) packet to its neighbours. Neighbour nodes which receive RREQ forward the packet to its neighbour nodes. This process continues until RREQ reach to the destination or the node who know the path to destination.

When the intermediate nodes receive RREQ, they record in their tables the address of neighbours, thereby establishing a reverse path. When the node which knows the path to destination or destination node itself receive RREQ, it send back Route Reply (RREP) packet to source node. This RREP packet is transmitted by using reverse path. When the source node receives RREP packet, it can know the path to destination node and it stores the discovered path information in its route table. This is the end of route discovery process. Then, AODV performs route maintenance process. In route maintenance process, each node periodically transmits a Hello message to detect link breakage.

3.2.3. DYMO

DYMO is a new reactive (on demand) routing protocol, which is currently developed in the scope of the IETF's MANET working group. DYMO builds upon experience with previous approaches to reactive routing, especially with the routing protocol AODV. It aims at a somewhat simpler design, helping to

reduce the system requirements of participating nodes, and simplifying the protocol implementation. DYMO retains proven mechanisms of previously explored routing protocols like the use of sequence numbers to enforce loop freedom. At the same time, DYMO provides enhanced features, such as covering possible MANET-Internet gateway scenarios and implementing path accumulation.

Besides route information about a requested target, a node will also receive information about all intermediate nodes of a newly discovered path. There is a major difference between DYMO and AODV. AODV only generates route table entries for the destination node and the next hop, while DYMO stores routes for each intermediate hop. To efficiently deal with highly dynamic scenarios, links on known routes may be actively monitored, e.g. by using the MANET Neighbourhood Discovery Protocol or by examining feedback obtained from the data link layer. Detected link failures are made known to the MANET by sending a route error message (RERR) to all nodes in range, informing them of all routes that now became unavailable. Should this RERR in turn invalidate any routes known to these nodes, they will again inform all their neighbours by multicasting a RERR containing the routes concerned, thus effectively flooding information about a link breakage through the MANET.

DYMO was also designed with possible future enhancements in mind. It uses a generic MANET packet and message format and offers ways of dealing with unsupported elements in a sensible way.

3.3. Vehicle model

Every vehicle is a data structure VE_i indexed by its position on the lane. The data structure for the ith vehicle stores: the gap, the velocity, and the current lane position. The relative euclidean position on the lane given by X_i is a unique identifier used for the generation of mobility trace. Moreover, for closed boundaries, i.e. if we suppose circular movement of vehicle on the lane, we check if a shift has taken place. This information will serve to properly generate the trace for ns-2. It is straightforward to arrange all these information in a vector form, what is the preferred form used in MATLAB.

3.4. Lane construction

Instead of using a particular textual language for describing the position of the lanes in the plane, we use a more general approach. Besides its length, every lane is given a lane transformation, which is used in order to set its real aspect on the plane. This information is used at the mobility trace generation stage. The transformation is a simple affine transformation of the vector $\mathbf{X}_i^k = (X_i, Y_i, 1)$, i.e. the coordinate vector of the ith vehicle on the kth road with respect to the relative reference system. For example, for the lane $\mathbf{L}^k$, we have the vehicle structure VE_i^k. This structure contains the vector $\mathbf{X}_i^k$. The real position on the plane is computed as $\widetilde{\mathbf{X}}_i^k = \mathbf{A}(k)\mathbf{X}_i^k$ where $\mathbf{A}(k)$ is the lane transformation matrix associated with the k-th lane, and $\widetilde{\mathbf{X}}_i^k$ is the vector of coordinates in the absolute reference system (i.e. that used for exporting the ns-2 traces). For example, in Fig. 3, the third lane has the following absolute coordinates:

$$\widetilde{\mathbf{X}}_i^3 = \begin{pmatrix} 0 & 1 & \frac{\mathrm{XS}}{2} \\ 1 & 0 & \Delta \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X_i \\ 0 \\ 1 \end{pmatrix},$$

where XS is the length of the simulation area.[3]

[3]The parameter Δ is used to avoid an apparent bug in ns-2, which fires strange errors when the absolute position is 0.

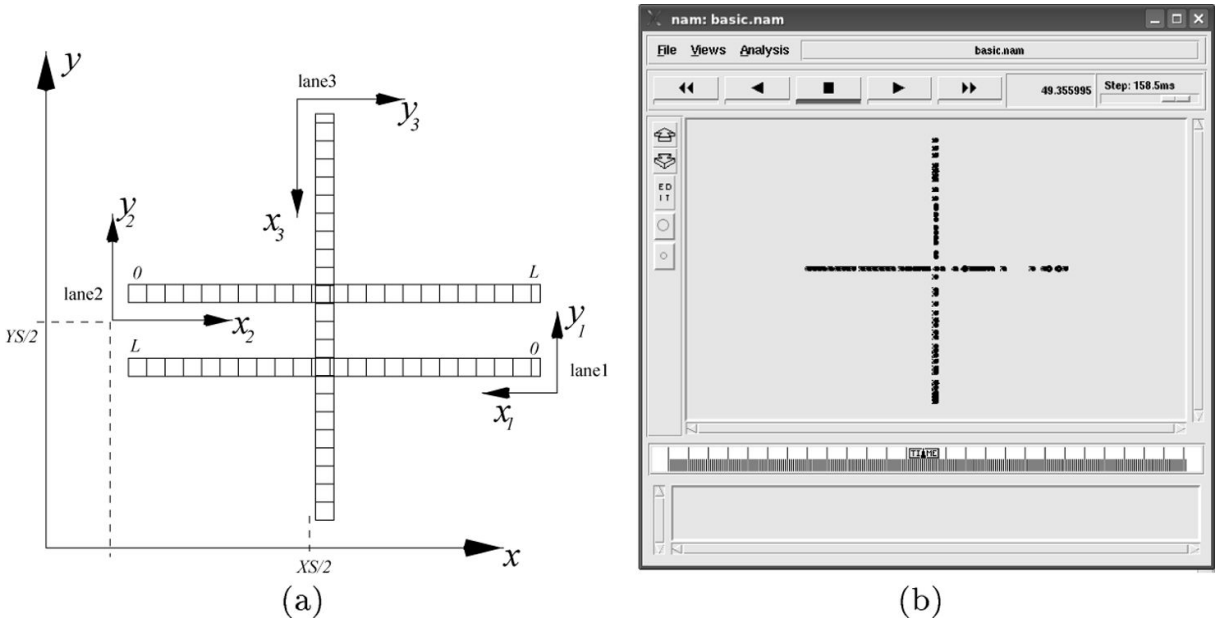

Fig. 3. Lanes construction and ns-2 trace: a) Line construction; b) Excerpt of the generated ns-2 trace for 2 lanes network.

4. Simulation results

4.1. Mobility model validation

We present here some basic simulations for the NaS model by means of CAVENET. Hereinafter, we use as simulation variable the average velocity $\overline{v}(t) = N^{-1}\sum_{i=1}^{N} v_i(t) = N^{-1}\| \mathbf{v(t)} \|_1$ of all cars. CAVENET can analyze and design single and multiple lanes traces. It can also run Monte Carlo simulations. For example, in Fig. 4, we report the results for the so called fundamental diagram, i.e. the flow vs. density diagram. The flow at a particular lane section is defined as $J = \rho\overline{v}$. Each point in the figure is the ensemble average over 20 trials of a simulation trace lasting 500 iterations. Moreover, we can also visualize the space-time plot of the traffic, i.e. the evolution of the velocity for every vehicle along the road as shown in Fig. 5. We obtain the two traffic regimes, namely the laminar regime and the jammed or congested regime, as shown in Fig. 5-a and Fig. 5-b, respectively. We are interested in the the stationary distribution and transient time, which are very important to assess the next stage simulations, i.e. those related to the communication protocol analysis.

4.2. Stationary distribution

Usually, RW-like mobility models used in simulation exhibit the velocity decay problem. That means that the simulation variable slowly decays towards a steady state value as the simulation time proceeds. This is problematic, because we do not know when this transient ends. Consequently, we do not know precisely how to remove the transient values. The root of this phenomenon has to be attributed to the underlying mobility model, which has been assumed random. Every node randomly picks a velocity from a continuous uniformly distributed random variable between $[v_{\min}, v_{\max}]$. The velocity is changed at particular points called waypoints. In this way, the system has an infinite (but countable) number

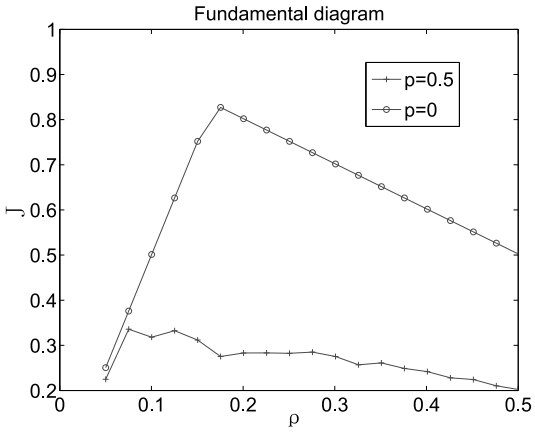

Fig. 4. Traffic flow as a function of ρ and p for $L = 400$.

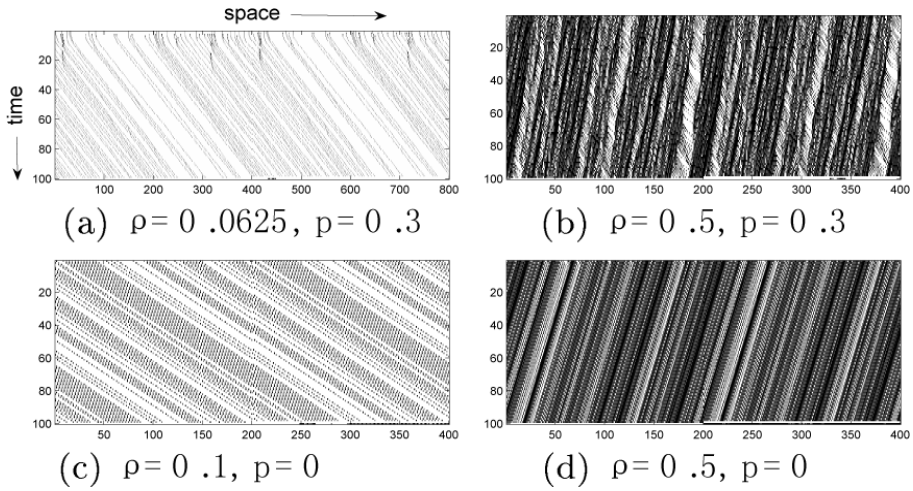

Fig. 5. Space-time plots showing the jam wave in different settings.

of states. The general solution to this problem consists in finding the steady state distribution of the simulation variable and let the system starts with that distribution. This reasoning is also equivalent to consider Palm probability distributions instead of the usual ones [16].

In our case, the system has inherently a finite state space. The automaton could be represented by a discrete-time finite-state Markov chain. We know that a Markov chain with a single class of recurrent states has always a steady state distribution. Moreover, since a Markov chain with finite state space has always at least one recurrent state, we conclude that the steady state distribution exists and is unique. The convergence rate toward this steady-state distribution depends on the eigenstructure of the transition probability matrix of the Markov chain. The problem here is that a Markov chain model is not suitable, because the process can be, in general, LRD, for $0 < p < 1$. Moreover, even in the SRD case, finding the transition probabilities is not easy.

In general, mobility models for vehicular traffic exhibits a phase transition around a particular value of ρ. As we can see in Fig. 5, for $p > 0$, the traffic is composed of jammed regions which travel on the opposite direction of movement. For low densities, these waves die out very quickly, as shown also

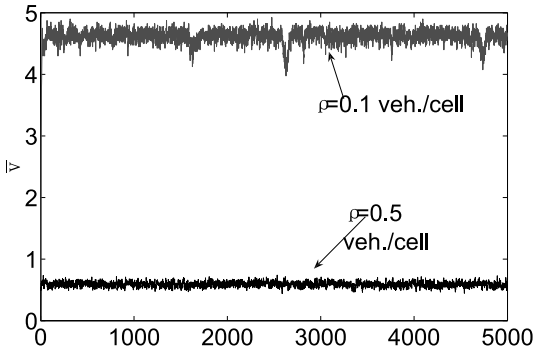

Fig. 6. Sample realizations of $\overline{v}(t)$.

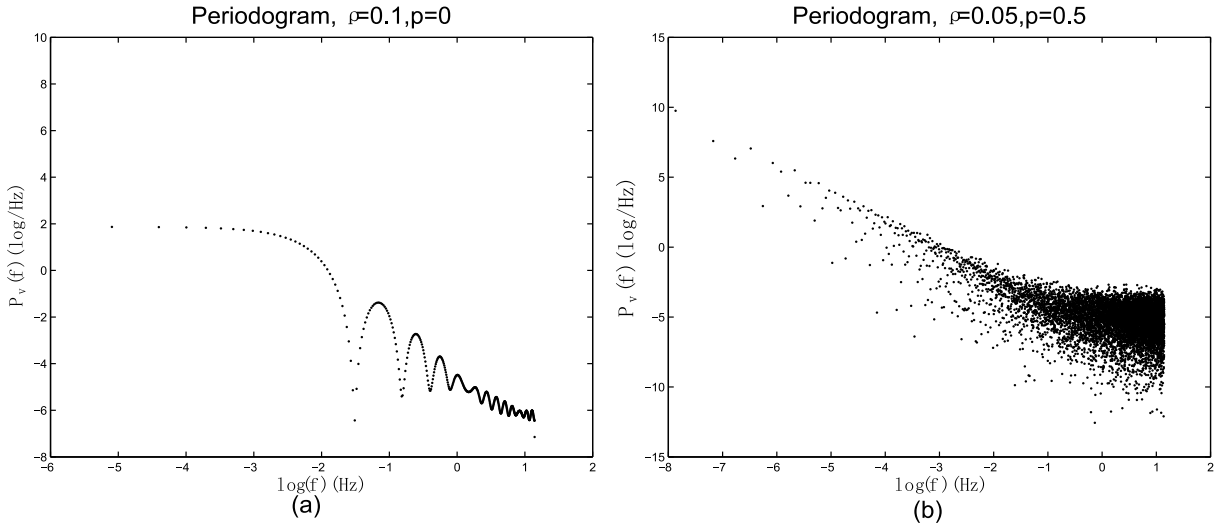

Fig. 7. Deterministic model and stochastic version: a) deterministic model; b) 1/f noise like spectrum form the the stochastic version of the NaS model.

in Fig. 6, but for higher densities there are many interconnected clusters of jammed vehicles. In this case, the steady state is reached very slowly. Therefore, it is important to investigate how many samples should be removed from the staring point in order to sample a process in its stationary regime.

In order to clarify this phenomenon, we measured the transient time τ for $p = 0$, i.e. the deterministic case. In this case, $\overline{v}(t)$ is not LRD. We can show this fact also by plotting the periodogram of $\overline{v}(t)$. In Fig. 7-a, we see that for $f \rightarrow 0$, the periodogram does not diverge. On the other hand, for $p > 0$, in Fig. 7-b, the estimated spectrum diverges at the origin, i.e. the underlying process has the LRD property.

4.3. Routing protocols evaluation

As evaluation metrics, we use the goodput and Packet Delivery Ratio (PDR). The simulation parameters are shown in Table 1. We used one line and 30 nodes for simulations. The simulation time is 100 seconds.

The receiving node is node 0 and the sending nodes are from node 1 to node 8. We prepared each scenario based on nodes ID. The mobility pattern for all scenarios is the same. In order to evaluate the

Table 1
Simulation parameters

Network Simulator	ns-2
Routing Protocol	AODV, OLSR, DYMO
Simulation Time	100 s
Simulation Area	3000 m Circuit
Number of Nodes	30
Traffic Source/Destination	Deterministic
DATA TYPE	CBR
Packets Generation Rate	5 packets/s
Packet Size	512 bytes
MAC Protocol	IEEE802.11 DCF
MAC Rate	2 Mbps
RTS/CTS	None
Transmission Range	250 m
Radio Propagation Models	Two-ray Ground
Hello$_{AODV}$ Interval	1 s
Hello$_{OLSR}$ Interval	1 s
TC$_{OLSR}$ Interval	2 s
Hello$_{DYMO}$ Interval	1 s

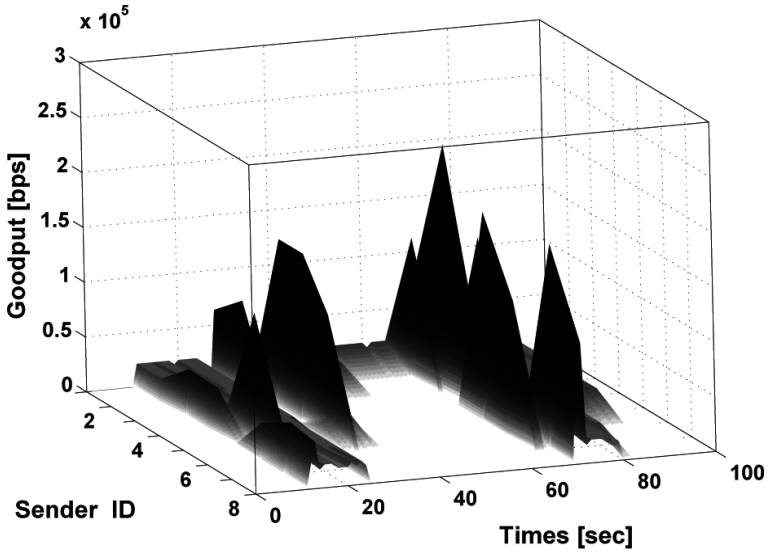

Fig. 8. AODV Goodput.

performance of each protocol, 5 packets per second as a Constant Bit Rate (CBR) traffic were transmitted between 10 seconds and 90 seconds.

The simulations results are shown from Figs 8 to 11. In Fig. 8 is shown the goodput of AODV protocol. The goodput of AODV is about ten times of CBR packet size. This is because after a back-off time all the accumulated data packets are transmitted in the discovered route. If we increase the background traffic, the number of transmitted packets will again increases and the network may be congested. Also, after 60 seconds, in AODV protocol, there is a delay caused by route finding mechanism. Comparing Fig. 8, Figs 9 and 10, we can see that reactive protocols (AODV and DYMO) have better goodput than OLSR. For AODV and DYMO, even the nodes are far from each other they can communicate between 10 seconds to 20 seconds.

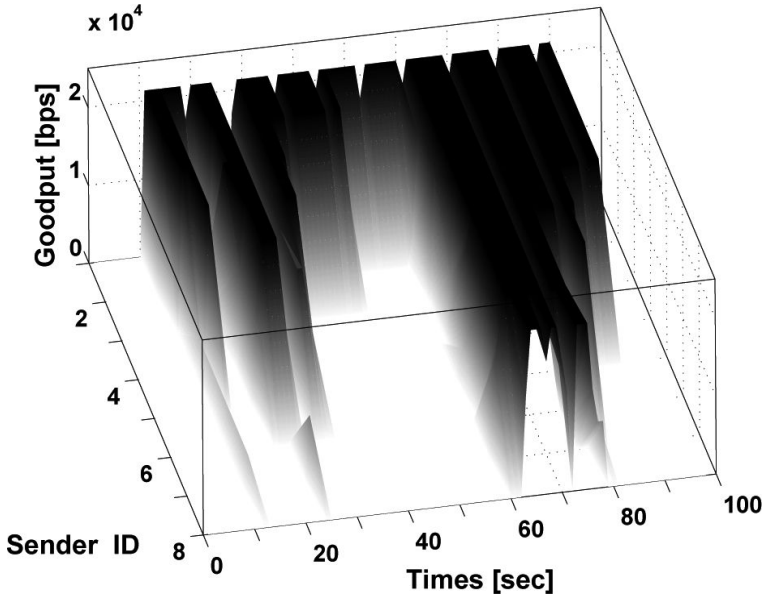

Fig. 9. OLSR Goodput.

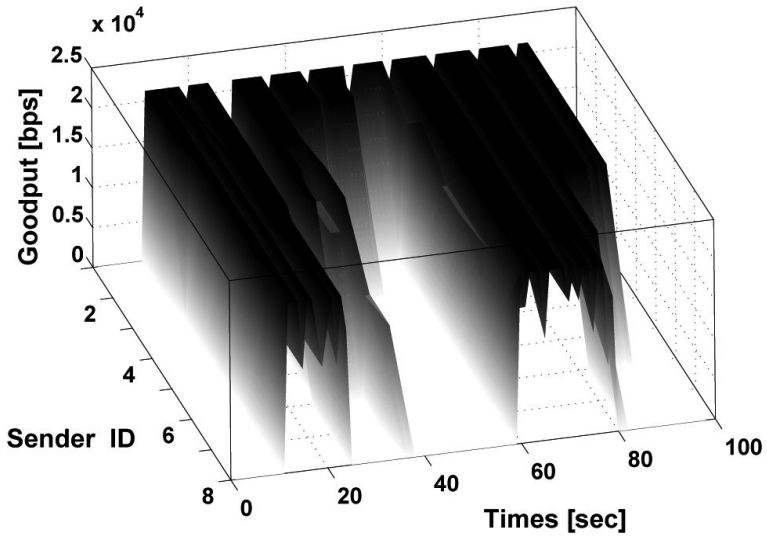

Fig. 10. DYMO Goodput.

In Fig. 11, we show the PDR for three routing protocols. We can see that among three protocols AODV has a better goodput. However, the AODV need more time for searching a new route compared with DYMO. So, the delay of AODV is higher than DYMO. The route searching time of DYMO is almost the same with OLSR protocol. However, DYMO have better goodput than OLSR. Thus, DYMO has a better performance than AODV and OLSR protocols.

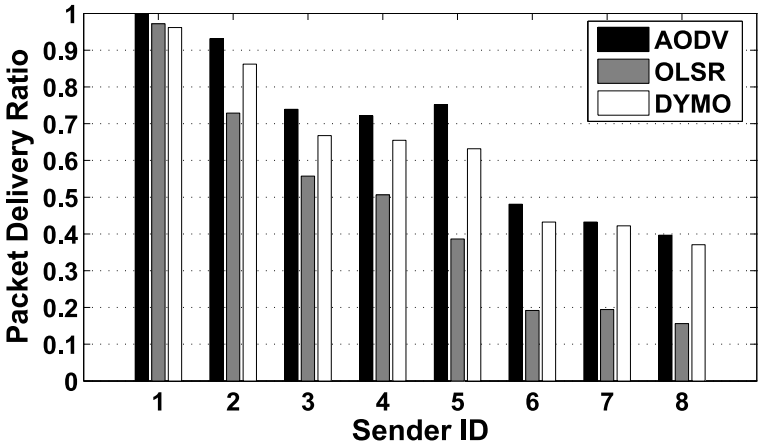

Fig. 11. PDR for AODV, OLSR and DYMO.

5. Conclusions

In this paper, we presented CAVENET, a simple simulator for VANETs. The system is modular and it separates mobility from the protocol simulation. The mobility model code is written in a language at a level as high as possible, in order to give the researcher a quick understanding of the basic properties of his/her model. For this reason, we used MATLAB. The basic structure in CAVENET is the vector representing the configuration of a linear lane. The geometry of the lanes is set by affine transformations which are stored in a text file. In such way, the user does not need to learn a particular file format, as in other traffic simulators. By means of CAVENET, we have shown some fundamental properties of vehicular traffic which should taken into account when performing network protocols simulations.

We improved the CAVENET by changing the movement pattern of the vehicles from the straight line to a circle and implemented three routing protocols: AODV, OLSR and DYMO. We evaluated the performance of these protocols in VANETs and we found that DYMO has better performance.

In this work, we evaluated AODV, OLSR and DYMO considering goodput and PDR metrics. In the future, we would like to consider other parameters such as routing overhead, traffic quantity and topology change. We also plan to extend our work for different radio propagation models and environments [1,7].

Acknowledgment

This work is support by a Grant-in-Aid for scientific research of Japanese Society for the Promotion of Science (JSPS). The authors would like to thank JSPS for the financial support.

References

[1] L. Barolli, Gj. Mino, F. Xhafa, G. De Marco, A. Durresi, A. Koyama, Analysis of Ad-Hoc Networks Connectivity Considering Shadowing Radio Model, *Proc. of MoMM-2009*, Kuala Lumpur, Malaysia, (2009), 464–468.

[2] F. Bai, N. Sadagopan and A. Helmy, IMPORTANT: A Framework to Systematically Analyze the Impact of Mobility on Performance of Routing Protocols for Ad-hoc Networks, *Proc of IEEE INFOCOM-2003*, (2003) 825–835.

[3] T. Clausen and P. Jacquet, Optimized Link State Routing Protocol (OLSR), *IETF RFC 3626*, (October 2003).

[4] I. Chakeres and C. Perkins, Dynamic MANET On-demand (DYMO) Routing, *Internet Draft* (*draft-ietf-manet-dymo-14*), (June 2008).

[5] W.P. Chen, J.C. How and L. Sha, Dynamic Clustering for Acoustic Target Tracking in Wireless Sensor Networks, *IEEE Trans. on Mobile Computing* **3**(3) (2004), 258–271.
[6] G. De Marco, M. Tadauchi and L. Barolli, Description and Analysis of a Toolbox for Vehicular Networks Simulation, *Proc. of IEEE ICPADS/PMAC-2WN-2007* **2** (2007), 1–6.
[7] D. Dhoutaut, A. Régis and F. Spies, Impact of Radio Propagation Models in Vehicular Ad Hoc Networks Simulations, *Proc. VANET-2006*, (2006) 40–49.
[8] M. Fiore, J. Harri, F. Filali and C. Bonnet, Vehicular Mobility Simulation for VANETs, *Proc. of the 40-th Annual Simulation Symposium* (*ANSS-2007*) (2007), 301–309.
[9] J. Goh, and D. Taniar, Mining Frequency Pattern from Mobile Users, *Proceedings of the 8th International Conference on Knowledge-Based Intelligent Information and Engineering Systems* (*KES-2004*), Part III, Lecture Notes in Computer Science 3215 Springer (2004), 795–801.
[10] J. Harri, F. Filali and C. Bonnet, Mobility Models for Vehicular Ad Hoc Networks: A Survey and Taxonomy, *IEEE Communications Surveys & Tutorials* **11**(4) (2009), 19–41.
[11] J. Haerri, F. Filali and C. Bonnet, Performance Comparison of AODV and OLSR in VANETs Urban Environments under Realistic Mobility Patterns, *Proc. of 5th IFIP Mediterranean Ad-Hoc Networking Workshop* (*Med-Hoc-Net-2006*), (2006) Lipari, Italy.
[12] J. Haemi, M. Fiore, F. Filali and C. Bonnet, A Realistic Mobility Simulator for Vehicular Ad Hoc Networks, *EURECOM Technical Report*, (2007) Available at: http://www.eurecom.fr/util/publidownload.en.htm?id=1811.
[13] A.M. Hanashi, I. Awan and M. Woodward, Performance Evaluation with Different Mobility Models for Dynamic Probabilistic Flooding in MANETs, *Mobile Information Systems* **5**(1) (2009), 65–80.
[14] M. Ikeda, L. Barolli, G. De Marco, T. Yang, A. Durresi and F. Xhafa, Tools for Performance Assessment of OLSR Protocol, *Mobile Information Systems* **5**(2) (2009), 165–176.
[15] Information Science Institute (ISI), *Network Simulator Version 2* (*NS-2*), http://www.isi.edu/nsnam.
[16] J.Y. Le Boudec and M. Vojnovic, The Random Trip Model: Stability, Stationary Regime, and Perfect Simulation, *IEEE/ACM Transactions on Networking* **14**(6) (2006), 1153–1166.
[17] S.S. Manvi, M.S. Kakkasageri and Jeremy Pitt, Multiagent Based Information Dissemination in Vehicular Ad Hoc Networks, *Mobile Information Systems* **5**(4) (2009), 363–389.
[18] E. Natsheh and T.C. Wan, Adaptive and Fuzzy Approaches for Nodes Affinity Management in Wireless Ad-hoc Networks, *Mobile Information Systems* **4**(4) (2009), 273–295.
[19] K. Nagel and M. Schreckenberg, A Cellular Automaton Model for Freeway Traffic, *Journal of Physics I France* **2** (December 1992), 2221–2229.
[20] C. Perkins, E. Belding-Royer and S. Das, Ad hoc On-Demand Distance Vector (AODV) Routing, *IETF RFC 3561* (*Experimental*), (July 2003).
[21] M. Piorkowski, M. Raya, A.L. Lugo, M. Grossglauser and J.P. Hubaux, Joint Traffic and Network Simulator for VANETs, *Proc. of Mobile and Information Communication Systems Conference* (*MICS-2006*), (October 2006) Available on line at: http://www.mics.ch/.
[22] C. Sommer and F. Dressler, The DYMO Routing Protocol in VANET Scenarios, *Proc. of 66-th IEEE Vehicular Technology Conference* (*VTC2007-Fall*), (2007), 16–20.
[23] N. Sadagopan, F. Bai, B. Krishnamachari and A. Helmy, PATHS: Analysis of Path Duration Statistics and Their Impact on Reactive MANET Routing Protocols, *MobiHoc-2003: Proc. of the 4-th ACM International Symposium on Mobile Ad Hoc Networking & Computing* (2003), 245–256.
[24] L. Smith, R. Beckan, R. Anson, K. Nagel and M. Williams, TRANSIMS: Transportation Analysis and Simulation System, *Proc. of the 5-th National Transportation Planning Methods Applications Conference*, (1995), LA-UR 95-1664.
[25] D. Taniar and J. Goh, On Mining Movement Pattern from Mobile Users *International Journal of Distributed Sensor Networks* **3**(1) (2007), 69–86.
[26] K. Xuan, G. Zhao, D. Taniar and B. Srinivasan, Continuous Range Search Query Processing in Mobile Navigation, *Proc. of IEEE ICPADS-2008 International Conference* (2008), 361–368.
[27] T. Yang, G. De Marco, M. Ikeda and L. Barolli, Impact of Radio Randomness on Performances of Lattice Wireless Sensors Networks Based on Event-reliability Concept, *International Journal of Mobile Information Systems* **2**(4) (2006), 211–227.
[28] J. Yoon, M. Liu and B. Noble, A General Framework to Construct Stationary Mobility Models for the Simulation of Mobile Networks, *IEEE Transactions on Mobile Computing* **5**(7) (2006), 860–871.

Evjola Spaho received her B.S and M.S degrees at Faculty of Information Technology, Polytechnic University of Tirana (PUT) in 2008 and 2010, respectively. Presently, she is a Ph.D Student at Graduate School of Engineering, Fukuoka Institute of Technology (FIT), Japan. Her research interests include P2P networks, vehicular networks, ad-hoc networks and robot control.

Leonard Barolli received B.S and Ph.D degrees from Tirana University and Yamagata University in 1989 and 1997, respectively. From April 1997 to March 1999, he was a JSPS Post Doctor Fellow Researcher at Department of Electrical and Information Engineering, Yamagata University. From April 1999 to March 2002, he worked as a Research Associate at the Department of Public Policy and Social Studies, Yamagata University. From April 2002 to March 2003, he was an Assistant Professor at Department of Computer Science, Saitama Institute of Technology (SIT). From April 2003 to March 2005, he was an Associate Professor and presently is a Full Professor, at Department of Information and Communication Engineering, Fukuoka Institute of Technology (FIT). Dr. Barolli has published about 300 papers in referred Journals, Books and International Conference proceedings. He was an Editor of the IPSJ Journal and has served as a Guest Editor for many International Journals. Dr. Barolli has been a PC Member of many International Conferences and was the PC Chair of IEEE AINA-2004 and IEEE ICPADS-2005. He was General Co-Chair of IEEE AINA-2006, AINA-2008, AINA-2010, CISIS-2009 and CISIS-2010, Workshops Chair of iiWAS-2006/MoMM-2006 and iiWAS-2007/MoMM-2007, Workshop Co-Chair of ARES-2007, ARES-2008, IEEE AINA-2007 and ICPP-2009. Dr. Barolli is the Steering Committee Chair of CISIS and BWCCA International Conferences and is serving as Steering Committee Co-Chair of IEEE AINA, NBiS and 3PGCIC International Conferences. He is organizers of many International Workshops. Dr. Barolli has won many Awards for his scientific work and has received many research funds. He got the "Doctor Honoris Causa" Award from Polytechnic University of Tirana in 2009. His research interests include network traffic control, fuzzy control, genetic algorithms, agent-based systems, ad-hoc networks and sensor networks. He is a member of SOFT, IPSJ, and IEEE.

Gjergji Mino was graduated in Electronics at Tirana Polytechnic University, Albania in 1995. After the graduation, he was working for different telecommunication companies in Albania. From 1997, he worked for different companies such as Digital Corporation, Compaq, Computer Technology Solutions and Peripheral Computer Support in US. Presently, he is a Ph.D Student at Graduate School of Engineering, Fukuoka Institute of Technology (FIT), Japan. His research interests include wireless networks, wireless cellular networks, vehicular networks, CAC, handover and fuzzy logic.

Fatos Xhafa holds a PhD in Computer Science (1998) from the Department of Languages and Informatics Systems, Polytechnic University of Catalonia, Barcelona, Spain. He was graduated in Mathematics from the Faculty of Natural Sciences (FNS), University of Tirana (UT), Albania, in 1988. During 1989–1993 he was an Assistant Professor at the Department of Applied Mathematics of the FNS, UT (Albania). He joined the Polytechnic University of Catalonia as a Visiting Professor at the Applied Mathematics Department I, and later he entered the Department of Languages and Informatics Systems (LSI) as a PhD Student and in 1996 as an Assistant Professor. Currently, he is an Associate Professor at the Department of Languages and Informatics Systems and member of the ALBCOM Research Group He is also a collaborator of the Open University of Catalonia (Barcelona, Spain). His current research interests include parallel and distributed algorithms, combinatorial optimization and approximation, meta-heuristics for complex problems, distributed programming, Grid and P2P computing. His research is supported by several research projects from Spain, European Union and NSF/USA. He collaborates with different international research groups. He has published in leading international journals and conferences and has served in the Organizing Committees of many conferences and workshops. He served as the Organizing Chair of ARES-2008, PC Chair of CISIS-2008, Workshops Co-chair of CISIS-2008, General Co-chair of HIS-2008, General Co-Chair of CISIS-2010 and PC Co-Chair of IEEE AIAN-2010 conferences. He is also member of editorial board of several international journals.

Vladi Kolici received his B.S and M.S degrees in Telecommunication Engineering from Polytechnic University of Tirana (PUT) in 1997 and 2005, respectively. He obtained his Ph.D from PUT in May 2009. From 1997 to 2004, he was a Research Associate and from 2005 to present he is a Lecturer at Department of Electronics and Telecommunications, Faculty of Information Technology, PUT. He is teaching several courses in the areas of wireless and mobile networking, P2P systems and quality of services. Dr. Kolici has published several papers in International and National Conference Proceedings in the areas of P2P and Ad-Hoc networks. Dr. Kolici received the Best Application Paper Award at the 6th International Conference on Advances in Mobile Computing and Multimedia (MoMM-2008) in 2008, Linz, Austria. His research interests include P2P networks, wireless and mobile networks, and high speed networks.

Rozeta Miho received her B.S and M.S degrees in Electronic and Telecommunication Engineering from Polytechnic University of Tirana (PUT), Albania, in 1985 and 1989, respectively. She obtained her Ph.D from PUT in November 1995. From 1985 to 1995, she was a lecturer, from 1996 to 2001 Assistant Professor, from 2001 to 2007 Associate Professor, and presently she is a Full Professor of PUT. From May 2009, she is the Dean of Faculty of Information Technology, PUT. She is teaching several courses in the areas of telecommunication networks, optical communications and optical fibre networks. Prof. Miho has published several papers in International and National Conference Proceedings in the areas of optical WDM networks, P2P and Ad-Hoc networks. She is the co-author of the Best Application Paper Award at the 6th International Conference on Advances in Mobile Computing and Multimedia (MoMM-2008) in 2008, Linz, Austria. Her research interests include P2P networks, optical networks, wireless networks and high speed networks.

Mobile Information Systems 6 (2010) 259–280
DOI 10.3233/MIS-2010-0103
IOS Press

Robust video communication over an urban VANET

N. Qadri, M. Altaf, M. Fleury* and M. Ghanbari
School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, UK

Abstract. Video communication within a Vehicular Ad Hoc Network (VANET) has the potential to be of considerable benefit in an urban emergency, as it allows emergency vehicles approaching the scene to better understand the nature of the emergency. However, the lack of centralized routing and network resource management within a VANET is an impediment to video streaming. To overcome these problems the paper pioneers source-coding techniques for VANET video streaming. The paper firstly investigates two practical multiple-path schemes, Video Redundancy Coding (VRC) and the H.264/AVC codec's redundant frames. The VRC scheme is reinforced by gradual decoder refresh to improve the delivered video quality. Evaluation shows that multiple-path 'redundant frames' achieves acceptable video quality at some destinations, whereas VRC is insufficient. The paper also demonstrates a third source coding scheme, single-path streaming with Flexible Macroblock Ordering, which is also capable of delivery of reasonable quality video. Therefore, video communication between vehicles is indeed shown to be feasible in an urban emergency if the suitable source coding techniques are selected.

Keywords: Error resilience, IEEE 802.11p, multiple path delivery, redundant frames, VANET, video communication

1. Introduction

This paper considers how to support robust video communication across multi-hop networks between vehicles when an urban emergency occurs. Real-time video communication allows early responders approaching an incident [28] to better understand the nature of the problem at the scene of an emergency but the lack of centralized routing and network resource management is challenging. Crash scenes, views of fleeing vehicles or burning buildings are some illustrative applications, while there is also now a strategic incentive [28] to provide coverage during a more serious, general emergency. In all these scenarios, it is the other personnel in the emergency vehicle or passengers in the vehicle that view the arriving video stream and not the driver.

Vehicular Ad Hoc Networks (VANETs) bring several advantages to video streaming within an ad hoc network. Battery power is no longer a problem if built-in transceivers are employed, implying that larger buffers (with passive and active energy consumption) can now serve to absorb any latency arising from multi-hop routing. We consider urban VANETs. Within a city, because of traffic congestion, high speeds do not generally arise. Therefore, connections are on average longer and Doppler effects are limited. Vehicle motion is indeed restricted by the road geometry but compared to a highway VANET vehicle motion is no longer linear.

*Corresponding author. Tel.: +44 1206 872817; Fax: +44 1206 872900; E-mail: fleum@essex.ac.uk.

We examine three alternative video practical source-coding schemes for emergency video streaming, with one of the schemes applied in two different ways. The source-coding techniques applied exist in the context of IP networking [39] but, as far as the authors are aware, they have not been applied elsewhere in the way described within a VANET context. The first scheme examined is a variant of Multiple Description Coding (MDC) [36] in which two or more versions or descriptions of the same video stream are sent over different, preferably disjoint, routes across a network. Either description can serve to reconstruct the video but enhanced quality is produced by combining both descriptions. If adverse conditions occur on one of the paths then the packetized encoded bitstream from the other path can compensate. Video Redundancy Coding (VRC) [40] is the simplified MDC scheme employed by us that in the event of packet loss does not require additional decoder reconciliation between the two descriptions. Additionally, the VRC scheme was also trialed using distributed intra-coded macroblocks, that is H.264/AVC (Advanced Video Coding)'s [39] Gradual Decoder Refresh (GDR) [30], to avoid the reliance on prior reference frames.

This paper also proposes a second MDC scheme, employing H.264/AVC redundant frames, which when combined with multiple-path video transfer will result in higher-quality delivered video at a cost in higher data traffic. However, this cost may well be justified in an emergency. Redundant frames [32,42] (or strictly redundant slices[1] making up a frame) are coarsely quantized frames that can avoid sudden drops in quality marked by freeze frame effects if a complete frame (or slice) is lost. Again assuming even and odd frames are sent separately in two streams, then redundant frames are predicted from previous frames in the same stream but do not act as a reference to later frames.

In a third scheme, our paper proposes Flexible Macroblock Ordering (FMO) [38] with Checkerboard FMO pattern for single-path video stream transfer as an alternative to multiple-path methods. Error resilience [31] is applied at a source encoder to counter potential packet loss. FMO is an error-resilient technique newly included in the H.264/AVC codec that is suitable for error-prone channels. In good channel conditions, the overhead from sending the FMO mapping is a disadvantage but this is unlikely to be a problem when multi-hop routing occurs. Through source-encoder-independent error concealment at the decoder, FMO can aid the reconstruction of frames that have lost some of their constituent packets.

To the best of the authors' knowledge, though investigation of the concept of video streaming within a VANET has occurred, source-coding techniques have not been applied to any extent to VANETs. Some of the literature that exists on this subject so far in comparison to our work is examined in Section 2. The University of California, Los Angles (UCLA) research group under the leadership of Mario Gerla has produced a range of creative ideas on VANETs, for example [12,25,28]. However, their focus is on wireless aspects and at the 2009 Wireless Days Conference they have confirmed their interest in using a variant of our FMO scheme within live vehicle convoys as a security measure. We are flattered by their interest. Finally, we should add that another apparently unique feature of this work is that we group the emergency vehicles into a multicast group to receive the video. But we also use the other vehicles in the vicinity whether emergency or not to relay the video. In this way, the efficiency of the transfer is greatly improved. This latter feature is described in Section 3 but firstly this paper reviews other investigations of video over VANETs.

[1] A slice is headed by a decoder resynchronisation marker and may include reversible Variable Length Decoding, aimed at countering propagation of errors arising from the sequential dependencies of entropic encoding. Consequently, a slice is a self-contained decoding unit.

2. Related research

Earlier work on video communication over highway (not urban) VANETs [15] considered the problem of triggering remote video sources in the event of forward traffic congestion. The main problems in triggering [28] are how to reduce the number of messages reaching the remote camera(s) and how to reduce the latency in reaching those cameras (by reducing the number of hops), which is principally an issue of protocol design. In an emergency scenario as opposed to obtaining forward views of traffic congestion, it may be that video sources can be locally generated. Then an entirely different problem arises: how best to deal with heavy packet losses in the harsh urban environment. Video quality is strongly influenced by the impact of packet loss. Because successive video frames are broadly similar (except at scene cuts and changes of camera shots), to increase coding efficiency only the difference between successive frames is encoded. Consequently, at the frame level, removing temporal redundancy introduces a dependency on previously transmitted data that implies lost packets from reference frames will have an impact on future frames until a successful delivery of the next spatially-encoded anchor frame, when the decoder can be reset.

In [15], multiple vehicle video sources were modeled traveling on a 4–5 lane highway in Atlanta. Video was collected by sending from a car approaching a destination a request trigger to a camera on a remote vehicle passing that destination region. Video transport back to the requestor was by a store-carry-and-forward sub-system, though the method was not detailed in [15]. The main analysis in [15] was of delay characteristics, presumably because on a highway there should be sufficient time for the approaching vehicle to take evasive action if the forward view shows congestion or an accident.

Research in [25], extending the work in [15], simulated a two-ray wireless propagation model and imposed an application-layer Forward Error Control (FEC)-based solution through network coding. Though network coding of FEC and in particular rateless error coding is an effective means of limiting the impact of packet erasures upon streamed video, it depends on action by intervening nodes. When these nodes are not possible destinations and consequently may not be expected to make special provision for video data, then network coding is not feasible. Therefore, our paper considers alternative video protection methods that act in an end-to-end fashion, without the need for processing by intervening vehicles. Error resilience (in our schemes) is able to complement physical layer FEC, whereas when applying higher layer forms of FEC, it is better to do so in such a way that the channel code acts as an inner code to the PHY coding.

The feasibility of H.264/AVC video communication between two vehicles with IEEE 802.11b transceivers in a live setting was examined in [5]. With speeds between the two vehicles on average 15 mph (6.71 m/s) in a city setting (in Japan) it was reported that 'link availability' was 97.78%, as opposed to on a highway at an average speed of 55 mps (24.59 m/s) it was only 33.98%. Average SNR was somewhat worse in the urban setting, 19.14 dB, as opposed to 22.49 dB on the highway. Relative video quality was good (around 30–35 dB Peak Signal to Noise Ratio (PSNR)) and better in the city scenario with (slow-scan) rates ranging from 15–20 Hz and for Quarter Common Interchange Format (QCIF) (176×144 pixels/frame) to CIF-resolution (352×288 pixels/frame). The test clip was the well-known 'Foreman' clip at QCIF resolution (employed in this paper also) with medium coding complexity, though 'Paris' with less temporal complexity was employed at the larger CIF resolution. The study [5] established that, for vehicles traveling in proximity to each other, video exchange is entirely feasible albeit at slow-scan rates and resolution depending on coding complexity (which is a reflection of spatial activity (within frames) and temporal activity (between frames)). Of course, these results do not necessarily translate to multi-hop video transfer.

In a general context, the dissemination of multimedia information is a subject of active investigation within mobile systems. For example, in [21] progressive transmission of multi-resolution documents occurs so that the viewer can first view the relevance of the information before continuing with full transfer. To reduce the impact of low bandwidth capacity and delay over multi-hop connections, caching of data in the vicinity of an ad hoc node [6] is a promising approach. To support such systems in an ad hoc network, it is important that the routing protocols are optimized, for example [16] by tuning the route request flooding mechanism.

3. VANET system

3.1. Emergency application

Our system usage is captured in Fig. 1, showing an encircled crash scene. In such an urban emergency, it is envisaged that a scene is captured by one vehicle (the first emergency vehicle at the scene) that acts as the video source. Emergency vehicles now commonly carry video cameras, which in the case of the police act as a source of evidence in traffic offences. Therefore, there may be no need to capture the scene from cameras mounted on roadside masts, though these can be triggered locally as alternative video sources. Thus, either the scene can be captured manually by an emergency worker operating a vehicle mounted video camera, as already occurs when traffic police gather evidence or it can be captured through vehicle to roadside communication (vrc) or a roadside camera could be controlled by an emergency vehicle through remote communication. As how the video is captured is not central to this paper, we refer the reader to discussion of the feasibility of vrc such as in [8].

The video is distributed via WLAN-enabled vehicles to a multicast group of patrol cars, fire engines, ambulances or the like, acting as early responders to a crisis. If the multicast group consisted only of emergency vehicles (assuming less than ten responders for any one incident) then the ad hoc network size or density would be an impediment to communication. By routing the video stream over other intermediate car wireless transceivers, even though these cars are not destinations for the video stream (only the emergency vehicles are) multi-hop packet routing is more effective. Notice that though the destinations form a multicast group within the larger set of VANET-enabled vehicles, to improve robustness in all scenarios considered point-to-point communication is employed rather than a multicast protocol.

One emergency vehicle, acting as the video source, transfers the captured video by sending individual copies to each destination forming a virtual multicast group. When an MDC schemes is used each stream is split over two paths, refer to Fig. 1. Separate threads of control are able to generate these descriptions, possibly utilizing multi-core processors. It is assumed that available destinations, corresponding to other emergency vehicles in a group, are known through another emergency channel. In tests, six destinations were employed and it was found that, depending on choice of scheme, reasonable to good quality video was possible, though not for every destinations. Nevertheless, sufficient emergency vehicles would have a view of the emergency to allow preparations to be made as vehicles approached the scene.

3.2. VANET communication

Direct inter-vehicle communication can be an aid both to passenger comfort and to road safety [43]. Compared to a cellular network, a VANET may be toll free, avoids the delay in setting up a long communication circuit, and on a highway will operate where there are coverage gaps in a cellular

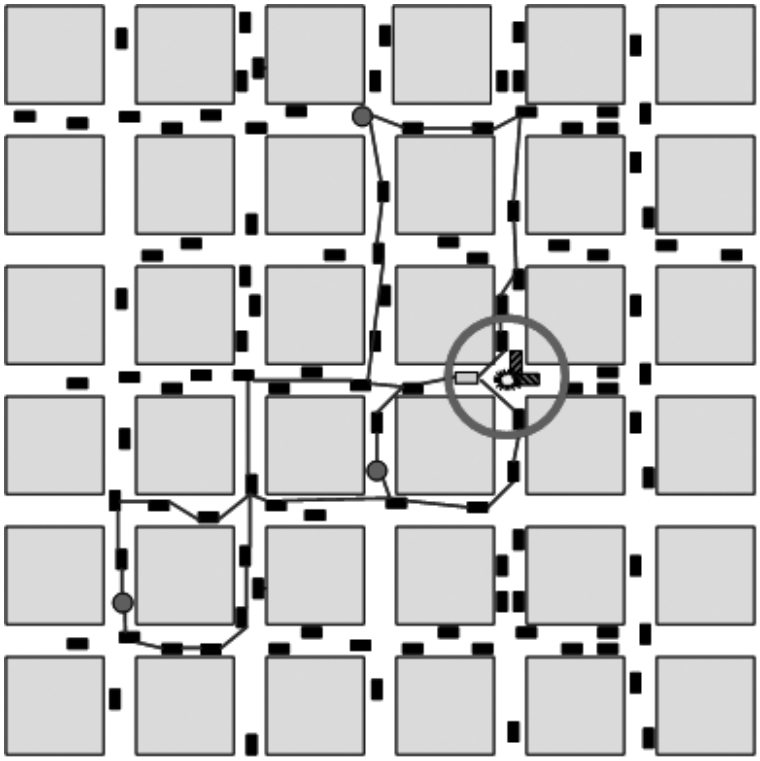

Fig. 1. VANET operating in city blocks with collision (encircled) videoed by light colored patrol car. Black rectangles are other vehicles acting as relays. Small circles are other emergency vehicles. Connecting lines show possible multi-paths for transmission of video.

network. There are strong pressures pushing car manufacturers towards equipping cars with WLAN capability, if they have not already done so. The IEEE 802.11p standard [17] will take advantage of 75 MHz of spectrum allocated both in Europe and the USA in the 5.9 GHz range with 10 MHz channels operating at up to 27 Mbps depending on modulation mode. The increased safety [3] that may arise from wireless provision is under active investigation. As well as safety alerts through wayside access points, the possibility of advertising localized services provides an additional commercial incentive to wireless take-up. It is thought that early adoption will result in around 20% of WLAN-enabled cars [4] in the near term. Therefore, at least 20% of the available cars in a city are likely to be available as relays to aid in video communication in an emergency.

If a VANET is to present an alternative to private cellular radio such as the Terrestrial Trunk Radio (TETRA) system [9] then it should provide similar services. Video communication over TETRA was explored in [7] and TETRA-2 was provided with extra bandwidth in support of multimedia communication. As an example, HW Communications Ltd. recently presented T-Serv for slow-scan video communication over TETRA, with in-vehicle video communication through IEEE 802.15.1 (Bluetooth). Compared to TETRA's cellular system, a VANET system can additionally make use of vehicles other than the emergency vehicles themselves, thus increasing coverage. (TETRA has an ad hoc mode but this is obviously confined to emergency vehicles, thus restricting the size of the ad hoc network.) Ad hoc radio is also potentially not limited by the urban 'canyons' caused by high buildings (if a base antenna is employed).

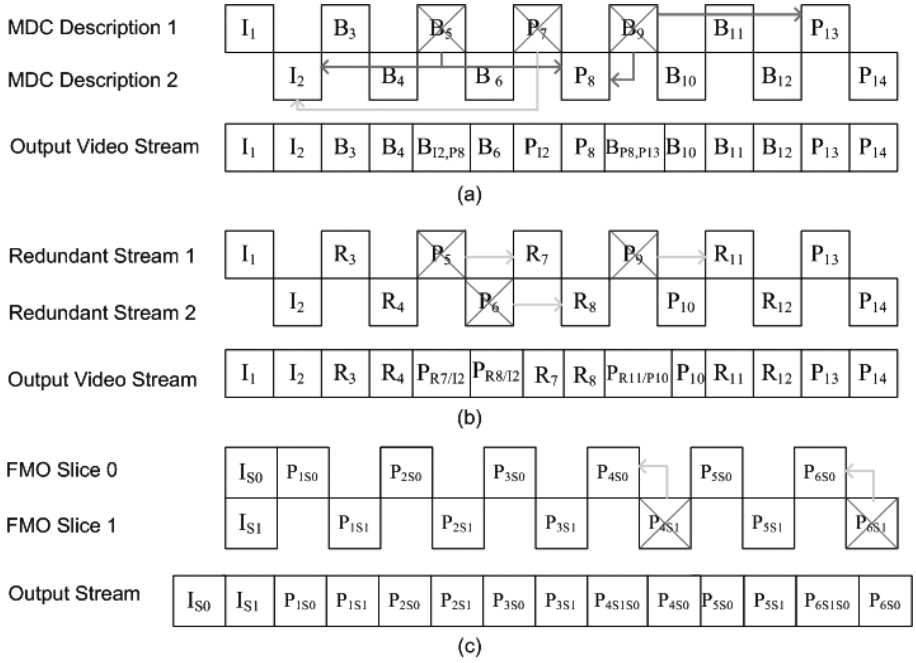

Fig. 2. Different path diversity schemes: a) VRC with odd and even descriptions b) Two streams with redundant frames, c) FMO slice replacement scheme.

If the source and destination are both assumed to be emergency vehicles, as in the scheme presented in this paper, then, when the video is routed via non-emergency vehicles, its confidentiality can be preserved through a stream cipher or alternatively through selective encryption of compressed video [44] to reduce the computation load of full encryption. (For example, only motion vectors in the compressed steam need be encrypted, as without these it is difficult to reconstruct a video). Another approach to security [22], which requires the cooperation of intermediate nodes to perform network coding, takes advantage of disjoint paths. However, though multiple paths may be available in a VANET, it is difficult and unnecessary to ensure they are node disjoint. A further approach [12] is to allow partial disclosure of some data after a time limit has expired. However, though this scheme may reduce dissemination latency, it is not clear that it allows streaming, as it requires an all-or-nothing transform, i.e. receipt of all data, before the data can be reconstructed. Partial disclosure of content is also permitted.

3.3. Video transfer schemes

Figure 2 illustrates the schemes tested in this paper. The frame numbers indicate the raw video frame from which a coded frame is constructed. Frames are decoded with motion compensation from reference frames in the same stream. By separately decoding from each stream, the problem of MDC decoder complexity is avoided. Figure 2a shows a number of frames have been dropped (marked by crossing out). Lost frames in one description can be reconstructed by reference to other correctly received packets in either description, with arrowed lines in Fig. 2a indicating the reconstruction route. For example, B_5 has been lost in description 1 and is reconstructed from either I_2 and P_8. The final row of frames in Fig. 2a shows the frame display sequence arising after substitution of reconstructed frames. Not shown in Fig. 2a is the VRC variant in which IDR-frames are no longer included in the sequence (refer to Section 4).

In Fig. 2b, showing the redundant frame scheme, the absence of B-frames allows use of the computationally efficient H.264/AVC Baseline profile. Redundant frames are sent in each stream, at a potential cost in latency but a potential gain in delivered video quality. There is only one initial IDR-frame in each sequence, which can be replaced by an intra-coded redundant frame. All other redundant frames are normally encoded in inter-mode (with the same reference as the frame that they back up) with normally a coarser quantization setting than the frame they back-up. Again example packet losses are shown and the result of reconstructing the sequences appears in the final row of Fig. 2b. For example, P_5 has been lost and reconstructed from R_7 but if R_7did not survive it could also have been reconstructed from I_2. Figure 2c shows the FMO method of error resilience in which each frame has been split into two slice groups. That is each frame is divided into two slices and sent in different packets. The packets are multiplexed onto a single stream with slice 0's packet preceding slice 1's packet (though the order is not important). When a packet bearing one of a frame's slices is lost then the corresponding slice is normally employed to reconstruct it through the non-normative error concealment procedure. If both slices are lost then previous frame replacement is reverted to. The final row now shows the frame receiving order from a single stream.

4. Proposed schemes

The first scheme proposed for use in the VANET emergency is VRC [40], which avoids the need for decoder reconciliation in the event of packet loss. This is because two *independent* streams are formed from separately encoded odd and even frame sequences. Some lack of coding efficiency occurs as the motion between frames in any one description is likely to be greater than if the frames were coded in their original order. Consequently, the residual or difference data, which is actually coded by predictive coding, has a larger dynamic range requiring more bits to code. By insertion of IDR-frames in both descriptions (streams), the descriptions can be resynchronized even if one of the IDR frames is lost, at a cost in increased data redundancy compared to sending a single set of IDR-frames in a single stream. The macroblocks of IDR frames are completely spatially- (intra-) coded without removal of temporal redundancy and, therefore, do not reference any other frames. Consequently, they act as anchor frames for predictively- (inter-) coded subsequent frames. (In a hybrid video codec, each frame is split into macroblocks for processing purposes. Further details can be found in a textbook such as [13].)

In compensation, for the reduction in coding efficiency resulting from employing IDR-frames, bi-predictive B-frames are included within each Group of Pictures (GOP) within a VRC description to improve efficiency through multiple references (with a 10% bit-rate reduction in H.264/AVC [29]). (Notice that in the H.264/Advanced Video Codec (AVC) [41], IDR frames prevent predictive reference across GOP boundaries, whereas H.264/AVC I-frames do not (unlike their usage in earlier codecs)). Either an IDR- or an I-frame is inserted after every 12 or 15 frames making up a GOP.

IDR frames cause periodic increases in the data rate and consequently introduce additional buffering delay. Therefore, as an alternative form of VRC, we also distributed an equivalent number of intra-coded macroblocks [30] to those contained in the IDR-frames across the two VRC descriptions. Though another function of IDR-frame insertion is to provide a random access facility (supporting video player functions), this is unlikely to be required in an emergency scenario. In this variant of VRC apart from initial IDR-frames in each description only predictively-coded P-frames occur. This results in some loss of coding efficiency but improves computational efficiency (as the need to conduct more than one

predictive search for B-frames is no longer needed)[2]. Along with Constant Bit Rate (CBR) encoding, an all P-frame sequence reduces delay for real-time applications. The risk of continued error propagation from the loss of any one IDR-frame-bearing packet is also reduced by distributing intra-macroblocks across all frames, H.264/AVC's GDR [39].

In general, MDC is computationally complex and requires specialist codecs [36], because synchronization between encoder and decoder is necessary to reduce motion estimation error drift. In a two stream MDC scheme, synchronization normally requires a third decoder in addition to the decoders that produce the reduced quality streams from single descriptions [14]. VRC is a simplified version of MDC with only two decoders, the output of which is interleaved before display. As mentioned previously, VRC normally requires the inclusion of IDR frames to allow decoder reset in the event of packet loss. If the packets happen to belong to an IDR frame then an IDR frame in the other sequence can serve as an anchor.

To avoid the need to send IDR frames, in Multiple State Video Coding (MSVC) [1] lost frames in one description are reconstructed from temporally adjacent frames in the other description. In this solution, all frames apart from the first IDR frame in each description are P-frames and reconstruction may also occur with the aid of past and future P-frames. However, reconstruction with P-frames from a different description reintroduces the risk of picture drift from the lack of synchronization between an encoder and decoder. For that reason MSVC is not tested in this paper, though MSVC can be credited as the basis of several later practical MDC schemes.

To overcome picture drift, redundant frames intended for error resilience in H.264/AVC [32,42], can serve to better reconstruct P-frames received in error. Though redundant frames were originally intended for Internet video streaming, applying redundant frames to multiple path streaming over general ad hoc network video was independently investigated in [26,33]. However, in [26] redundant pictures in one stream were encoded based on primary frames (frames for which redundant representations are generated) in a second stream, which requires modification of the operation of the H.264 codec. Another multiple path version [33] combines slicing with redundant data. In this version, a frame is split between alternate slices (formed from macroblock rows). Each slice is either a primary slice or a redundant slice for the matching primary slice in the other description. The need to generate this alternating pattern of slice types in each description prevents independent stream generation. That is, it is no longer possible to generate each description or decode it within its own control thread. Therefore, in this paper we apply redundant frames in a more direct manner that does not involve the need for a customized codec operation but does allow independently generated descriptions.

As also introduced in Section 1, FMO [19,38] is a promising form of video error resilience, which we use for single-path video transfer, adopting the type one checkerboard FMO pattern. By default in H.264/AVC, each frame forms a single slice group and macroblocks within that group are decoded in raster scan order. However, within a frame up to eight slice groups are possible. There are also seven different types of mapping between macroblock and slice group. Because type six supports arbitrary slice group mappings, its overhead is the greatest, as in addition to the slice group header a mapping must also be transmitted to the decoder (in a parameter set). Types three to five allow the size of each group to evolve over time, though macroblocks within a group remain geometrically contiguous. Only type one allows the assignment of geometrically dispersed macroblocks within a frame to form a slice group. The assignment is made through a mathematically function that in the two slice case results in a checkerboard

[2]It may also be possible to employ intra-coded macroblocks within B-frames in H.264/AVC, because unlike MPEG-4, this is now supported in H.264/AVC.

pattern. Therefore, the overhead is lighter for this type. To reduce overhead, it is also preferable to choose the option in H.264/AVC that prevents reference outside the slice group, though at some cost in coding efficiency. A detailed analysis of overhead, which depends on encoder configuration, can be found in [19].

Significantly, the type one FMO checkerboard pattern is the only H.264 predefined mapping function that supports error concealment by interpolation of data from adjacent macroblocks in order to reconstruct missing macroblocks (if one of the two checkerboard slice packets were to be lost). Error concealment in H.264/AVC is a non-normative feature [35] in which the motion vectors of correctly received slices are computed if the average motion activity is sufficient (more than a quarter pixel). The recommendation [35] gives details of which motion vector to select to give the smoothest block transition. It is also possible to select the intra-coded frame method of spatial interpolation. In our FMO experiments, though experience favors a motion vector-based method, we employed both methods and selected the superior result in terms of average PSNR across the video sequence. In a live situation, it is possible to choose the method that best reduces 'blockiness' at macroblock boundaries. Notice that in non-FMO experiments, previous frame replacement was employed at the decoder to reconstruct a frame, as this is the normal form of error concealment for comparison purposes in such tests.

5. Simulation model

Simulation is the main tool for research on VANETs [34], because it is difficult to find an analytical solution due to the large number of variables involved such as vehicle density, speeds, and mobility patterns. It is also difficult to conduct repeated live experiments.

5.1. Simulating IEEE 802.11p

The Global Mobile System Simulator (GloMoSim) [45] simulation library was employed to generate our results. GloMoSim was developed based on a layered approach similar to the OSI seven-layer network architecture. Total simulation time was 900 s, with the emergency video distribution starting after 100 s. We employed IP framing with UDP transport, as TCP transport can introduce unbounded delay, which is not suitable for delay-intolerant video streaming. We simulated a multi-path variety of the Ad Hoc On demand Distance Vector (AODV) protocol but without strict enforcement of disjoint paths (either path or node disjoint) for the grid of Fig. 1. This allowed two paths to be selected for the MDC schemes. Though we are aware of considerable research in the field of multi-path protocol design, e.g. see [23] for multi-path video transport with a random waypoint mobility model, the intention of the current paper is to concentrate on video transport aspects. The BonnMotion mobility generator (http://web.informatik.uni-bonn.de/IV/Mitarbeiter/dewaal/BonnMotion/ accessed Sept. 2009) was chosen. Though BonnMotion does not model driver behaviour in the way that a mobility model such as VanetMobiSim [10] does, we considered it sufficient for generic simulations mainly intended to check the application's behavior. Besides, the driving behavior of emergency vehicles will be quite different from normal drivers.

A two-ray propagation model with an omni-directional antenna height of 1.5 m at receiver and transmitter was selected for which the reflection coefficient was -0.7, which is the same as that of asphalt on tarmaced roads. The plane earth path loss exponent was set to 4.0, with the direct path exponent set for free space propagation (2.0). As in IEEE 802.11p, transmission was at 5.9 GHz with a bandwidth of 10 MHz. The transmission power was 33 dBm (2W). The receiver power threshold was set to -93 dBm, a normal value. Lastly, IEEE 802.11p's robust Binary Phase Shift Keying (BPSK)

Table 1
Default simulation settings for Manhattan grid mobility model

Parameter	Value
Terrain dimension	1000×1000 m^2
No. of vehicles	100
Size of multicast group	6
Number of x-, y- blocks	10, 10
Turn probability	0.5
Speed change probability	0.2
Minimum speed	0.5 m/s
Average speed	10.0 m/s (22 mph)
Speed standard deviation[1]	0.2 m/s
Speed update distance	10 m
Pause probability[2]	0
Transmission range	150 m
Routing protocol	AODV
Wireless technology	IEEE 802.11p
Channel model	Two-ray

[1]For normally distributed speeds.[2] If no change of speed.

modulation mode was simulated, introducing a packet length dependency through Bit Error Rate (BER) modeling in a Additive Gaussian White Noise (AGWN) channel. Accordingly, the data-rate was set to 3 Mbps.

5.2. Urban mobility model

The essential features of an urban scenario are captured by mobility models. Two approaches are possible: either the detailed microcellular approach [34] or generic models [2]. The microcellular approach has the advantage that it includes the effect of obstacles such as lane closures, uphill gradients, and potholes. Though the generic models lack the detail of the microcellular approach, these models do allow systematic investigation and easily interpretable results. For the purposes of assessing how video streams can be effectively transferred, we have used a generic model that captures the essential feature of an urban scenario, the restricted mobility patterns imposed by the presence of city blocks.

In [2], two generic models relevant to vehicular mobility are described, namely Freeway Mobility and Manhattan Mobility. The Freeway model limits vehicles to 1-D motion in either direction. Vehicles are tied to one of several lanes; the speed is dependent on a vehicle's previous speed; and in the 'car-following' restriction, a following vehicle cannot exceed the speed of a preceding vehicle to avoid approaching within a safety distance. The Manhattan model, an extension of the Freeway model, restricts the number of lanes in either direction to just one, but introduces a turning probability to give greater mobility. Both Freeway and Manhattan are related in that they should result in high spatial and temporal dependency.

Default simulation settings for the Manhattan grid mobility model are given in Table 1, while individual simulations varied from the given defaults. In many urban settings it is likely that wireless-enabled cars would be restricted to average speeds of around 10 m/s by congestion and traffic regulation, though vehicles responding to an emergency may go at faster speeds. The city block dimensions are chiefly related to the wireless range. Consequently, it is the relationship between range and block dimension that is important rather than the absolute settings.

5.3. Video configuration

The reference Foreman video clip was encoded at QCIF resolution with 4:2:0 sampling. Foreman, intended for judging communication between mobile devices, exhibits the typical features of a hand-held camera and, because of camera pans, exhibits high to medium coding complexity. Each frame was generally coded as a single slice and encapsulated in an H.264/AVC Network Abstraction Layer unit (NALU) [39] before being placed in a single packet. The combination of RTP/UDP/IP headers results in a further 40 B of overhead. IDR-frames, however, were split into two slices, which reduces the peak data rate. If one of the IDR-frame packets arrives before another [37] partial decoding can still take place while the other packet arrives. The encoder was set to output in CBR mode, with initial quantization parameter of 32. The frame rate of the video stream was set at a slow scan rate of 15 Hz to avoid injecting too great a data-rate into the network. Consequently, for each stream in the MDC schemes the data rate was approximately 60 kbps.

For VRC, the skip frame(s) facility of H.264/AVC allowed the creation of even and odd frame sequences. For each sequence when using IDR frames, the Main profile of H.264 allowed B-frames to be included. The GOP size was 15 frames with the usual repeating pattern of two B- and one P-frame until the next IDR-frame. In the Main Profile, Context-Adaptive Binary Arithmetic Coding (CABAC) results in a 9–14% bit saving at a small cost in computational complexity [24]. In the VRC variant and the redundant frame scheme, B-frames were no longer used. Intra-refresh macroblocks now provide the coding anchor points previously provided by IDR-frames. Intra-coded macroblocks were randomly selected (from a Uniform distribution) of H.264 macroblocks and were embedded with P-frames to the equivalent number needed for a QCIF IDR-frame. That is seven macroblocks per frame results in 105 macroblocks within a GOP, as opposed to 99 8×8 macroblocks for a single IDR-frame. Again the ability to generate random intra-coded macroblocks is a facility of H.264/AVC (H.264 JM Ref. Software, http://iphome.hhi.de/suehring/tml/download/, accessed Sept. 2009).

For FMO experiments, the Baseline Profile of H.264/AVC was selected with a GOP structure of IPP... In this Profile, intended for mobile devices and consequently with a smaller code footprint, Context Adaptive Variable Length Codes (dynamic Huffman entropic coding) is employed for simplicity (rather than CABAC), with some reduction in latency.

6. Evaluation

6.1. Preliminary tests

Figure 3 is a comparison between the luminance PSNR resulting from different error-resilient techniques upon Foreman, as the packet loss rate was varied. 100 simulation runs with different starting seeds were averaged to ensure convergence for Fig. 3. Errors in Fig. 3 followed a Uniform probability distribution function. Each frame was coded as a single slice, unless otherwise stated. Further detail of the error resilience methods compared is available in [31]. Notice that we have tested the methods separately for clarity, whereas a combination of methods in an error resilience strategy depending on channel conditions is possible, though there is a cost in extra overhead.

At around zero error packet loss-rate, FMO results in a somewhat lower video quality than omitting error resilience, because of its overhead, resulting in a lower coding efficiency. Separating into three independently-coded slices ('Slices' in Fig. 3), rather than one slice, is seen to be a little more effective at lower loss rates than in other loss regimes, as the risk of packet error is lower for shorter packet lengths.

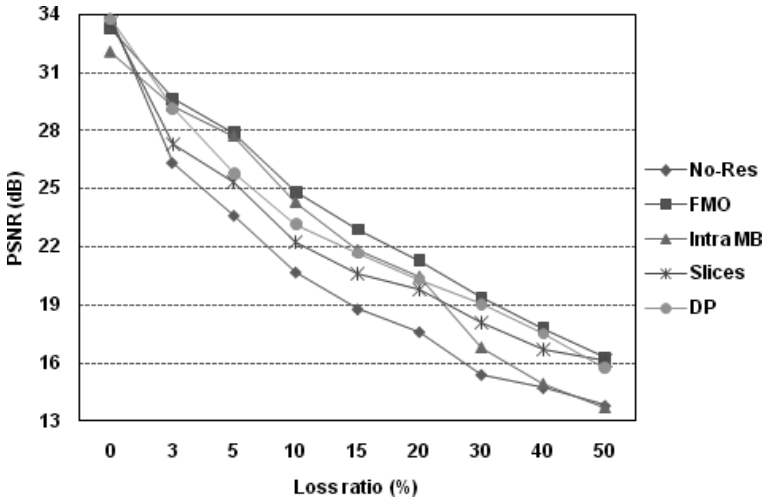

Fig. 3. Comparison between several H.264/AVC error resiliency methods and no resilience (No-Res) communication with Uniform bit errors, for Foreman QCIF sequence.

Data-partitioning ('DP' in Fig. 3 – separating configuration data and motion vectors, intra-coded, and inter-coded data into respectively A, B, and C NALUs) allows a frame to be effectively reconstructed even if the inter-coded macroblocks are missing, provided motion vectors in partition A are protected (which was simulated by assuming strong FEC protection of A NALUs). This technique is most useful at low loss rates. Insertion of Intra-coded Macroblocks ('Intra MB' in Fig. 3) allows temporal error propagation to be gradually arrested if some of a previous reference frame's macroblocks have been lost. In the tests, each row of macroblocks in rotation on a frame-by-frame basis was intra-coded. Intra-coded macroblocks in the tests was most helpful at medium loss rates. In the error conditions simulated, at medium to higher loss rates checkerboard FMO was the most effective method, though delivered video quality can no longer be considered fair for Foreman at a 10% loss rate and higher. However, users normally accept quality at 25 to 30 dB if it is in a mobile application. As the vertical axis in Fig. 3 is effectively logarithmic, a gain of less than one dB will still make a difference at the boundary between a sequence being viewable or not viewable, though at PSNRs below 25 dB that gain is of little importance. Therefore, the main conclusion from Fig. 3 is that FMO is superior to other built-in error resilience methods of H.264/AVC except at very low error rates.

We also compared a selection of alternative FMO patterns, though for reasons of brevity and because it is not central to the main theme of this paper, the results are not plotted herein. The H.264/AVC Checkerboard pattern was superior to other built-in FMO patterns at higher loss rates, when coding QCIF sequences with two slices. At lower loss rates, checkerboard FMO is somewhat weaker but the quality is anyway good with whatever FMO pattern. FMO patterns tested were: a selection of foreground in one slice group and the remainder in another; row interleaving; raster scan ordering with two groups; and selection of columns or part columns. Therefore, the Checkerboard pattern should normally be selected for FMO, especially as it can be used for up to eight slice groups in H.264/AVC and is not confined to a two-slice scheme.

6.2. Packet loss experiments

Table 2 gives the packet size features of the protection schemes in the experiments. Packet sizes grow with the inclusion of intra-coded macroblocks and, because IDR-frames are split into at least two packets,

Table 2
Protection schemes according to packet payload size

Scheme	Size range (B)	Characteristic
VRC with IDR frames	70 B to 885 B	10% over 500 B
VRC with Intra-refresh	183 B to 966 B	50% over 450 B
Redundant frames multiple path	260 B to 780 B	10% over 500 B
FMO single path	39 B to 493 B	55% below 125 B

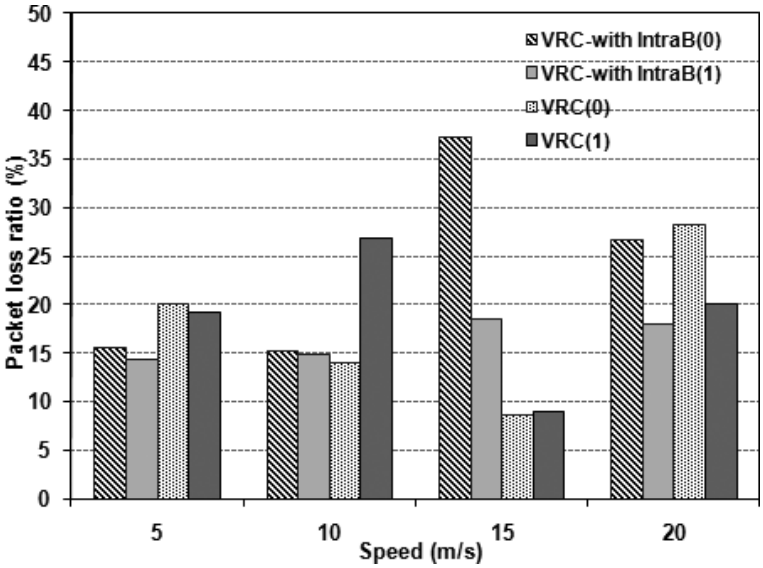

Fig. 4. Packet loss ratio for VRC for description (0) and description (1) versus vehicle speed with Table 1's settings, with IDR-frames and with intra-refresh macroblocks (marked IntraB).

larger packet sizes arise from VRC with intra-refresh than for VRC with IDR-frames. The smaller packet sizes for FMO are the result of splitting every frame into two slices, with a slice per NALU packet.

We firstly make some general remarks about the relationship between packet loss, vehicle speed and resulting video quality. As shown in Table 1, the number of emergency vehicle destinations is taken to be six, with one other video source vehicle, and the default total number of nodes is 100. At a given speed, if vehicles are on average in proximity to each other for sufficient time for packet transfer then one can expect less packet loss to occur. However, a vehicle must also avoid travelling at too slow a speed, as then that vehicle may not have sufficient time to approach another vehicle to affect a further transfer. In general, our experiments showed packet loss rates to be difficult to predict and as Section 6.3 demonstrates, an unreliable indicator of video quality when different protection schemes are compared. There now follows experimental results using the three source-coding schemes.

6.2.1. Packet loss under VRC

Figure 4 shows that packet losses under multiple-path VRC whether with IDR-frames or with Intra-refresh are generally higher than 10%. A level below about 10% is normally required for reasonable delivered video quality *unless* error protection is provided. There is only one speed (15 m/s) at which packet loss is within bounds for the classical VRC scheme. Figure 5 shows the result of altering the number of vehicles available for multi-hop relay. There is a trend towards reduction of packet losses to an acceptable level as the network size is increased, because there are greater opportunities for packet

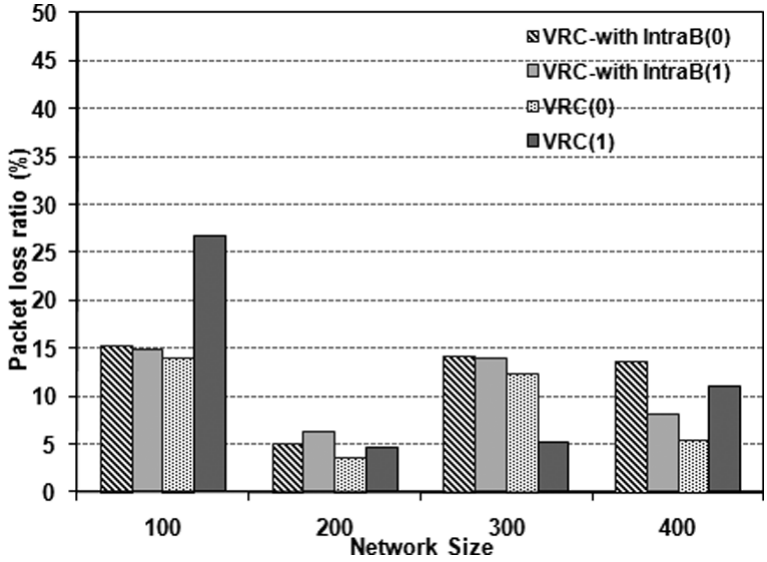

Fig. 5. Packet loss ratio for VRC for description (0) and description (1) versus network size with Table 1's settings, with IDR-frames and with intra-refresh macroblocks (marked IntraB).

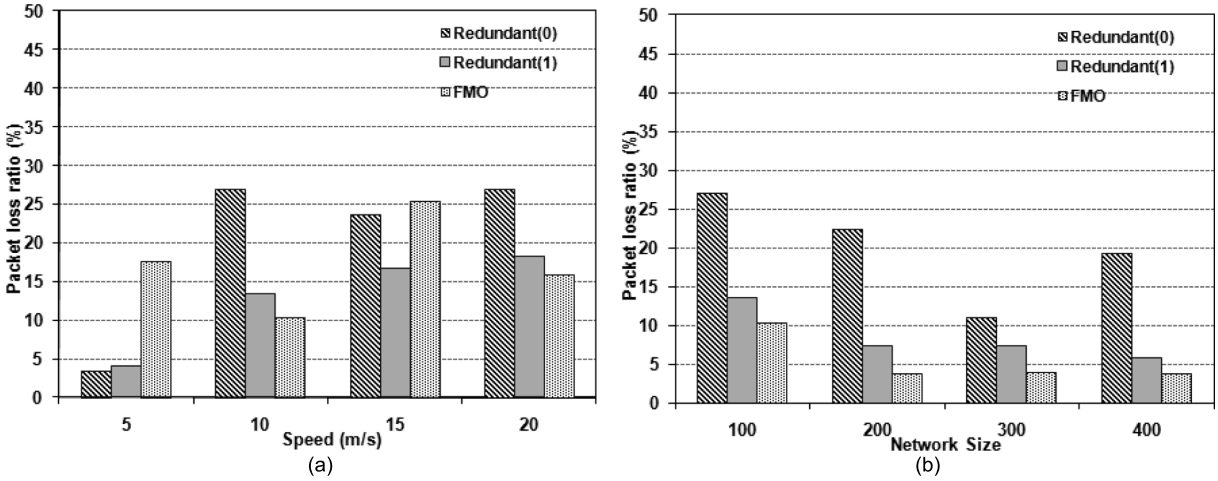

Fig. 6. Packet loss ratio for redundant frames and FMO with Table 1's settings with variation of a) vehicle speed and b) network size.

transfer to occur. In the experiment for IDR-frame VRC, each path might compensate for packet losses in the other path or description. Perhaps, because of the more regular data rate, this trend is not strongly apparent in the Intra-refresh version of VRC streaming. However, the main conclusion that can be drawn from these tests is that VRC in the form used herein is not satisfactory for video transport.

6.2.2. Packet loss under Redundant Frames or FMO

Figure 6 presents multiple-path video transfer packet loss when employing redundant frames. The packet loss levels with variation of speed or network size are generally higher than for VRC. However, packet loss rates are especially not a reliable guide to the likely video quality in this case, as, if redundant

Table 3
Packet losses by protection scheme for the selected destination node

	Packet loss ratio (%)			
	VRC (Intra-Refresh)	VRC (IDR-frames)	Redundant frames	FMO
Network size				
100	6	14	12	8
200	1	2	15	3
300	5	4	3	4
400	4	6	10	2
Speed				
5	2	3	1	15
10	6	14	12	8
15	18	6	10	25
20	8	9	15	13

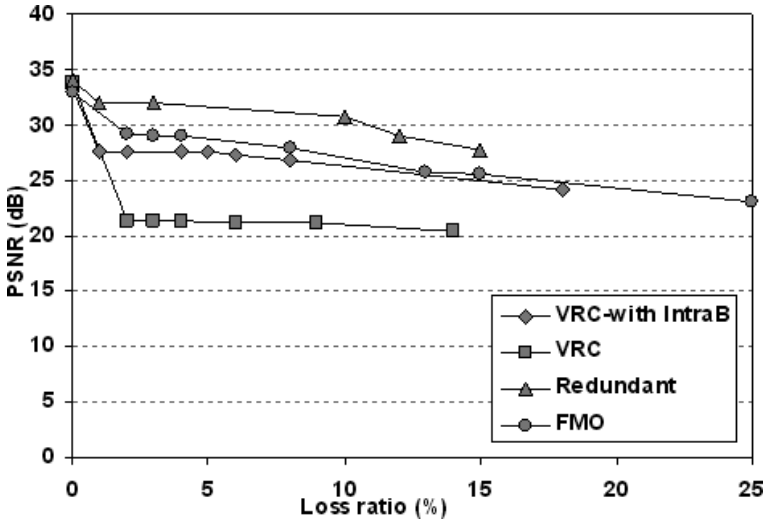

Fig. 7. Video quality by protection scheme for the selected node according to the packet loss ratios of Table 3. VRC = VRC with IDR frames, Redundant = Multiple path with redundant frames, VRC with IntraB = VRC with Intra-refresh macroblocks

frame packets are dropped, the video quality remains the same. Of course, a mixture of redundant and active frame packet loss will actually occur. Turning to single-path FMO error resilience, from Fig. 6 also, there is generally a much reduced level of packet loss for larger network sizes at the default speed, but FMO packet loss is sensitive to vehicle speed. Therefore, the main conclusions are that redundant frames lead to larger packet losses, though this may not necessarily result in worse video quality, whereas single-path FMO suffers less packet loss.

6.3. Video performance

From the six destination nodes, a 'median' node was selected in terms of packet loss, rather than one experiencing very high or very low packet loss. Table 3 reports the packet loss statistics for the chosen node. There is a decreasing trend in packet losses with network size, but no clear trends are apparent in terms of vehicle speed.

Fig. 8. Sample equivalent frames from the protection schemes. IDR = VRC with IDR frames, Rand = VRC with randomised Intra-refresh, Red = Redundant frames with multiple path

6.3.1. Video quality

Figure 7 compares resulting delivered video quality (PSNR) at the default settings. Recall that for PSNR the vertical scale is logarithmic. Employing redundant frames, despite the comparatively high packet loss ratios, results in good video quality. On average, with the packet loss ratios found from the simulation, sufficient redundant frames survive to repair the composite video frame sequence. FMO allows single stream video transfer to compete with multiple-path transfer but video quality is only fair in some settings tested. Including Intra-refresh macroblocks does improve the PSNR of VCR but not sufficiently to compete with the redundant frame scheme. Therefore, the main conclusion is that both redundant frames and FMO achieve reasonable video quality in the given scenario.

For illustrative purposes only, a sample frame in the Foreman sequence is shown in Fig. 8 for each of the schemes. In the two VRC schemes, there is considerable distortion around the face as previous frame error concealment has been affected by motion of the face between the frames. FMO results in less distortion but the cream-slated background shows some distortions.

6.3.2. Video latency and control overhead

Moreover, in broad terms from Fig. 9, average end-to-end delay (for all vehicles) is longer in duration for the VRC schemes. Generally, across all schemes packet delay times are significant and do require buffering. For other applications, e.g. a Web click and view service, there would be a perceptible start-up delay between selecting the video and starting to view the stream. For the emergency scenario, the destination emergency vehicles are unaware of the start time of the stream at the source and, therefore, would not notice the small delay. However, the time available to view the scene as a vehicle approached

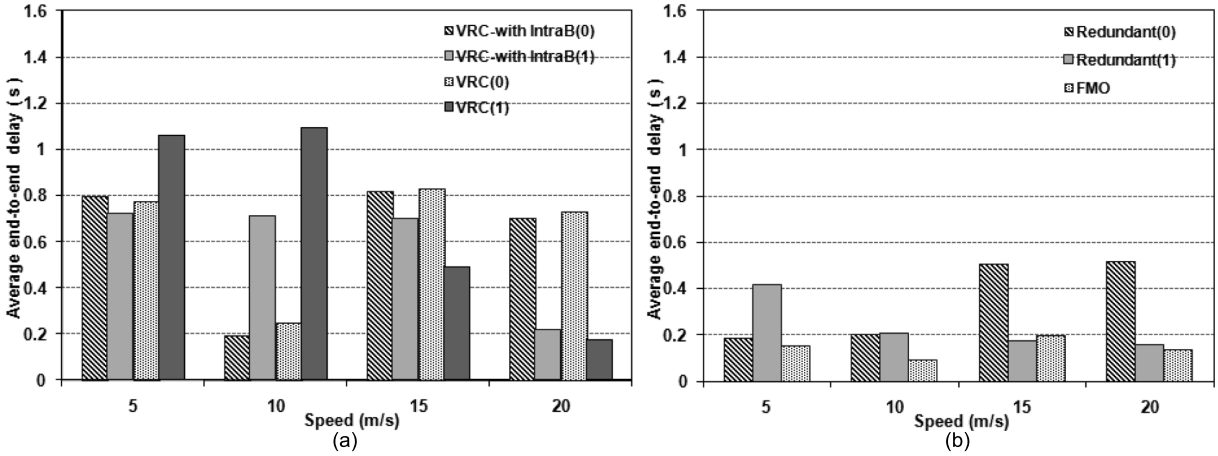

Fig. 9. End-to-end packet delay according to vehicle speed for a) VRC schemes. b) non-VRC schemes.

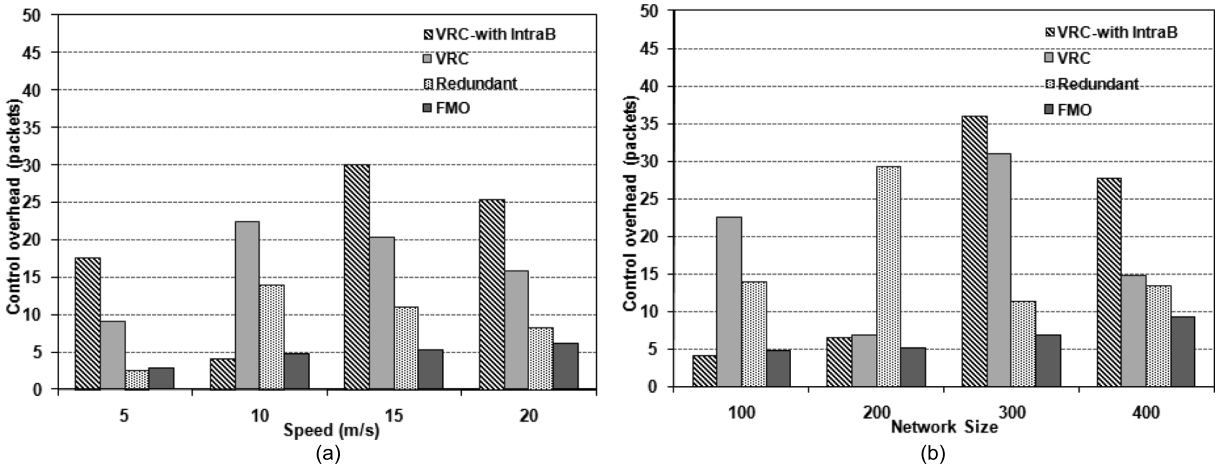

Fig. 10. Control packet overhead according to a) vehicle speed b) network size.

the emergency site would be reduced. The end-to-delay levels for VRC schemes were found to be similar to those for the FMO and redundant frames schemes when the network size was greater then 100 nodes. Therefore, Fig. 9a represents a worst case for the VRC schemes. Jitter was found to be broadly similar across the schemes and scenarios, with a range from 0.04 to 0.07 s. At 15 frame/s, this represents a need for a two-frame jitter buffer.

From Fig. 10, the number of control overhead packets was least for FMO, because only one stream was transmitted, whereas especially when speed was varied the redundant frame scheme resulted in reduced control packet overhead compared to the VRC schemes. However, the number of control packets expended to deliver a packet can be high. Moreover control packets add to the overall congestion in a network even though within a VANET the extra energy consumed in transmitting them may be discounted (as a vehicle's alternator is a convenient power source). This might be alleviated by employing a different multi-path routing protocol, e.g. [20], to route over multiple paths. However, this aspect is left to future work as the authors are aware that there are many issues involved in choice of protocol.

Therefore, combining the impact of extra latency and control overhead implies that a compromise

solution is to use FMO rather than redundant frames fro transmitting video streams in this scenario. In our tests, VRC is not only weaker in terms of delivered video quality but it increases latency and overhead.

7. Tentative analytical model

This Section constructs the components of a tentative analytical model that aims to provide a way of calculating delivered video quality. The long term delivered video quality depends on the connectivity of the network, as this determines the ability of vehicles to communicate across the VANET. In turn, the speed of the vehicles is required to determine the connectivity. For example, the speed of vehicle i at time $t, \nu_i(t)$, in the Manhattan grid model [2] of the simulations is given by:

$$\begin{aligned}\nu_i(t+\Delta t) &= \nu_{i+1}(t) - a/2, \text{ if } \Delta x_i(t) \leqslant \Delta x_{\min} \\ &= \tilde{\nu}_i(t+\Delta t), \text{ otherwise}\end{aligned} \tag{1}$$

$$\tilde{\nu}_i(t+\Delta t) = \min(\max(\nu_i(t) + \eta a \Delta t, \nu_{\min}), \nu_{\max}) \tag{2}$$

where $\Delta x_i(t)$ is the bumper-to-bumper distance from vehicle i to the vehicle in front, $\Delta x_{\min}$ is the minimum safety distance, a is the vehicle acceleration, and η is a uniformly distributed random variable in the range [0, 1]. Notice that unlike the well-known Random Waypoint model, in the Manhattan grid model the motion of a vehicle is dependent on its previous motion and on the position of the car in front. Other micro-cellular motion models can be found in [11]. As observed from the simulations in Section 6, node density has an important role. In order to incorporate node density into a stochastic model, the mathematically-tractable Poisson arrival process can be employed [18] to model vehicle arrivals along roads leading into the terrain (the Manhattan grid road layout). It is not claimed that a Poisson process is a model for the vehicle arrival process but use of this process does provide a point of comparison. The traffic intensity λ is then given by

$$\lambda = \frac{2f}{\nu_{mean}} \tag{3}$$

where ν_{mean} is the average speed of vehicles with a Poisson arrival process of intensity f in terms of vehicles/hour/road, with two lanes per road assumed in Eq. (3). In a Manhattan grid of size N $\times$ N blocks, there are 4(N-1) roads for vehicles to enter or leave the terrain. In [18], for a grid with N $= 10$, a road segment length of 400 m (100 m segments were used in our simulations), with a car-following driver model (dependent on the car in front with some modelling of driver behaviour [11]), it was found that intensity saturated at around $f = 800$ to 1000 vehicles/h/lane.

Assuming all vehicles are wireless-enabled, then wireless communication essentially depends on the range r. If there are some cars that are not wireless enabled then a factor ρ in the range [0,1] is introduced. The factor ρ also affects traffic congestion and in that way connectivity. However, in the simulations of this paper ρ is effectively set to unity. In steady-state, i.e. the number of vehicles leaving and entering the terrain is balanced, an upper bound on the expected number of vehicles that are *not* connected to any other vehicle, n, is:

$$E[n] = \exp(-2\lambda\rho r) \tag{4}$$

In the aforementioned setting [18], Eq. (4) was found to give a reasonable fit to simulated results for values of λ up to about 15 vehicles/km. However, this was for linear motion across the terrain, i.e. entering at one end of a road and leaving at the other end, and not to random destinations, as in this paper's results.

The range is governed by the propagation model. Thus, for the two-ray ground reflection path loss model used in our simulations, if the distance d is more than the cross-over distance [27] then the received power is:

$$P_r(d) = \frac{P_t g_t g_r h_t^2 h_r^2}{d^4 L} \tag{5}$$

where g_t and g_r are the gains of the transmitter and receiver respectively, h_t and h_r are the antenna heights of the transmitter and receiver respectively, P_t is the power at the transmitter, and L is the system loss. For distances less than the cross-over distance, the Friis free-space model [27] is applied, namely:

$$P_r(d) = \frac{P_t g_t g_r \Lambda^2}{(4\pi)^2 d^2 L} \tag{6}$$

where Λ is the wavelength. If power thresholding is applied at the receiver then this model allows a simple prediction of range. However, in the simulations packet length modelling was performed according to the form of the modulation for an AGWN channel [27]. The cross-over distance, d_c, is:

$$d_c = (4\pi h_t h_r)/\Lambda \tag{7}$$

Assuming some function, g, can be found that relates connectivity to packet loss then, from experience with the error-resilience techniques in the paper, these techniques act to additively increase the PSNR. For example, a 10% packet loss rate may achieve overall a fair video quality of 25 dB PSNR or above. However, if the packet loss rate is more that 10%, then error-resilience can restore the quality for excess (over 10%) packet loss rates up to total packet loss rate of about 20%. Thus, the effective packet loss rate is given by:

$$R_{eff} = g(E[n]) - C \tag{8}$$

where C is the reduction in packet loss rate due to error resilience (say 10%). IfR_{eff} is greater than a given threshold value (say) 10% then video quality is judged to be unacceptable for all vehicles in the VANET multicast group. This simple model may be adjusted according to the number of paths employed and the strength factor of the error-resilience method. However, further development of this model is beyond the scope of the present paper.

8. Conclusions

This paper has investigated whether the operating conditions in a city are likely to permit video communication. It has found that MDC with redundant frame insertion (a new suggestion for VANETs) and single stream FMO, both source coding techniques included in the H.264/AVC codec, can support robust communication when packet loss rates are relatively high. Video communication to a group of emergency vehicles allows recognition of suspect vehicles, description of burning buildings and the like to be passed from the first vehicle on the scene to approaching vehicles. End-to-end delay remains a

concern which should be addressed by reduction of hop counts by the routing protocol. As this is an outdoor scenario, location aware routing based on the global positioning system will be investigated. Further work will also consist of detailed investigation of urban wireless propagation conditions and modelling of driver behaviour.

References

[1] J. Apostolopoulos, Reliable video communication over lossy packet networks using multiple state encoding and path diversity, *Visual Comms: Image Processing* (Jan 2001), 392–409.

[2] F. Bai, N. Adagopan and A. Helmy, IMPORTANT: A framework to systematically analyze the Impact of Mobility on Performance of routing protocols over Ad hoc NeTworks, *IEEE INFOCOM* (April 2003), 825–835.

[3] S. Biswas, R. Tatchiko and F. Dion, Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety, *IEEE Communications Mag* **44**(1) (Jan 2007), 74–82.

[4] J.J. Blum, A. Eskandarian and L.J. Hoffman, Challenges of inter-vehicle ad hoc networks, *IEEE Trans. on Intelligent Transportation Systems* **5**(4) (2004), 347–351.

[5] P. Bucciol, E. Masala, N. Kawaguchi, K. Takeda and J.C. de Martin, Performance evaluation of H.264 video streaming over inter-vehicular 802.11 ad hoc networks, *IEEE 16th Int Symp on Personal, Indoor and Mobile Radio Comms* (Sept 2005), 1936–1940.

[6] N. Chand, R.C. Joshi and M. Misra, Cooperative caching in mobile ad hoc networks based on data utility, *Mobile Information Systems* **3**(1) (2006), 19–37.

[7] Y.C. Chang, M.S. Beg and T.S. Tang, Performance evaluation of MPEG-4 video error resilient tools over a mobile channel, *IEEE Trans on Consumer Electronics* **49**(1) (Feb 2003), 6–13.

[8] S.R. Dickey, C.-L. Huang and X. Guan, Field measurements of vehicle to roadside communication performance, *IEEE Vehicular Technol Conf* (Fall 2007), 2179–2183.

[9] J. Dunlop, D. Girma and J. Irvine, *Digital Mobile Communications and the TETRA System*, J. Wiley & Sons, Chichester, UK, 2000.

[10] M. Fiore, J. Härri, F. Filali and C. Bonnet, Vehicular mobility simulation for VANETs, *40th Annual Simulation Symposium* (Mar 2007), 301–307.

[11] M. Fiore and J. Härri, The network shape of vehicle mobility, *9th ACM Int'l Symposium on Mobile Ad Hoc Networking and Computing* (2008), 261–272.

[12] M. Gerla, R.G. Cascella, Z. Cao, B. Crispo and R. Battiti, An efficient weak secrecy scheme for network coding data dissemination in VANET, *IEEE PIMRC* (Sept 2008), 1–15.

[13] M. Ghanbari, Standard codecs: Image compression to advanced video coding, IET Press, London, UK, 2003.

[14] V.K. Goyal, Multiple description coding: Compression meets the network, *IEEE Signal Process Mag* **18**(5) (Sept 2001), 74–93.

[15] M. Guo, M.H. Ammar and E.W. Zegura, V3: A vehicle-to-vehicle live video streaming architecture, *3rd IEEE Int'l Conf. on Pervasive Computing and Comms* (Mar 2005), 171–180.

[16] A.M. Hanashi, I. Awan and M. Woodward, Performance evaluation with different mobility models for dynamic probabilistic flooding in MANETs, *Mobile Information Systems* **5**(1) (2009), 65–80.

[17] D. Jiang and L. Delgrossi, IEEE 802.11p: Towards an international standard for wireless access in vehicular environments, *IEEE Vehicular Technol Conf* (May 2008), 2036–2040.

[18] M. Kafsi, P. Papadimitratos, O. Dousse, T. Alpcan and J.-P. Hubaux, VANET connectivity analysis, *IEEE Workshop on Automotive Networking and Applications* (Dec 2008).

[19] P. Lambert, W. Deneve, Y. Dhondt and R. Vandewalle, Flexible macroblock ordering in H.264/AVC, *J of Visual Communication and Image Representation* **17** (2006), 358–375.

[20] S.-J. Lee and M. Gerla, Split multipath routing with maximally disjoint paths in ad hoc networks, *IEEE Int'l Conf. on Communications* (Jun 2001), 3201–3205.

[21] H.V. Leong and A. Si, Multi-resolution information transmission in mobile environments, *Mobile Information Systems* **1**(1) (2005), 25–40.

[22] L. Lima, M. Médard and J. Barros, Random linear network coding: A free cipher? *IEEE Int'l Symp on Info Theory* (Jun 2007).

[23] Q. Lu, L. Du, Z. Zuo and X. Xiao, Improved multi-path AODV protocols for real-time video transport over Mobile Ad Hoc Networks, *IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application* (2008), 621–625.

[24] D. Marpe, H. Schwarz and T. Wiegand, Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard, *IEEE Trans on Circuits and Systems for Video Technol* **13**(7) (2003), 620–636.

[25] J.S. Park, L. Uichin, S.Y. Oh, M. Gerla and D. S. Lun, Emergency related video streaming in VANET using network coding, *3rd International Workshop on Vehicular Ad Hoc Networks* (2006), 102–103.
[26] I. Radulovic, Y.-K. Wang, S. Wenger, A. Hallapuro, M.H. Hannuksela and P. Frossard, Multiple description H.264 video coding with redundant pictures, *Int'l Workshop on Mobile Video* (Sept 2007), 37–42.
[27] T.S. Rappaport, *Wireless Communications*, (2nd edition), Prentice-Hall, Upper Saddle River, NJ, 20012.
[28] M. Roccetti, M. Gerla, C.E. Palazzi, S. Ferretti and G. Pau, First responders' crystal ball: How to scry the emergency from a remote vehicle, *IEEE 26th Int'l Conf on Performance of Computing and Communs* (April 2007), 556–556.
[29] S. Saponara, C. Blanch, K. Denolf and J. Bormans, The JVT advanced video coding standard: Complexity and performance analysis on a tool-by-tool basis, *Int'l Packet Video Workshop* (April 2003).
[30] R.M. Schreier and A. Rothermel, Motion adaptive intra refresh for the H.264 video coding standard, *IEEE Trans. on Consumer Electronics* **52**(1) (2006), 249–253.
[31] T. Stockhammer and W. Zia, Error-resilient coding and decoding strategies for video communication, in: *Multimedia over IP and Wireless Networks*, M. van der Schaar and P.A. Chou, eds, Academic Press, Amsterdam, 2007, pp. 59–80.
[32] D. Tian, M.M. Hannuksela, Y.-K. Wang and M. Gabbouj, Error resilience coding techniques using spare pictures, *Int'l Packet Video Workshop* (April 2003).
[33] T. Tillo, M. Grangetto and G. Olmo, Redundant slice optimal allocation for H.264 multiple description coding, *IEEE Trans Circuits and Syst for Video Technol* **18**(1) (2008), 59–70.
[34] O.K. Tonguz, W. Viriyasitavat and F. Bai, Modeling urban traffic: A Cellular automata approach, *IEEE Communications Mag* **47**(5) (May 2009), 142–150.
[35] V. Varsa, M.N. Hannuksela and Y. Wang, Non-normative error concealment algorithms, ITU-T SGI6 Doc., VCEG-N62, 2001.
[36] Y. Wang, A.R. Reibman and S. Lee, Multiple description coding for video delivery, *Proc of the IEEE* **93**(1) (2005), 57–70.
[37] W. Wei and A. Zakhor, Multipath unicast and multicast video communication over wireless ad hoc networks, *Int'l Conf on Broadband Networks* (Oct 2004), 494–505.
[38] S. Wenger and M. Horowitz, Flexible MB ordering—a new error resilience tool for IP-based video, *Int'l Workshop on Digital Communications*, Capri, Italy, (2002).
[39] S. Wenger, H264/AVC over IP, *IEEE Trans Circuits and Syst for Video Technol* **13**(7) (2003), 645–656.
[40] S. Wenger, G.D. Knorr, J. Ou and F. Kossentini, Error resilience support in H.263+, *IEEE Trans Circuits and Syst for Video Technol* **8**(7), 867–877.
[41] T. Wiegand, G.J. Sullivan, G. Bjøntegaard and A. Luthra, Overview of the H.264/AVC video coding standard, *IEEE Trans Circuits and Syst for Video Technol* **13**(7) (July 2003), 560–576.
[42] Z. Wu and J.M. Boyce, Adaptive error resilient video coding based on redundant slices of H.264/AVC, *IEEE Int'l Conf. on Multimedia and Expo* (July 2007), 2138–2141.
[43] S. Yousefi, M.S. Mousavi and M. Fathy, Vehicular ad hoc networks (VANETS): Challenges and perspectives, *6th Int'l Conf on ITS Telecommunications* (Jan 2006), 761–766.
[44] H. Yu, Streaming media encryption, in: *Multimedia Security*, B. Furht and D. Kirovski, eds, CRC Press, Boca Raton, FO, 2005, pp. 197–220.
[45] X. Zeng, R. Bagrodia and M. Gerla, GloMoSim: A library for parallel simulation of large-scale wireless networks, *12th Workshop on Parallel and Distributed Simulations* (May 1998).

Nadia N. Qadri received her PhD at the School of Computer Science and Electronics Engineering, University of Essex, UK in 2010. She received her Masters of Engineering (Communication Systems and Networks) and Bachelors of Engineering (Computer Systems), from Mehran University of Engineering and Technology, Jamshoro, Pakistan in 2004 and 2002 respectively. She has more than four years of teaching and research experience at renowned universities of Pakistan viz. Mehran University of Engineering & Technology, Fatima Jinnah Women's University and COMSATS Institute of Information Technology. Her research interests include video streaming for mobile ad hoc networks and vehicular ad hoc networks, along with P2P streaming.

Muhammad Altaf received his BSc degree from the University of Engineering and Technology, Peshawar, Pakistan in 2001 and his MSc degree in computer system engineering from the National University of Science and Technology, Rawalpindi, Pakistan in 2004. He has recently been awarded his PhD at the University of Essex, UK. His research interests are video compression and video streaming over wired and wireless networks.

Martin Fleury has a degree in Modern History (Oxford University, UK) and a Maths/Physics based degree from the Open University, Milton Keynes, UK. He obtained an MSc in Astrophysics from QMW College, University of London, UK in 1990 and an MSc from the University of South-West England, Bristol in Parallel Computing Systems in 1991. He holds a PhD in Parallel Image Processing Systems from the University of Essex, Colchester, UK. He is currently employed as a Senior

Lecturer at the University of Essex. Martin has authored over 160 articles, and other publications on the subjects of low-level image- and signal-processing algorithms (including document and image compression algorithms), performance prediction of parallel systems, software engineering, and vision systems. His current research interests are video communication over MANS, WLANs, PANs, BANs, MANETs, and VANETs.

Mohammed Ghanbari is best known for his pioneering work on two-layer video coding for ATM networks (which earned him an IEEE Fellowship in 2001), now known as SNR scalability in the standard video codecs. He has served as an Associate Editor for IEEE Trans. on Multimedia. He has registered for eleven international patents on various aspects of video networking and was the co-recipient of A.H. Reeves prize for the best paper published in the 1995 Proc. of IEE on the theme of digital coding. He is the co-author of "Principles of Performance Engineering", a book published by IET press in 1997, the author of "Video Coding: An Introduction to Standard Codecs", a book also published by IET press in 1999, which received the year 2000 best book award by the IEE, and the author of "Standard Codecs: Image Compression to Advanced Video Coding" also published by the IET press in 2003. Prof. Ghanbari has authored or co-authored about 450 journal and conference papers, many of which have had a fundamental influence in this field.

Mobile Information Systems 6 (2010) 249–257
DOI 10.3233/MIS-2010-0102
IOS Press

An enhanced MPR-based solution for flooding of broadcast messages in OLSR wireless ad hoc networks

Tzu-Hua Lin[a], Han-Chieh Chao[a,b,*] and Isaac Woungang[c]
[a]*Department of Electrical Engineering, National Dong Hwa University, Hualien, Taiwan*
[b]*Institute of Computer Science & Information Engineering, and Department of Electronic Engineering, National Ilan University, I-Lan, Taiwan*
[c]*Department of Computer Science, Ryerson University, Toronto, Canada*

Abstract. In an Optimized Link State Routing (OLSR)-based mobile wireless network, optimizing the flooding of broadcast messages is a challenging task due to node's mobility and bandwidth resource consumption. To complement existing solutions to this problem, the Multi-Point Relays (MPR) selection has recently been advocated as a promising technique that has an additional feature of reducing the number of redundant re-transmission occurring in the network. This paper continuous on the investigation of an existing MPR-based solution, arguing that by considering a cost factor as an additional decision parameter in selecting the MPR nodes, the enhanced MPR selection algorithm leads to less packet loss in the network. Simulation experiments are presented to validate the stated goal, using the average packet loss ratio as the performance metric.

Keywords: Mobile wireless ad hoc network, link stability, OLSR, proactive routing, flooding, MPR selection

1. Introduction

With the widespread availability and rapid evolution of wireless local area network technologies such as 802.11 [8], Bluetooth [6], to name a few, the use of MANETs is growing fast [9–11]. In a MANET, there is no need for an infrastructure or a centralized administration for nodes to communicate with each other. Nodes cooperate to provide connectivity and services. However, this flexibility is accompanied by several challenges, in particular, from a routing prospective. In MANETs, the bandwidth, power, speed of mobile nodes, density of the topology, distribution and location of mobile nodes, are among the various factors that can influence the routing process [3,12], hereby the establishment of a routing path between a node pair. These factors, if not well controlled, can lead to an increasing packet loss and a decreasing network performance and link stability. As far as routing is concerned, one of the primary challenges in a MANET is to determine methods for reducing the waste of bandwidth and power consumption while ensuring a stable transmission of information, and quickly responding to network

*Corresponding author: Han-Chieh Chao, Institute of Computer Science and Information Engineering, National I-lan University, Sec. 1, Shen-Lung Rd., I-Lan, Taiwan, Republic of China. Tel.: +886 3 9357400 – x.251; Fax: +866 3 9354238; E-mail: hcc@niu.edu.tw.

changes. Of course, these methods depend on the capability of optimizing broadcast messages in the network.

To cope with this issue, one of the recent proactive protocols known as OLSR [13] has been advocated as a promising routing protocol for multi-hop wireless ad-hoc networks. In OLSR, only a subset of pre-selected nodes referred to as MPR nodes have specific functionalities of performing topological advertisements as well as broadcasting and forwarding of control messages, with the goal to reduce the impact of message flooding and control overhead. The set of MPR nodes is chosen in such a way that a minimum of one-hop symmetric neighbors is able to reach all the symmetric two-hop neighbors. To calculate the MPR set, the node must possess the knowledge of the link state information about all one-hop and two-hop neighbors. The objective of the MPR technique is to reduce the number of redundant retransmissions, while ensuring reliable delivery of broadcast messages. In a MPR-based OLSR wireless ad hoc network, every node needs to issue periodic Hello message to update the 1-hop neighbors set and 2-hop neighbors set tables. Every node must dynamically select a neighbor node as MPR node at a time. After this selection is completed, messages are broadcasted, and control messages as well as traffic data that all packets should transmit through MPR nodes are identified.

This paper continues on the investigation of a heuristic introduced in [4] (our so–called MPR-based OLSR heuristic) for the selection of MPR nodes. We consider a cost factor [4] as an additional decision parameter within the MPR-based OLSR heuristic, and we demonstrate by simulation that the resulting heuristic (so-called Enhanced MPR-based OLSR heuristic – here denoted as EMPR-based OLSR) has superior performance over MPR-OLSR in terms of average packet loss ratio in the network. Our primary objective is to investigate the impact of packet loss during transmission when the coverage of a node's selected MPR increases due to the introduction of the above-mentioned additional cost factor in the computation of the MPR selection node.

The paper is organized as follows. Section 2 presents the related works. Section 3 overviews the MPR selection concept. Section 4 describes the proposed EMPR-based OLSR heuristic and contrasts it against the MPR-based OLSR approach. In Section 5, simulation results comparing the proposed EMPR-based heuristic against the MPR-based heuristic are presented. To this effect, OPNET [14] is used as the simulation tool, and the packet loss is considered as the performance metric. Finally, in Section 6, some concluding remarks are given.

2. Related work

The selection of the MPR set of nodes is a fundamental operation in the OLSR protocol. Most solutions to this problem are in the form of heuristics. Representative ones deal with MPR-based flooding techniques [2,4,15,16], as well as reducing the number of collisions or maximizing the bandwidth [7, 17]. With respect to link stability, the authors in [18] studied the link lifetime using various mobility models. They derived a formula to calculate the expected value of each node in order to select high expected value of MPR. Similarly, in [5], some statistical techniques based on link stability metrics were used for selecting MPR nodes while enhancing the route reliability and decreasing the packet loss. Other OLSR enhancements deal with quality of service routing based MPR selection, by combining the MPR selection mechanism with the path determination algorithm [19–21]. In almost all these representative solutions, the goal has been to determine the impact the selection of MPRs has on the performance of OLSR

This paper introduces an additional parameter in the MPR selection method proposed in [4], namely the cost values calculation by all 1-hop and 2-hop neighbor nodes. Our goal is to improve the process of MPR

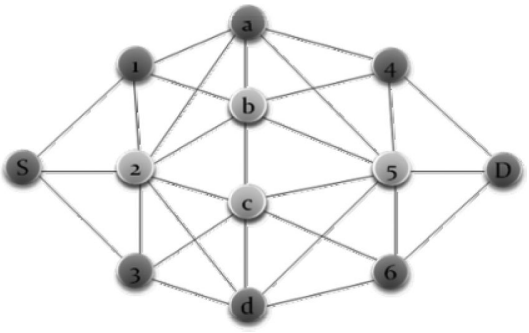

Fig. 1. Example network.

through reducing the total packet loss during transmission, thereby, achieving a superior performance than that of the MPR-based OLSR heuristic.

3. Overview of MPR selection

The OLSR protocol has been extensively investigated in the literature. Rather than re-describing its basic functionality, this Section is meant to setup the context of the study carried in this paper. OLSR is an optimization of the pure link state-based routing protocol [13]. Two important concepts were introduced in the RFC3626 OLSR draft [23] that can enable OLSR extensions, namely, neighbor discovery and Multipoint-Relay selection (MPR). This paper focuses on using MPR. The later is particularly important since it has an immediate effect on the routing protocol's performance. Indeed, the overhead of control traffic generated by the OLSR protocol as well as the flooding efficiency depends on the MPR selection, which itself, is known as a NP-hard problem [17]. In the MPR-based OLSR protocol, each node selects a subset of its one-hop neighbors as its MPR list, based on periodic HELLO messages received by these neighborhood nodes (one-hop neighbors and two-hop neighbors). The compilation of MPR lists then builds a database of nodes that are used in the routing process to compute the shortest paths to all possible destinations in the network. Compared to classical link-state algorithms, MPR-based OLSR algorithms have been proved to significantly reduce the number of message re-transmissions and control traffic [1, 2,4,7,16,18,19,22–24], to name a few.

This paper continues the investigation of the MPR-based OLSR heuristic proposed in [4]. One of the drawbacks of this heuristic is that in the employed MPR technique, only a selected number of nodes propagate the message, which might limit the chances that the message reaches all the nodes in the network. The situation is even worse when the bit error rate in the radio transmission is high [4,13], leading to dramatic packets loss in the network. The MPR-based OLSR heuristic [4] proceeds in three main steps (as detailed in the next section) among which the second one is devoted to the optimization of the MPR set. An analysis of this MPR-based OLSR heuristic has shown that it is within a log n factor from optimality [4].

Our enhancement to the above MPR-based OLSR (so-called EMPR-based OLSR) consists of keeping the core of the MPR-based OLSR heuristic, but introducing a cost factor (in the above-mentioned Step 2) as an additional decision parameter when designing the optimized MPR set.

The EMPR-based OLSR heuristic produces a larger cover range for the MPR set compared to that of the MPR-based OLSR heuristic. As an example, the MPR-based OLSR heuristic applied to the example network shown in Fig. 1 produces $\{2, b, c, 5\}$ as optimized MPR set with a cover range as depicted in

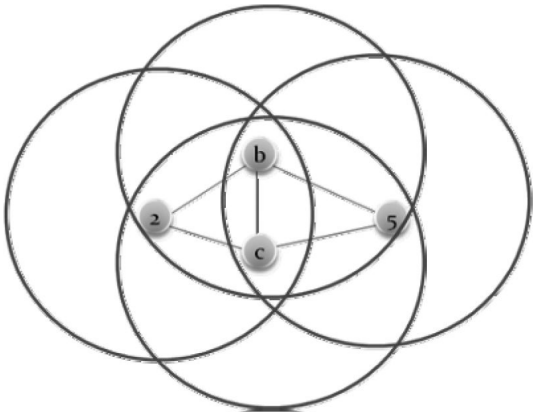

Fig. 2. MPR set cover range when using the MPR-based OLSR heuristic.

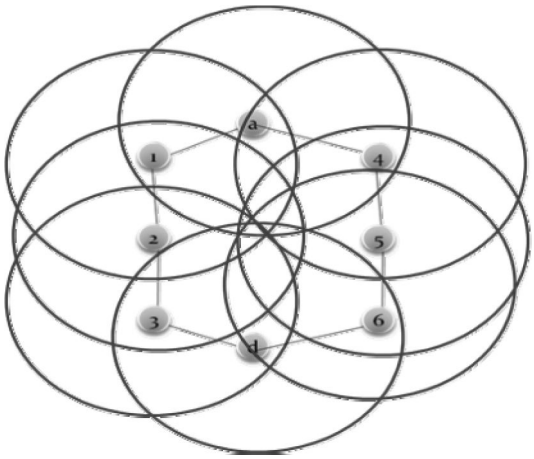

Fig. 3. MPR set cover range when using the EMPR-based OLSR heuristic.

Fig. 2, whereas for the same network, our EMPR-based OLSR heuristic yields {1, 2, 3, a, d, 4, 5, 6} as optimized MPR set with a cover range as depicted in Fig. 3.

Our goal is to demonstrate that the EMPR-based OLSR heuristic outperforms the MPR-based OLSR heuristic in terms of packets loss reduction in the network. In the sequel, we adopt the same notations that were used in [4] and we describe the MPR-based OLSR and the EMPR-based OLSR heuristics.

4. Enhanced MPR-based OLSR

The network is represented as a graph G (i.e. a set V of nodes and a set of neighbors for each node). For each element x in V, M(x) is the neighborhood of x, N(x) is the set of one-hop neighbors of x (i.e. x covers any element of N(x)), $N^2(x)$ is the set of two-hop neighbors of x, D(x) is the degree of x (i.e. the number of links connected to x), and MPR(x) is the selected MPR set of node x.

The MPR-based OLSR heuristic as introduced in [9] can be summarized as follows:

– Step 1: Start with an empty MPR set MPR(x)

– Step 2: Calculate N(x), N^2(x) and D(x) for all x in N(x)
– Step 3: Select as MPR nodes those nodes in N(x) which are the only neighbor of some node in N^2(x)
– Step 4: As long as there exist some node in N^2(x) which is not covered by the set MPR(x), do the following:

  * For each node z in N(x) which is not element of MPR(x), determine the number of nodes that z covers among uncovered nodes in N^2(x).
  * Add in MPR(x) the node z for which this number is maximum

Let S_1 be the set of nodes selected in Step 3 and (x_1,..., x_k) be the set of nodes selected in the above-mentioned Step 3 and Step 4 of MPR-based OLSR heuristic respectively. Let N_1^2be the set of nodes in N^2(s) that are neighbors of some nodes s in S_1. Let $N^{2'} = N^2(s) - N_1^2$. A unit cost Cy, associated with each node y in $N^{2'}$, is defined as follows [8]:

$$c_y = \frac{1}{|N'(x_j) - \cup_{j=1}^{j-1} N'(x_j)|} \tag{1}$$

where |P| denotes the cardinality of the set P. As pointed out in [4], for each x_t chosen by the MPR-based OLSR algorithm, this unit cost is equally distributed among nodes that are newly covered in N^2. Our proposed EMPR-based OLSR heuristic takes advantage of this feature. Its stepwise description follows.

– Step 1: Same as in the MPR-based OLSR heuristic
– Step 2: Calculate N(x), N^2(x) and D(x) for all x in N(x). Calculate the cost value C_z associated to each node z in N(x) using Eq. (1)
– Step 3: Select as MPR nodes those nodes in N(x) which are the only neighbor of some node in N^2(x). Enlarge the MPR(x) set by adding those nodes in N(x) that have the lowest cost values (there might be only one such node or many such nodes)
– Step 4: As long as there exists some node in N^2(x) which is not covered by the set MPR(x), do the following:

  * For each node z in N(x) which is not element of MPR(x), determine the number of nodes that z covers among uncovered nodes in N^2(x).
  * Add in MPR(x) the node z for which this number is maximum or add in MPR(x) the node z with lower cost value.

The distinction between the MPR-based OLSR heuristic and the EMPR-based heuristic relies in the introduction of the aforementioned cost value C_z associated with each node z in N(x), which contributes to the construction of the optimized MPR(x) set.

It should be noticed that the introduction of the cost value in the MPR selection calculation has resulted to the following impacts. The first one is an increase of the MPR count. In this case, Step 4 of the proposed EMPR heuristic will attempt to control this side effect by allowing each node to periodically update its neighbor table using "HELLO Messages", which would result in allowing each MPR selected node to flood TC message to all other MPR nodes in the network. The second one is an increase of the coverage of a node's selected MPRs, which in turn will increase the redundancy of TC-messages forwarding. In this case, the MPR-forwarding strategy described in Step 4 of our EMPR heuristic would handle the duplicate re-transmissions directly at the design level when selecting the MPR node set. Indeed, when a mobile node n1 will receive a packet, each node n2 would be required to determine whether node n1 has been selected as MPR node – note that this is realized through computing the node

Table 1
Simulation Environment and Parameters setup

Parameter	Meaning
Number of nodes	50
Communication range	200
Mobility model	Random waypoint
Wireless LAN	802.11g
IP protocol	IPv6
Transmission type	G723 voice bps
Data rate	11 Mbps
Maximum received lifetime	0.5 sec
Simulation start time	10 sec
Channel match criterion	Strict match
Traffic duration	3600 sec
Traffic mix	All explicit
Traffic start time	60 sec
Simulation area	2*2 km^2

n2's neighbor's table, the node n2's two-hop neighbor table and the number of neighbors of node n2. If the answer is yes, then node n1 will be sending the packets to all other designated neighbor nodes. Otherwise, node n1 will not send the packets, but will remain active.

For both heuristics, experimenting the processes of MPR selection within a specific node using OPNET revealed that this additional decision parameter helps the EMPR-based OLSR heuristic producing a larger cover range for the MPR set compared to that generated by the MPR-based OLSR heuristic. This feature helps reducing the number of packets loss in the network during the transmission period (as illustrated through the simulation results shown in Section 5).

5. Performance evaluation

In this Section, we evaluate the performance of our EMPR-based OLSR heuristic against that of the MPR-based heuristic via simulations. The performance comparison metric is the average packet loss ratio, representing the ratio of packets lost to the total packets generated in a certain time period. The simulation tool is OPNET [14] and the location of the nodes is random. The simulation environment and parameters setup are captured in Table 1.

We have used a single seed value and have ran several simulations. For the same scenario, the simulation time was set to 1 hour and the node's speed was varied from 10 m/s to 90 m/s.

Figure 4 depicts the speed of the mobile node (x axis) versus the amount of traffic it has received (y axis) in terms of number of bits. Figure 5 illustrates the MPR count time average versus the mobile node's speed when running both heuristics. Finally, Fig. 6 depicts the average packet loss ratio versus the mobile node's speed when running both heuristics. It is observed that the EMPR-based heuristic outperforms the MPR-based OLSR heuristic in terms of total number of packet loss during transmission.

6. Concluding remarks

In this paper, we have proposed an enhanced version (called EMPR-based OLSR heuristic) of an existing MPR-based OLSR selection method in MANETs (referred to as MPR-based OLSR heuristic). Our proposal considers the cost value as an additional factor of MPR selection. Our simulation results

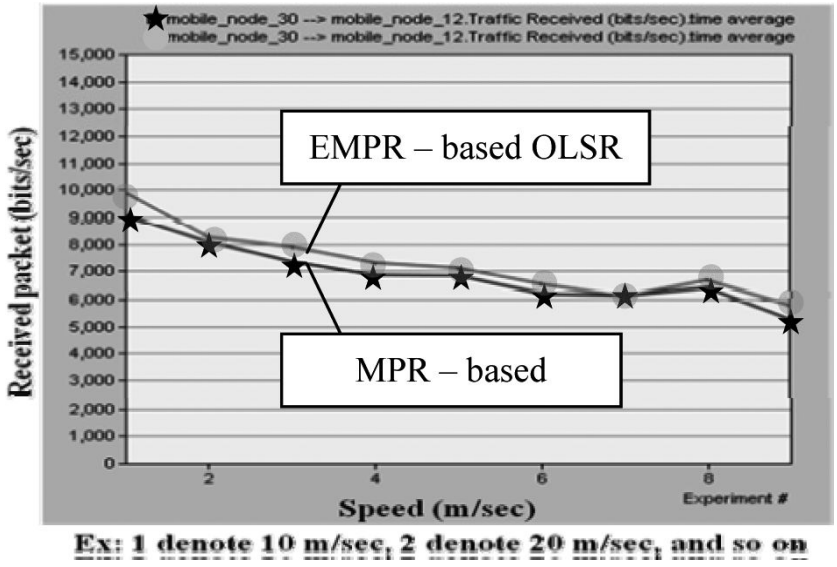

Fig. 4. Mobile node's speed versus the amount of traffic received.

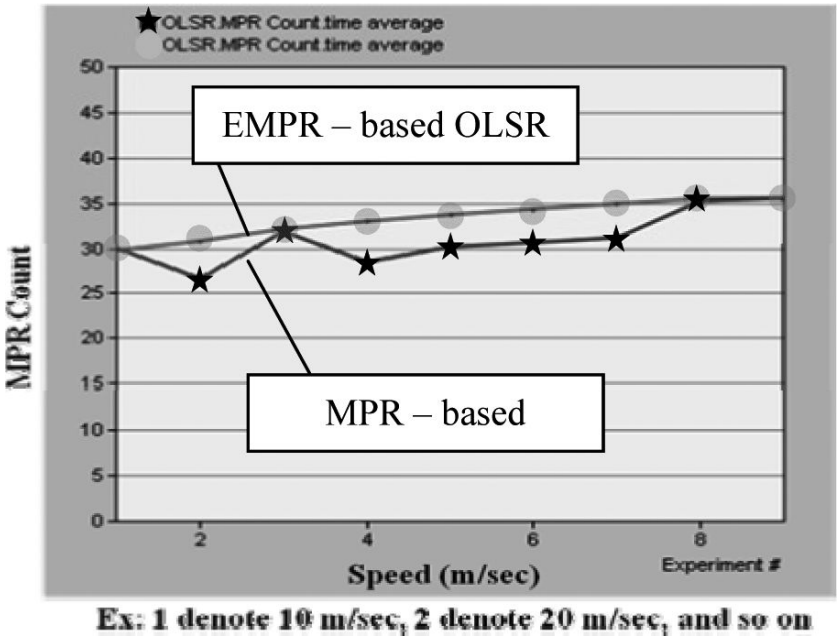

Fig. 5. MPR count versus mobile node's speed.

have shown that the EMPR-based OLSR heuristic yields a better average packet loss ratio compared to the MPR-based OLSR heuristic. As future work, comparing the proposed EMPR-based OLSR heuristic against few other existing MPR-based selection methods for link stability in MANETs would be an interesting work. Also, we have not simulated the change in MPR coverage, nor investigating the NS_MPR forwarding TC message strategy.

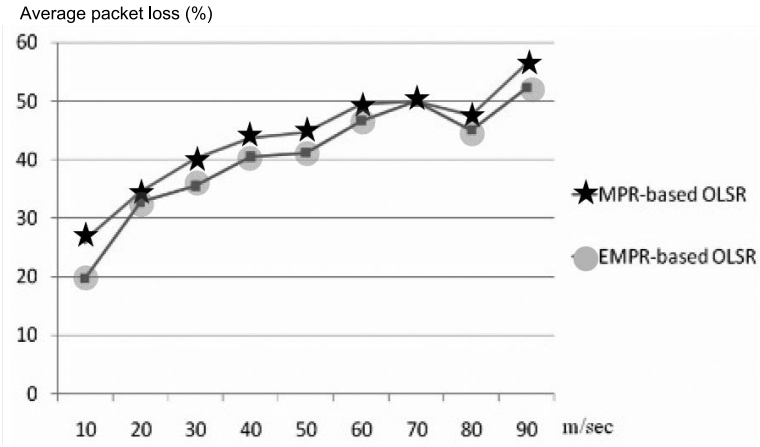

Fig. 6. Average packet loss versus the mobile node's speed.

Acknowledgements

This work is partially supported by National Science Council of Taiwan, under grant number NSC97-2219-E-197-001 and NSC97-2219-E-197-002.

References

[1] A. Benslimane, R. El Khoury, R. El Azouzi and S. Pierre, *Energy Power-Aware Routing in OLSR Protocol*, In Proc. 1st International Mobile Computing and Wireless Communication Conference, Sept. 2006, pp. 14–19.
[2] A. Laouiti, P. Muhlethaler, A. Najid and E. Plakoo, *Simulation Results of the OLSR Routing Protocol for Wireless Network*, MedHoc-Net, Sardegna, Italy, 2002.
[3] A.M. Hanashi, I. Awan and M.E. Woodward, Performance Evaluation with Different Mobility Models for Dynamic Probabilistic Flooding in MANETs, *Mobile Information Systems* **5**(1) (2009), 65–80.
[4] A. Qayyum, L. Viennot and A. Laouiti, *Multipoint Relaying for Flooding Broadcast Messages in Mobile Ad Hoc Networks*, Proc. of 35th International Conference on System Sciences, Hawaii, USA, 2002.
[5] B. Mans and N. Shrestha, *Performance Evaluation of Approximation Algorithms for Multipoint Relay Selection*, Med-Hoc-Net04, Bodrum, Turkey, June 27–30, 2004
[6] Bluetooth Specifications, http://www.bluetooth.org/ (Last visited June 3, 2010).
[7] D. Gantsou, P. Sondi and S. Hanafi, Revisiting Multipoint Relay Selection in the Optimized Link State Routing Protocol, International Journal of Communication Networks and Distributed Systems, *Inderscience, UK* **2**(1) (2009), 4–15.
[8] IEEE Standards, Standards.ieee.org/getieee802/802.11.html (Last visited June 3, 2010).
[9] S. Corson and J. Macker, *Mobile Ad hoc Networking* (*MANET*)*: Routing Protocol Performance Issues and Evaluation Considerations*, RFC2501, Jan. 1999, http://www.faqs.org/rfcs/rfc2501.html (Last visited June 3, 2010).
[10] H. Zhang, H. Zhang, J. Guan and Y. Qin, A New Architecture of Multicast Interworking between MANET and Fixed Network, *Journal of Internet Technology* **8**(1) (2007), 141–150.
[11] J. Chen, AMNP: ad hoc multichannel negotiation protocol with broadcast solutions for multi-hop mobile wireless networks, *IET Communications* **4**(5) (2010), 521–531.
[12] V. Pham, E. Larsen, ØKure and P. Engelstad, Routing of internal MANET traffic over external networks, *Mobile Information Systems* **5**(3) (2009), 291–311.
[13] T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, A. Qayyum and L. Viennot, *Optimized Link State Routing Protocol*, IEEE INMIC Pakistan, 2001.
[14] OPNET Simulator, http://www.opnet.com/ (Considerations, RFC2501, Jan. 1999, http://www.faqs.org/rfcs/rfc2501.html), (Last visited June 3, 2010).
[15] T. H. Clausen, G. Hansen, L. Christensen and G. Behrmann, *The Optimized Link State Routing Protocol, Evaluation through Experiments and Simulation*, IEEE Symposium on Wireless Personal Mobile Communications, Sept. 2001.
[16] P. Jacquet, A. Laouiti, P. Minet and L. Viennot, *Performance of Multipoint Relaying in Ad Hoc Mobile Routing Protocols*, Networking 2002, Pise, Italy, 2002.

[17] P. Jacquet, A. Laouiti, P. Minet and L. Viennot, *Performance Analysis of OLSR Multipoint Relay Flooding in Two Ad Hoc Wireless Network Models*, Research Report-4260, INRIA, Sept. 2001, RSRCP Journal, Special Issue on Mobility and Internet.
[18] E. Baccelli and P. Jacquet, Flooding Techniques in Mobile Ad-Hoc Networks, Research Report RR-5002, INRIA, 2003.
[19] M. Gerharz, C. Waal, de, M. Frank and P. Martini, *Link Stability in Mobile Wireless Ad Hoc Networks*, In Proc. of the 27th Annual IEEE Conference on Local Computer Networks (LCN), Nov 6–8, Tampa, Florida, USA, pp. 30–39, 2002.
[20] S. Obilisetty, A. Jasti and R. Pendse, *Link Stability Based Enhancements to OLSR* (*LS-OLSR*), In Proc. of 62nd IEEE Vehicular Technology Conference (VTC-2005), Dallas, USA, Sept. 2005.
[21] N. Ghanem, S. Boumerdassi and E. Renault, *New Energy Saving Mechanisms for Mobile Ad Hoc Networks using OLSR*, In Proc. of the 2nd ACM Intl. Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks, Oct. 2005, pp. 273–274.
[22] Z. Guo and B. Malakooti, *Energy Aware Proactive MANET Routing with Prediction on Energy Consumption*, In Proc. of the Intl. Conference on Wireless Algorithms, Systems and Applications, Aug. 2007, pp. 287–293.
[23] RFC3626 Draft, http://www.ietf.org/rfc/rfc3626.txt (Last visited June 3, 2010).
[24] M. Ikeda, L. Barolli, G. De Marco, T. Yang, A. Durresi and F. Xhafa, Tools for Performance Assessment of OLSR Protocol, *Mobile Information Systems* **5**(2) (2009), 165–176.

Lin Tzu-Hua was born in Kaohsiung, Taiwan in 1984. He received his B.S. and M.S. degrees from the Department Electrical Engineering, Da-Yeh University and National Dong Hwa University in 2003 and 2009 respectively. Since then, he has been with the National Cheng Kung University, Tainan, Taiwan, working towards his Ph. D degree. His research interests include Mobile communication systems, Vehicle wireless communication, Cooperative networking, and Network coding.

Han-Chieh Chao is a joint appointed Full Professor of the Department of Electronic Engineering and Institute of Computer Science & Information Engineering. He also serves as the Dean of the College of Electrical Engineering & Computer Science for National Ilan University, I-Lan, Taiwan, R.O.C. He has been appointed as the Director of the Computer Center for Ministry of Education starting from September 2009 as well. His research interests include High Speed Networks, Wireless Networks, IPv6 based Networks, Digital Creative Arts and Digital Divide. He received his MS and Ph.D. degrees in Electrical Engineering from Purdue University in 1989 and 1993 respectively. He has authored or co-authored 4 books and has published about 200 refereed professional research papers. He has completed 80 MSEE thesis students and 2 PhD students. Dr. Chao has received many research awards, including Purdue University SRC awards, and NSC research awards (National Science Council of Taiwan). He also received many funded research grants from NSC, Ministry of Education (MOE), RDEC, Industrial Technology of Research Institute, Institute of Information Industry and FarEasTone Telecommunications Lab. Dr. Chao has been invited frequently to give talks at national and international conferences and research organizations. Dr. Chao is also serving as an IPv6 Steering Committee member and co-chair of R&D division of the NICI (National Information and Communication Initiative, a ministry level government agency which aims to integrate domestic IT and Telecom projects of Taiwan), Co-chair of the Technical Area for IPv6 Forum Taiwan, the executive editor of the Journal of Internet Technology and the Editor-in-Chief for International Journal of Internet Protocol Technology and International Journal of Ad Hoc and Ubiquitous Computing. Dr. Chao has served as the guest editors for Mobile Networking and Applications (ACM MONET), IEEE JSAC, IEEE Communications Magazine, Computer Communications, IEE Proceedings Communications, the Computer Journal, Telecommunication Systems, Wireless Personal Communications, and Wireless Communications & Mobile Computing. Dr. Chao is an IEEE senior member and a Fellow of IET (IEE). He is a Chartered Fellow of British Computer Society.

Isaac Woungang received his M.S. and Ph. D degrees in Mathematics from the Universit¤*f* de la M¤*f*diterran¤*f*e-Aix Marseille II, France, and the Universit¤*f* du Sud, Toulon et Var, France, in 1990 and 1994 respectively. In 1999, he received a M.S from the INRS-Materials and Telecommunications, University of Quebec, Montreal, Canada. From 1999 to 2002, he worked as a software engineer at Nortel Networks. Since 2002, he has been with Ryerson University, where he is an Associate professor of Computer Science. In 2004, he founded the Distributed Applications and Broadband NEtworks Laboratory (DABNEL) R&D group. His research interest includes Network security, Computer communication networks, Mobile communication systems, computational intelligence applications in telecommunications, and Coding theory. Dr. Woungang serves as Editor-in-Chief of the International Journal of Communication Networks & Distributed Systems (IJCNDS), Inderscience, U.K, and the International Journal of Information and Coding Theory (IJICoT), Inderscience, U.K, as Associate Editor of the International Journal of Communication Systems (IJCS), Wiley, and Associate Editor of Computer and Electrical Engineering, An International Journal (C&EE), Elsevier. Dr. Woungang edited several books in the areas of wireless ad hoc networks, wireless sensor networks, wireless mesh networks, communication networks & distributed systems, and information & coding theory, published by reputed publishers such as Springer, Elsevier, and World Scientific.

Mobile Information Systems 6 (2010) 229–247
DOI 10.3233/MIS-2010-0101
IOS Press

Fault reconnaissance agent for sensor networks

Elhadi M. Shakshuki[a,*], Xinyu Xing[b] and Tarek R. Sheltami[c]
[a]*Jodrey School of Computer Science, Acadia University Wolfville, Nova Scotia, B4P 2R6 Canada*
[b]*Department of Computer Science, University of Colorado at Boulder, CO, USA*
[c]*Computer Engineering Department, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia*

Abstract. One of the key prerequisite for a scalable, effective and efficient sensor network is the utilization of low-cost, low-overhead and high-resilient fault-inference techniques. To this end, we propose an intelligent agent system with a problem solving capability to address the issue of fault inference in sensor network environments. The intelligent agent system is designed and implemented at base-station side. The core of the agent system – problem solver – implements a fault-detection inference engine which harnesses Expectation Maximization (EM) algorithm to estimate fault probabilities of sensor nodes. To validate the correctness and effectiveness of the intelligent agent system, a set of experiments in a wireless sensor testbed are conducted. The experimental results show that our intelligent agent system is able to precisely estimate the fault probability of sensor nodes.

Keywords: Management, intelligent agents, expectation maximization algorithm, wireless sensor networks

1. Introduction

An embedded sensor network is a system of nodes, each of which is equipped with a certain amount of sensing, actuating, computation, communication and storage components. Two major components of sensor nodes are sensing unit and wireless transceiver. They directly interact with nodes in wireless sensor networks (WSNs) that are easily prone to failure due to hardware failure, communication link errors, energy depletion, malicious attacks, etc. Even if the sensor node hardware is in excellent condition, still the communication between sensor nodes is dependent upon many factors such as signal strength, obstacles and interferences. Degradation in these factors results in low reliability of sensor nodes. One of the key prerequisite for a scalable, effective and efficient sensor network is the utilization of low-cost, low-overhead and high-resilient fault-inference techniques. To this end, we propose an intelligent agent system with a problem solving capability to address the issue of fault inference in sensor network environments. The core of the intelligent agent system is the problem solver, which implements a fault-detection inference engine that harnesses Expectation Maximization (EM) algorithm to estimate fault probabilities of sensor nodes.

Due to the characteristics (e.g. energy awareness, constraint bandwidth and so on) of wireless sensor networks, it is infeasible for each sensor to announce its working state to a centralized node (base station).

*Corresponding author: Jodrey School of Computer Science, Acadia University, Nova Scotia, Canada B4P 2R6, E-mail: elhadi.shakshuki@acadiau.ca.

Therefore, we propose an intelligent agent system which utilizes EM algorithm to infer the sensor-node fault probabilities from a limited dataset. The benefits of our intelligent agent system are summarized as follows. (1) The intelligent agent system does not involve extra message transmission, since it can infer sensor-node fault probabilities directly from aggregated dataset that sensor nodes collect. (2) The intelligent agent system can provide highly accurate estimation.

In WSNs, it is natural that sensor nodes experience some faults at high frequencies due to the following two factors [2]. Firstly, WSNs put significant constraints on the resource expenditure. Nodes operate under strict energy constraints, which limits the amount of energy devoted to testing and fault tolerance. Secondly, some applications are equally complex with the involved technology and architecture. Sensor networks often operate without human intervention [31]. Furthermore, security and privacy concerns prevent extensive testing procedures. It should be noted that not only testing and fault tolerance are adversely impacted, but related tasks such as debugging where reproduction of specific conditions under which fault has occurred are extremely difficult. Similarly, sensor nodes are often deployed in uncontrolled and sometimes even hostile environment for surveillance and detection [35,36]. Therefore, knowing sensor-node working states is indispensable.

Due to the energy awareness characteristics in WSNs, there is a trade-off between prolonging the network lifetime via conserving the energy of individual nodes, and maintaining the high quality of network services by implementing complex fault management schemes in a sensor network. To the best of our knowledge, contemporary fault management mechanisms have to involve extra traffic overhead and energy expenditure. Therefore, most researchers regularly believed that the expenditure for fault management is fairly expensive. Based on our previous study and the following two motivations; we however, introduce an intelligent agent named ATLAS (An inTelligent agent for fauLt reconnAisSance in sensor networks).

1.1. Motivation-1: Energy awareness based on data aggregation

Over the past few years, several researches have attempted to propose various mechanisms to prolong the network lifetime [32]. An agent-based paradigm is a trend [3–9]. This paradigm adopts energy-efficient data aggregation to eliminate the redundant transmission and thus to minimize energy consumption. Under the agent-based paradigm, communication links and partial nodes in a sensor network are usually perceived as a reverse multicast tree. Generally, a sink node dispatches mobile agents with processing code to destination nodes through the reverse multicast tree. After collecting the sensing data from the destination nodes, mobile agent migrates to the sink carrying aggregated data. To the best of our knowledge, contemporary researches on the agent-based paradigm fail to take the fault management into consideration.

1.2. Motivation-2: Infeasible individual fault reconnaissance

Fault management in sensor networks can be classified into three phases: fault reconnaissance, fault diagnosis and recovery. Fault reconnaissance is the most fundamental phase for the simple reason that only if faults are identified, previously established mechanisms and algorithms [16–19,37] can be further applied for fault diagnosis and its recovery. Obviously, the trustworthiness and effectiveness of fault reconnaissance, to a large extent, rely on utilization of communication and hardware resources (i.e. bandwidth and energy). Consider the following scenario. Each of sensor nodes attempts to report (either directly or indirectly) its own fault probability to a command center (sink) in a scalable sensor network. This incurs tremendous energy consumption and wastes limited bandwidth resources. According to

our literature survey on fault management, even though some researchers presented some efficient optimization methodologies for fault reconnaissance (as mentioned in Section 2), these methodologies still result in extra traffic overhead and energy consumption.

To achieve effective, efficient fault reconnaissance with "zero" extra traffic overhead as well as energy expenditure, we introduce ATLAS – an intelligent agent for fault reconnaissance in sensor networks. The core of ATLAS, inference engine, is abstracted as a mathematical model based on Expectation Maximization (EM) algorithm [20]. The mathematical model of our proposed agent's problem solver is based on two assumptions. (1) Data gathering mechanism utilize reverse multicast tree. (2) The reverse multicast tree is known and is relatively stationary. One of the main contributions of this work is to utilize an agent-based centralized approach [33] to infer and identify failures among sensor nodes. At sink, an agent is loaded with fault inference engine that profits from an expectation-maximization algorithm to identify faults at the cost of reduced energy expenditure and bandwidth saturation. Experimental evaluation of our ATLAS prototype in a wireless sensor testbed, developed at University of Colorado, Boulder, shows that ATLAS is effective at inferring faults.

The rest of the paper is organized as follows. Section 2 presents related work. Section 3 provides the basic mathematical model. Section 4 introduces an intelligent agent – ATLAS. Section 5 describes and discusses experimental results. Finally, Section 6 provides conclusions and future research.

2. Related work

Fault reconnaissance is the first phase of fault management, where an unexpected failure should be properly identified within a network. The fault reconnaissance approaches in WSNs are generally classified into two types: centralized and distributed approach.

The centralized approach is the most common approach to identify the cause of failures or suspicious nodes in WSNs. Usually, a centralized sensor node equipped with abundant power supply (base station manager [22,23], central controller [24] or sink [25]) that monitors [failed or misbehaved nodes in a network. Centralized node banked with abundant power supply is able to execute a wide range of fault management maintenance.

In distributed approach, a node is allowed to make certain level of decisions before communicating with a central node. This is due to the fact that the more decisions a sensor node can make, the less information needs to be delivered to the central node. This reduces the unnecessary bandwidth utilization and energy expenditure. This means that the central node should not be informed unless there is really a fault occurred in the network. Examples of such approach are: node fault self-reconnaissance on its hardware (including physical malfunction, i.e. sensor, battery, RF transceiver) [26], failure detection via neighbour co-ordination [19], and utilization of Watchdog to detect misbehaving neighbour [27].

Several attempts have been made by many researchers to find effective centralized approaches that can achieve the fault management in WSNs with the minimal exploitation of limited bandwidth and energy. Specifically, Sympathy [25] used a message-flooding approach to pool event data and current states (metrics) from sensor nodes. In order to minimize the number of communication messages that nodes must send as well as to conserve node energy. A Sympathy node can selectively transmit important events to the Sympathy sink node. As a complementary of Sympathy, Staddon et al. [23], to minimize extra traffic overhead further, seek the solution of appending network topology information (i.e. node neighbour list) into node's routing update messages rather than in a separate approach. Based on this approach, the base station can construct the entire network topology by integrating each portion of the network topology information embedded in route update messages and identify the communication faults

within sensor nodes. In addition, some common routing protocols (e.g., SPINs [22]) can also detect failed or misbehaving nodes through routing discovery and update. However, all of these approaches typically require the nodes to send additional messages, which is relatively expensive.

Recently, agent-based approaches have been proposed for efficient data dissemination in WSNs. Furthermore, most of researchers attempted to harness a reverse multicast tree paradigm to achieve their agent-based approaches [3,5,7]. However, none of them take into account the efficient fault reconnaissance due to the resource constraints in WSNs. Based on the reverse multicast tree paradigm, some other researchers proposed a couple of efficient inference schemes [12,13], which to some extent, provided the foundation of fault reconnaissance. For example, the work proposed by Caceres et al. [13], a maximum-likelihood estimator for fault inference is developed. This mechanism is based on link losses observed by multicast receivers and exploits the inherent correlation between such observations to infer the performance of paths. Many researchers attempted to consider fault-reconnaissance inference using unicast measurements [10,11]. The end-to-end measurement scheme proposed by Coates et al. [10] is a case in point. It provides a statistical model and computation framework for network loss inference. However, both types of inference mechanisms (unicast and multicast inference schemes) are based on the traditional IP-based networks. In order to overcome the limitation of the traditional inference schemes, Hartel et al. [14] attempted to address such a drawback in WSNs. Theoretically their approach may lead into converge to local maxima; furthermore, it also requires sufficient observation data (over 300 observation data) to make approximately accurate fault inference. Therefore, while faults occur in a sensor node arbitrarily, the approach proposed in [14] may not be able to accurately infer the fault probability of a sensor node. All of the aforementioned mechanisms focused on the link measurements rather than on the sensor nodes. To address the aforementioned limitations, we propose an intelligent agent – ATLAS to perform fault detection inference in WSNs.

3. Basic mathematical algorithm

Expectation-maximization algorithm for finding maximum likelihood estimates of parameters is proposed by Dempster et al. [20]. This algorithm is applied for computing maximum likelihood estimates from incomplete data. The term "incomplete data" in its general form implies the existence of two sample spaces Φ and Ψ, and many-to-one mapping from Φ to Ψ, as shown in Fig. 1. The observed data y is a realization from Ψ, while the corresponding x in Φ is observed indirectly through y only. More specifically, we assume there is a mapping x $\rightarrow$ y(x) from Φ to Ψ, and x is known only to lie in Φ(y), the subset of Φ determined by the equation y = y(x). Where, y is the observed data and x is referred to it as complete data.

Consider first a family of sampling densities $f\,(x|\alpha)$ depending on parameter α (natural parameters) and derive its corresponding family of sampling densities $g\,(y|\alpha)$. The complete data specification $f\,(...|...)$ is related to the incomplete data specification $g\,(\ldots|\ldots)$, which can be described as:

$$g\,(y|\alpha) = \int_{\Phi(y)} f\,(x|\alpha)\,dx. \tag{1}$$

The EM algorithm is directed at finding a value of α which maximizes $g\,(y|\alpha)$ given an observed y, but it does so making essential use of the associated family $f\,(x|\alpha)$. The EM algorithm is briefly described in the following section, using an application case.

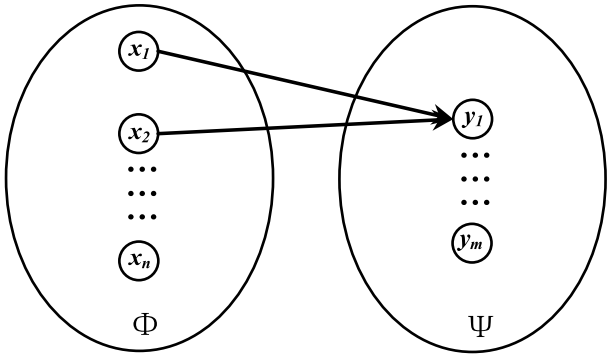

Fig. 1. Many-to-one mapping from the set Φ to set Ψ.

3.1. Application case in glance

In order to give a distinct explanation concerning the EM algorithm, we borrowed a classic application case from [34]. Suppose that in an image pattern-recognition problem, there are two general classes to be distinguished: a class of dark objects and a class of light objects. The class of dark objects may be further subdivided into two shapes: round and square. Using a pattern recognizer, it is desired to determine the probability of a dark object. In this case, let's assume that the objects are known to be trinomially distributed. Let the random variable $\boldsymbol{X}_1$ represent the number of round dark objects, $\boldsymbol{X}_2$ represent the number of square dark objects, and $\boldsymbol{X}_3$ represent the number of light objects. Also assume that we have enough information about the probabilities of the different classes. Therefore, the probability may be written as:

$$\begin{aligned} &P\left(X_1 = x_1, X_2 = x_2, X_3 = x_3|p\right) \\ &= \left(\frac{n!}{x_1!x_2!x_3!}\right)\left(\frac{1}{4}\right)^{x_1}\left(\frac{1}{4}+\frac{p}{4}\right)^{x_2}\left(\frac{1}{2}-\frac{p}{4}\right)^{x_3} \\ &= f\left(x_1, x_2, x_3|p\right) \ . \end{aligned} \tag{2}$$

Where, p is the unknown parameter of the distribution and $n = x_1 + x_2 + x_3$. The notation $f(x_1, x_2, x_3$ |p) is used to indicate the probability function which may be either a probability density function (pdf) or a probability mass function (pmf).

The pattern recognizer can distinguish between different classes of objects whether they are dark or light, but cannot distinguish between the shapes. Let $[y_1, y_2]^T = \boldsymbol{y}$ be the number of dark objects and number of light objects detected, respectively. Thus, $y_1 = x_1 + x_2$ and $y_2 = x_3$. Let's also assume that Y_1, and Y_2 are the corresponding random variables. There is a many-to-one mapping between $\{x_1, x_2\}$ and $\{y_l\}$. For example, if $y_1 = 3$, there is no way to tell from the measurements whether $x_1 = 1$ and $x_2 = 2$ or $x_1 = 2$ and $x_2 = 1$. Therefore, the EM algorithm is designed for addressing such problem with many-to-one mappings using the following probability function.

$$\begin{aligned} P\left(Y_1 = y_1|p\right) &= \binom{n}{y_1}\left(\frac{1}{2}+\frac{p}{4}\right)^{y_1}\left(\frac{1}{2}-\frac{p}{4}\right)^{n-y_1} \\ &= g\left(y_1|p\right) \ . \end{aligned} \tag{3}$$

The symbol g is used to indicate the probability function for the observed data. From the observation of y_1 and y_2, we can compute the maximum likelihood estimate of p, using the following equation:

$$p_{ML} = \arg\max_p g\left(Y_1 = y_1|p\right). \tag{4}$$

Where "arg max" means the value that maximizes the function. In this example, it would be a simple matter to determine a maximum likelihood estimate of p. In more interesting problems, however, such straightforward estimation is not possible. In the interest of introducing the EM algorithm, we will not take the direct approach to the maximum likelihood estimate. Taking the logarithm of the likelihood often simplifies the maximization and yields equivalent results since log is an increasing function, so Eq. (4) can be rewritten as:

$$p_{ML} = \arg\max_p \log \binom{n}{y_1} \left(\frac{1}{2} + \frac{p}{4}\right)^{y_1} \left(\frac{1}{2} - \frac{p}{4}\right)^{n-y_1}. \tag{5}$$

Based on EM algorithm, even though we do not know x_1 and x_2, the knowledge of the underlying distribution f (x_1, x_2, $x_3|$ p) can be used to determine an estimate for p. This is done by first estimating the underlying data, then using these data to update current estimate of the parameter. This is repeated until convergence. Let $p^{[k]}$ indicate the estimate of p after the k^{th} iteration ($k = 1, 2, 3, ...$). An initial parameter value $p^{[0]}$ is assumed. The algorithm consists of two major steps, namely: expectation step and maximization step.

Expectation Step (E-step). In this step, the expected value of x data is computed using the current estimate of the parameter and the observed data. In the current example, the expected value of x_l, given the measurement y_1 and based upon the current estimate of the parameter, may be computed as follows:

$$x_1^{[k+1]} = \boldsymbol{E}[x_1| \quad y_1, p^{[k]}].$$

$$x_1^{[k+1]} = y_1 \frac{\frac{1}{4}}{\frac{1}{2} + \frac{p^{[k]}}{2}}.$$

Similarly for x_2 and x_3,

$$\begin{aligned} x_2^{[k+1]} = E\left[x_1|y_1, p^{[k]}\right] &= y_1 \frac{\frac{1}{4} + \frac{p^{[k]}}{2}}{\frac{1}{2} + \frac{p^{[k]}}{2}}. \\ x_3 &= y_3. \end{aligned} \tag{6}$$

Maximization Step (M-step). In this step, a maximum likelihood estimate of the parameter is determined using the data from the expectation step as if it were actually measured data.

In this example, with $x_l^{[k+l]}$, $x_2^{[k+1]}$ and $\boldsymbol{x}_3$ available, the maximum likelihood estimate of the parameter is obtained by taking the derivative of *log* f ($\boldsymbol{x}_l^{[k+l]}$, $\boldsymbol{x}_2^{[k+1]}$, $\boldsymbol{x}_3|p$) with respect to p, set it to zero, and solving for p as follows:

$$\begin{aligned} 0 &= \frac{d}{dp} \log f\left(x_1^{[k+1]}, x_2^{[k+1]}, x_3|p\right) \\ \Rightarrow p^{[k+1]} &= \frac{2x_2^{[k+1]} - x_3}{x_2^{[k+1]} + x_3}. \end{aligned} \tag{7}$$

Table 1
Results of the Em Algorithm for pattern recognition problem

k	$x_1^{[k]}$	$x_2^{[k]}$	$p^{[k]}$
1	31.500000	31.500000	0.379562
2	26.475460	36.524540	0.490300
3	25.298157	37.701843	0.514093
4	25.058740	37.941260	0.518840
5	25.011514	37.941260	0.519773
6	25.002255	37.997745	0.519956
7	25.000441	37.999559	0.519991
8	25.000086	37.999914	0.519998
9	25.000017	37.999983	0.520000
10	25.000003	37.999997	0.520000

Now, we need to compute the estimate of $\boldsymbol{x}_1^{[k+1]}$. The estimate $\boldsymbol{x}_1^{[k+1]}$ is not used in Eq. (7). Thus, there is no need to compute it, for this example. The EM algorithm consists of E-step and M-step until convergence. Intermediate computation and storage may be eliminated by combining E-step equation and M-step one to obtain a one-step update.

Using a numerical example, suppose that the true parameter is $p = 0.5$, $n = 100$ samples, and $y_1 = 100$. The true values respectively of x_1 and x_2 are 25 and 38, but this is unknown to the algorithm. Table 1 illustrates the results of the computed algorithm, starting from $p^{[0]} = 0$ which is similar to [20].

4. ATLAS

This section first provides an intelligent agent-ATLAS- with failure Inference capacity. Furthermore, we provide terminology and a fundamental mathematical model used in the inference engine. Finally, we propose the core of ATLAS – a failure inference engine which is formulated as a maximum-likelihood estimate problem.

4.1. An Intelligent agent with fault inference

Most of agent-based distributed systems consist of two types of agents based on their functionalities, i.e. stationary and mobile agents. In this paper, ATLAS is a stationary agent with an embedded fault inference engine. The architecture of ATLAS (shown in Fig. 1) with fault-reconnaissance functionality is based on the agent model described in [1]. Generally, ATLAS resides on a central node equipped with surplus power supply (Base station).

As demonstrated in Fig. 2, the architecture of ATLAS consists of knowledge base and executable components. The knowledge base contains the information about the WSNs environment such as packet-store table and topology structure. The packet-store table is composed of the received packets from target sensor nodes; ATLAS maintains the topology structure of reverse multicast trees as well. For some applications in WSNs, an application may generate several reverse multicast trees to achieve a better quality of service. The learning component provides ATLAS with the capability of monitoring observation data stored in its knowledge base. Based on the observations, ATLAS collaborates with various routing protocols and thus dynamically selects a routing protocol that provides the best performance benefits for WSNs. The scheduler component provides ATLAS with a time agenda to start or stop certain activities such as monitoring and aggregating data. In this architecture, the problem solver

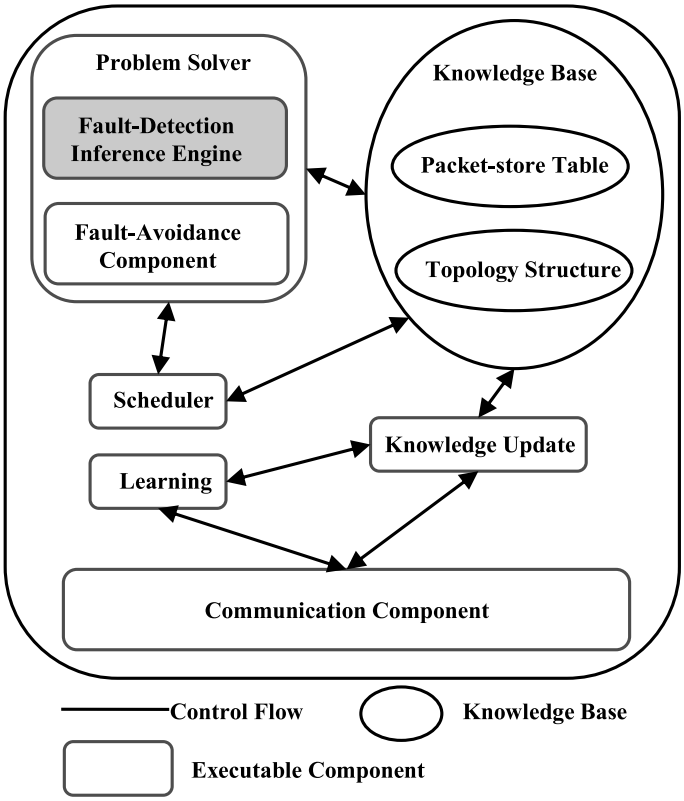

Fig. 2. The architecture of ATLAS.

is an intelligent component of ATLAS. It includes failure-reconnaissance inference engine (which is the core of ATLAS) and fault-avoidance component. Fault-reconnaissance inference engine harnesses the information in knowledge base to infer the working states of sensor nodes in a reverse multicast tree. And then, the inferred consequences assist fault-avoidance component in choosing and switching to the appropriate routing strategy among various routing protocols.

4.2. Terminology and mathematical model

Failure inference engine adopts a reverse-multicast-tree mechanism. This scheme is based on the fundamental of data aggregation. In the inference engine, we formulate a reverse multicast tree as a mathematical model $T := (V, E)$, where T is the reverse multicast tree that is subject to the following conditions:

V is a set, whose elements are called wireless sensor nodes; E is a set of pairs of distinct sensor nodes called communication link. Obviously, the node set $V = \{n_1, n_2, n_3, \ldots, S\}$ also contains sink S. And a communication link from node i to node j is represented by $e_{i,j} \in V \times V$.

For our modeling purposes in this paper, we make a simplifying assumption – the communication link between node i and j is always symmetrically bidirectional link such that $e_{i,j} \equiv e_{j,i}$. Further, let $c(n_i)$ and $d(n_i)$ denote the child set and the descendent set of node i, respectively. In reverse multicast tree paradigm, each node except sink has a unique parent node denoted by $p(n_i)$ where $n_i \in V \wedge n_i \notin \{S\}$. Finally, denote the set of leaf nodes in the tree by $l(T)$. That is $l\,(T) = \{n_k \in V \wedge c\,(n_k) = \phi\}$.

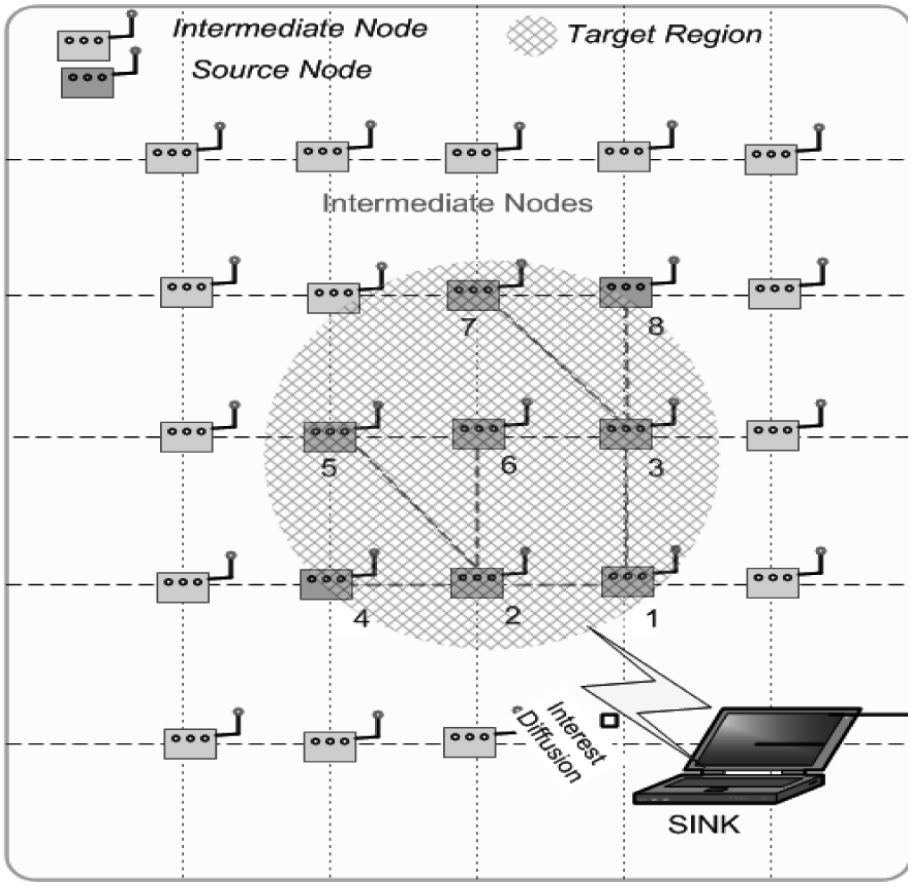

Fig. 3. Eight-node reserve multicast tree in a square grid topology.

For example, in the tree depicted in Fig. 3, for node 2 the proper descendant nodes are 4, 5 and 6 and the parent node is node 1 i.e., $d(n_2) = \{4, 5, 6\}$ and $p(n_2) = \{1\}$ respectively. In this whole tree, the leaf nodes are 4, 5, 6, 7, 8 denoted as $l(T) = \{4, 5, 6, 7, 8\}$. In wireless sensor networks, sensor nodes are prone to failures [15]. Therefore, each sensor node is modeled by an independent Bernoulli process [14].

Associated with every node $n_i \in V$, a success probability α_{n_i} represents the packets sent from node j ($\forall j\ n_j \in c(n_i)$) to node i that are received successfully. On the contrary, the fault probability of node i denotes $\beta_{n_i}(\alpha_{n_i} + \beta_{n_i} = 1)$. Finally, let $\alpha = \langle \alpha_{n_1}, \alpha_{n_2}, \alpha_{n_3}, ... \rangle$ denote the set of success probabilities for all sensor nodes. The packets transmitted through a reverse multicast tree, and the packets received by sink construct a sample space $\Omega = \langle x^1, x^2, x^3, ..., x^m \rangle$.

A possible outcome x^i in data collection round i is composed of a sequence of partial nodes in reverse multicast tree. Based on the reverse-multicast-tree paradigm, each node aggregates its data with that of its descendants. Therefore, if the sink does not receive data from a node, it also does not gather data from all the descendents of that node. For example in Fig. 3, the sink receives an aggregated packet x^i with the sequence of nodes <1, 2, 3, 5, 6, 7>. Furthermore, each round of data collection is considered a random trial. Since the outcome of each random trial denotes a record of which nodes the sink received data from that round, we consider a discrete random variable X. Let $p(X = x;\ \alpha) = p(x;\ \alpha)$ denotes the probability distribution for a given set of node success probabilities α. Also, assume an m-round

trial, the probability of observing x^1, ..., x^m in m data collection rounds is denoted by:

$$p\left(x^1,\ ...,x^m\right)=\prod_{i=1}^{m}p\left(x^i;\alpha\right). \tag{8}$$

4.3. An intelligent agent with fault inference

Failure inference engine adopts a reverse-multicast-tree mechanism. From previous section, $p\left(x;\ \alpha\right)$ is the probability mass function of a single outcome x. Nevertheless, the success probability α is an unknown parameter and a quantity we expect to estimate. So, our proposed inference engine utilizes maximum likelihood estimation to infer parameter of underlying probability distribution from a collected packet set. Assume that N represents packet collection rounds, the packet sequence x^1, ..., x^N observed at the sink comprises a random sample space (due to malfunction of WSNs, x^i may be an empty node sequence). Based on the sample space of N values, we choose $\widehat{\alpha}$ to maximize $p\left(x^1,\ ...,\ x^N;\widehat{\alpha}\right)$, i.e., $\widehat{\alpha}=\arg\ \max p\left(x^1,...,x^N;\alpha\right)$ which is maximum likelihood estimation.

In our approach however, we require a sophisticated estimation technique due to the fact that the current knowledge base in inference engine (sample space) is incomplete. Therefore, as an efficient iterative computing approach, the expectation maximization algorithm is incorporated in the fault-detection inference engine. This algorithm makes inference about parameter α using incomplete knowledge base. It should be noted that "incomplete knowledge base" is equivalent to "complete observed data" in expectation maximization algorithm.

It is more convenient to operate the log-likelihood function of the complete data. Let $\{x^1, x^2, \ldots, x^N\}$ be the set of complete data. Then, log-likelihood equation can be described as follows:

$$L\left(x^1,x^2,...,x^N;\alpha\right)=\log\ p\left(x^1,x^2,...,x^N;\alpha\right). \tag{9}$$

Where, the joint distribution probability is divided into two parts: leaf and non-leaf nodes. The expression for leaf nodes is:

$$p\left(x^1,x^2,...,x^N;\alpha\right)=\prod_{i=1}^{N}\alpha_{n_i}^{N_{n_i\to n_i}}\cdot\beta_{n_i}^{N-N_{n_i\to n_i}}. \tag{10}$$

Where $N_{n_i\to n_i}$ denotes the number of data collection rounds that node i successfully sent out packets. For the non-leaf nodes, it is described as:

$$p\left(x^1,x^2,...,x^N;\alpha\right)=\prod_{i=1}^{N}\alpha_{n_i}^{N_{d(n_i)\to n_i}+N_{n_i\to n_i}}\cdot\beta_{n_i}^{N+N_{c(n_i)\to n_i}-N_{d(n_i)\to n_i}}. \tag{11}$$

Where $N_{d(n_i)\to n_i}$ denotes the number of data collection rounds that node i successfully received packets from its descendents, and $N_{c(n_i)\to n_i}$ represents the number of data round that node i successfully received child nodes' packets.

To maximize the Eqs (10) and (11), we take the logarithm function to simplify maximization likelihood estimation. The results are shown in Eqs (12) and (13).

$$L\left(x^1,x^2,...,x^N;\alpha\right)=\sum_{n_i\in l(T)}\left(N_{n_i\to n_i}\log\alpha_{n_i}+N\log\beta_{n_i}-N_{n_i\to n_i}\log\beta_{n_i}\right). \tag{12}$$

$$L\left(x^1, x^2, ..., x^N; \alpha\right) = \sum_{n_i \notin l(T)} ((N_{d(n_i)\to n_i + N_{n_i \to n_i}}) \log \alpha_{n_i} + (N + N_{c(n_i)\to n_i} - N_{d(n_i)\to n_i}) \log \beta_{n_i}) . \tag{13}$$

For maximizing Eqs (12) and (13), we utilize Calculus Theory that leads to Eq. (14).

$$\frac{d}{d\,\alpha} L\left(x^1, x^2, ..., x^N; \alpha\right) = 0. \tag{14}$$

By combining Eqs (12), (13) and (14), we solve the equations and produce the parameter Eqs (15) and (16) as we expected.

$$\alpha_i = (N_{n_i \to n_i}/N) \cdot 100\% \quad n_i \in l\,(T)\ . \tag{15}$$

$$\alpha_i = \frac{\sum \left(N_{d(n_i)\to n_i} + N_{n_i \to n_i}\right)}{\sum \left(N + N_{c(n_i)\to n_i}\right)} \cdot 100\% \quad n_i \notin l\,(T)\ . \tag{16}$$

For example, in the square grid topology depicted in Fig. 4, the success probability of node 3 is formulated as follows:

$$\alpha_3 = \frac{(N_{n3\to n3} + N_{n5\to n3} + N_{n6\to n3} + N_{n4\to n3})}{(N + N_{n6\to n6} + N_{n5\to n5} + N_{n4\to n4})}$$

However, a complete knowledge base requires each sensor node to record and transmit packets that it received from descendents in each round. Unfortunately, this solution not only wastes constrained bandwidth but also consumes energy resources. Therefore, an efficient inference engine, which learns from the collected packets at sink, is utilized to complete knowledge base.

To complete the knowledge base, we adopt expectation maximization algorithm, using Eqs (8) and (9). This algorithm, in this case, consists of five steps. In step 1, initializing fault-detection parameter vector $\widehat{\alpha}^{[0]}$ and $\widehat{\beta}^{[0]}$. In step 2, the conditional expectation is used under the parameter vector $\widehat{\alpha}^{[k]}$ and $\widehat{\beta}^{[k]}$ to estimate incomplete data set $N^{[k+1]}_{d(n_i)\to n_i}$, $N^{[k+1]}_{c(n_i)\to n_i}$ and $N^{[k+1]}_{n_i \to n_i}$. In step 3, maximizing the expected log-likelihood and update the parameter vectors $\widehat{\alpha}^{[k+1]}$ and $\widehat{\beta}^{[k+1]}$. In step 4, repeating steps 2 and 3 until the algorithm converges to local maxima. In step 5, restarting with new initial parameter vector $\widehat{\alpha}^{[0]}$ and $\widehat{\beta}^{[0]}$ since the likelihood surface may not guarantee convex. If the likelihood is not convex, then the probability may converge to local maxima. The entire internal inference process described as a Non-deterministic Finite Accepter (NFA) as shown in Error! Reference source not found..

The iterative steps 2 and 3 require computing $N^{[k+1]}_{d(n_i)\to n_i}$, $N^{[k+1]}_{c(n_i)\to n_i}$ and $N^{[k+1]}_{n_i \to n_i}$ from the knowledge base, i.e., completely observed data set $\{x^1, x^2, \ldots, x^N\}$. The computation process is based on Chapman-Kolmogorov functional equation. The corresponding Markov chain model is shown in 5.

Therefore, the Chapman-Kolmogorov equation based on the matrix of transition probabilities can be represented by Eqs (17) and (18).

$$\begin{bmatrix} \alpha_{n_i} & \beta_{n_i} \\ \beta_{n_i} & \alpha_{n_i} \end{bmatrix} \cdot \begin{bmatrix} \alpha_{n_j} & \beta_{n_j} \\ \beta_{n_j} & \alpha_{n_j} \end{bmatrix} \cdot \ldots \cdot \begin{bmatrix} \alpha_{n_m} & \beta_{n_m} \\ \beta_{n_m} & \alpha_{n_m} \end{bmatrix} = \begin{bmatrix} \alpha & \beta \\ \beta & \alpha \end{bmatrix} \tag{17}$$

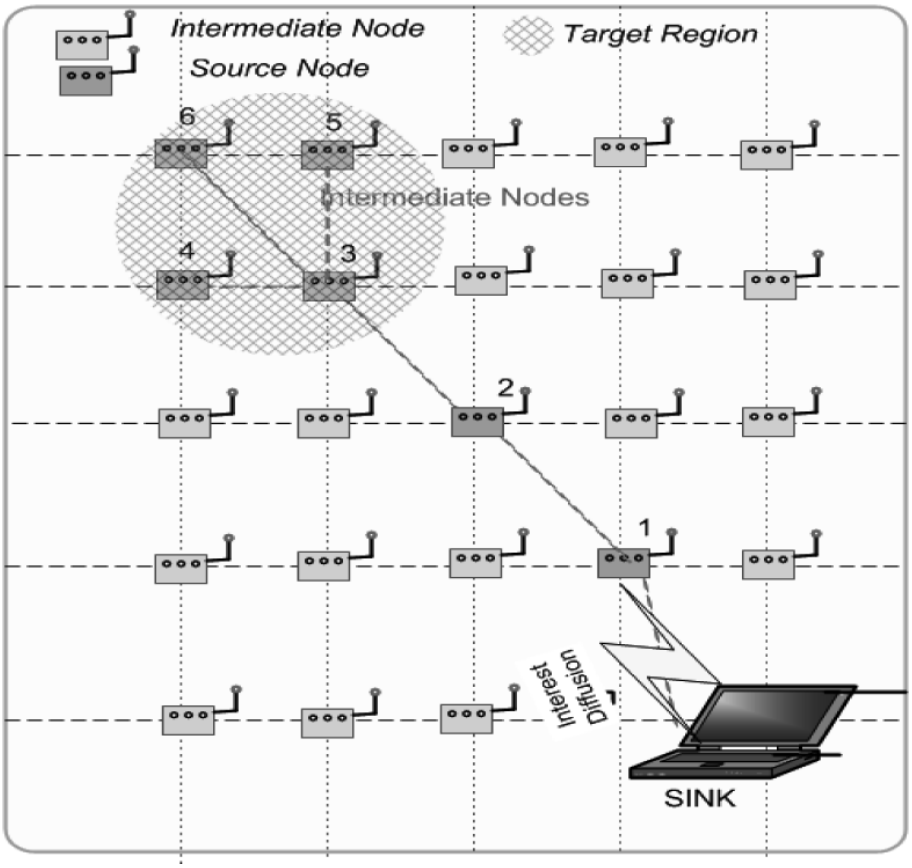

Fig. 4. Six-node reserve multicast tree in a square grid topology.

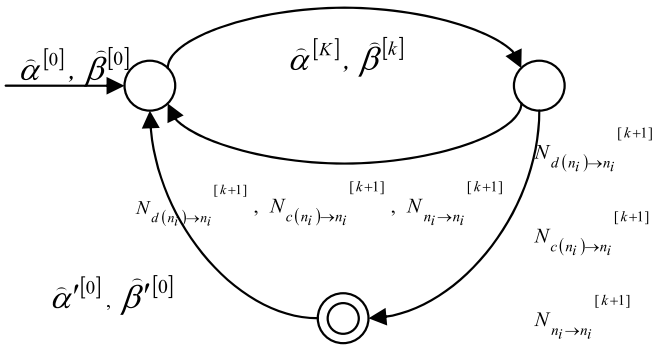

Fig. 5. Fault-detection inference engine described in NFA.

For leaf nodes,

$$N_{n_j\to n_i}^{[k+1]} = N_{n_j} + \left(N - N_{n_j}\right)\cdot \beta_{n_i}^{[k]}, \quad n_j \equiv n_i. \tag{18}$$

Where, N_{n_j} denotes the number of packet collection rounds with node identification j observed by sink.

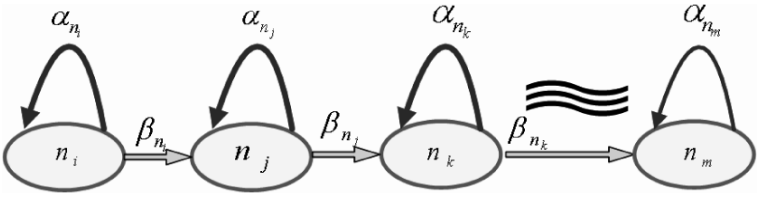

Fig. 6. Morkov chain model.

Fig. 7. A snapshot of Testbed at University of Colorado.

Similarly, this can be used to represent non-leaf nodes, using Eq. (19).

$$N_{n_j \to n_i}^{[k+1]} = N_{n_j} + \left(N - N_{n_j}\right) \cdot \beta \quad n_j \in c\left(n_i\right) \vee n_j \in d\left(n_i\right). \tag{19}$$

Combining E-Step Eqs (15) and (16) with M-Step Eqs (18) and (19), we iteratively compute a set of success probability values. Next section will show a 6-nodes reverse cast tree experiment in detail and analysis of our experiment results.

5. Experimental results

This section discusses our experimental results and analyzes the properties of inference engine in ATLAS. In order to validate the correctness and effectiveness of ATLAS, we implement the prototype of ATLAS on top of the Mantis Operating System (MOS). MOS is an open source, multithreaded operating system developed at University of Colorado at Boulder for use on wireless sensor networking platforms.

For our experiments, we deployed an indoor testbed of TelosB motes as shown in Fig. 7. The TelosB platform was developed at University of California at Berkeley and is marketed and sold by Moteiv and Crossbow. The radio used by the TelosB is the chipcon CC2420, which is an 802.15.4 compliant device, has a data rate of 250Kbps, and operates in the 2.4 GHz ISM band. The mote uses an 8 MHz TI MSP430 processor and has 1 MB of external flash.

The entire experimental scenario we adopted consists of 30 sensor nodes. In the 30-node topology, 7 nodes generated a reverse multicast tree. During each collection round, the sink cached received packets

Table 2
Results of the iterative computation based on equal fault probability

k	$n_1^{[k]}$	$n_2^{[k]}$	$n_3^{[k]}$
1	98.1507%	96.8647%	98.1432%
2	98.8739%	97.6735%	96.3148%
3	98.8795%	97.5608%	96.4541%
4	98.8774%	97.5733%	96.4631%
5	98.8776%	97.5728%	96.4570%
6	98.8776%	97.5727%	96.4587%
7	98.8776%	97.5727%	96.4583%
8	98.8776%	97.5727%	96.4584%
k	$n_4^{[k]}$	$n_5^{[k]}$	$n_6^{[k]}$
1	92.5000%	91.6250%	92.8750%
2	86.1250%	84.6528%	86.7653%
3	87.0812%	85.8206%	87.6359%
4	86.9378%	85.6250%	87.5118%
5	86.9593%	85.6578%	87.5295%
6	86.9561%	85.6523%	87.5270%
7	86.9565%	85.6532%	87.5273%
8	86.9565%	85.6530%	87.5273%

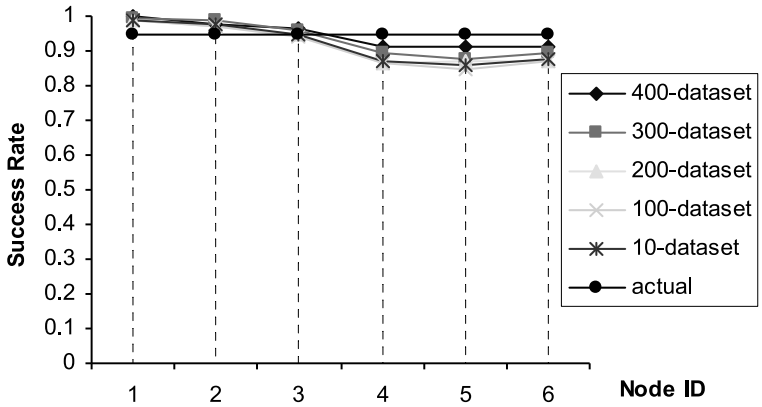

Fig. 8. Inference engine under scenario 1.

from its descendents. After 400 data collection round, the fault-inference engine was triggered at sink. For comparability, the inference engine analyzed observed dataset based on various sub-datasets.

The first experiment is conducted on an equal fault probability (Scenario 1). In this scenario, we assumed the normal success probability of sensor node to be 95%. The 5% of failure probability could be due to signal loss, interference, radio initialization, etc. Thus, the initial parameter $\widehat{\beta}^{[0]}$ is chosen as a set of different values in the interval [0,1] to avoid halting on local maxima. Table 2 illustrates the detailed results based on the iterative computation for equal fault probability scenario.

Furthermore, the failure probabilities as well as the absolute error for each node are reflected in Figs 8 and 9 respectively.

In Fig. 8, it was observed that inferred fault probabilities of the nodes are increasingly accurate along with the increase in the volume of dataset.

Furthermore, it is obvious from the plot of standard deviation shown in Fig. 10 that the inference

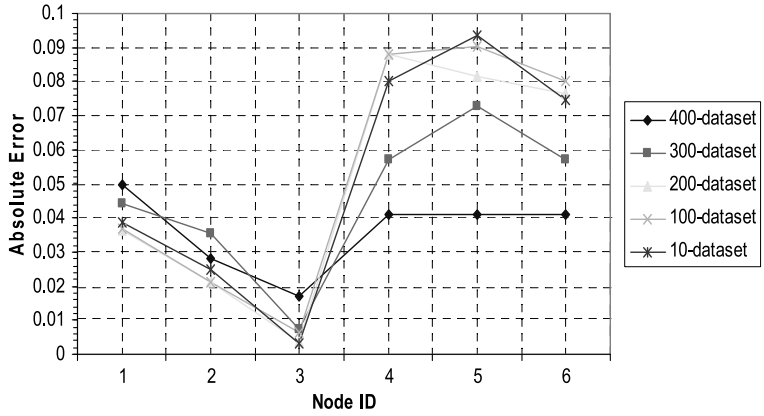

Fig. 9. Absolute error under scenario 1.

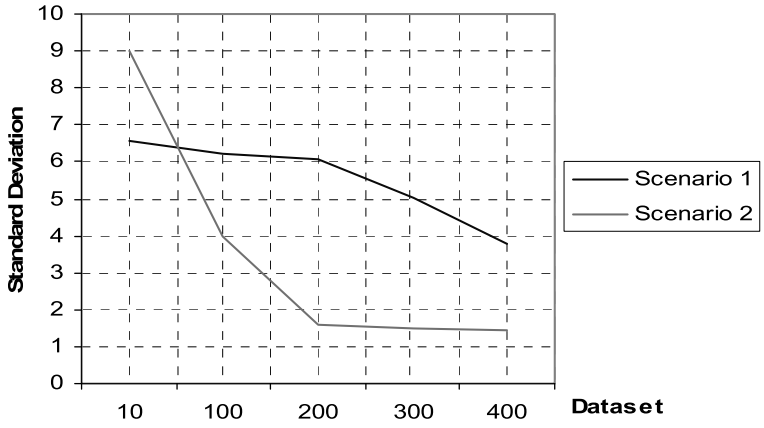

Fig. 10. Standard deviation under disparate scenario.

process is able to accurately reflect the real-time states of nodes.

The second experiment was conducted on a high-frequent-fault node (scenario 2). Faults kept occurring frequently at node 2.

We assume fault probability is set to 40%, while the other nodes ran normally with the same fault probability as that of nodes in first scenario. Based on the experimental results shown in Figs 11 and 12, we may make a reasonable inference that node 2 may experience unexpected high-frequency faults while the rest of nodes 1, 3, 4, 5 and 6 kept running normally.

The results of the reasoning using Eqs (15), (16), (18) and (19) are listed in Table 3.

The volume of dataset in this scenario has great impact on the property of inference engine. More importantly, it can be seen that the standard deviation of the inference engine shown in Fig. 10 drastically decreases along with an increase in dataset that was utilized to infer fault probabilities of nodes. Thus, the proposed inference mechanism is able to successfully target fault-experience nodes.

Furthermore, we study the reliability of maximum-likelihood estimator. Based on the same dataset generated by the scenario 2, the maximum-likelihood estimator performed poor inference result, as shown in Fig. 13.

With the comparison between our proposed mechanism and maximum-likelihood estimator, we observe

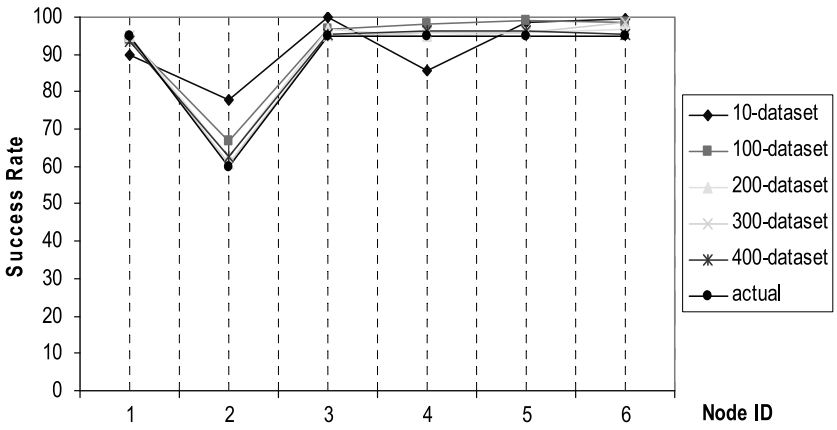

Fig. 11. Inference Engine under scenario 2.

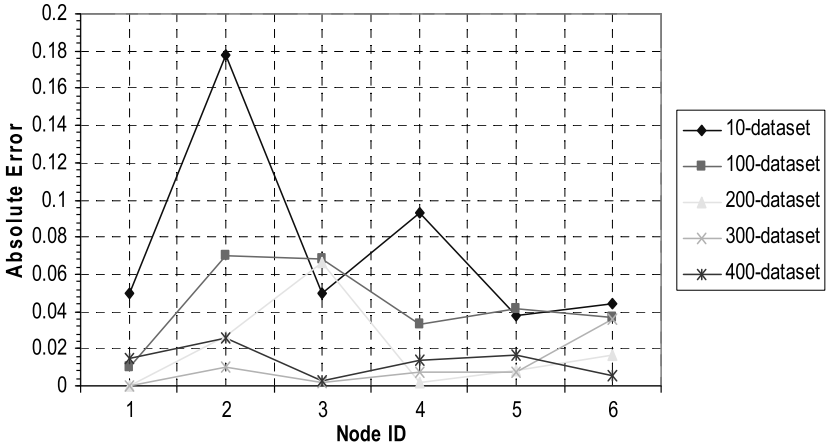

Fig. 12. Absolute Error under scenario 2.

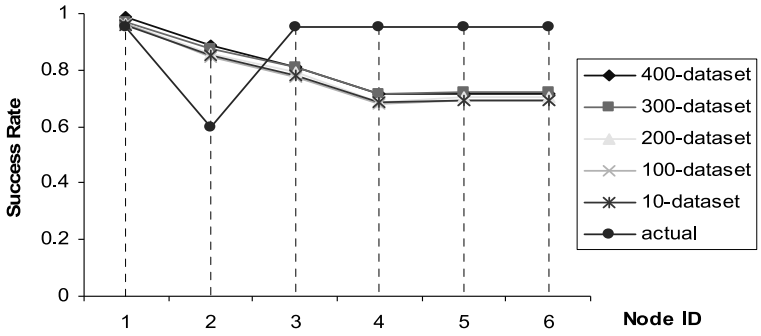

Fig. 13. Maximum-likelihood estimator under scenario 2.

Table 3
Results of the iterative computation based on high-frequent-fault node.

k	$n_1^{[k]}$	$n_2^{[k]}$	$n_3^{[k]}$
1	95.3451%	67.6250%	86.4805%
2	96.8658%	62.7234%	94.8478%
3	96.6865%	62.2154%	94.5067%
4	96.5479%	62.4503%	96.9278%
5	96.6532%	62.3417%	94.2011%
6	96.5946%	62.3919%	94.5636%
7	96.6240%	62.3687%	94.3865%
8	96.6095%	62.3794%	94.4729%
k	$n_4^{[k]}$	$n_5^{[k]}$	$n_6^{[k]}$
1	89.1566%	86.0250%	99.0250%
2	96.9956%	95.7234%	95.4564%
3	96.6219%	96.2154%	95.2655%
4	96.9259%	96.4503%	95.3500%
5	96.2501%	96.3417%	95.3126%
6	96.6086%	96.3919%	95.3291%
7	96.9208%	96.3687%	95.3218%
8	96.7695%	96.3794%	95.3250%

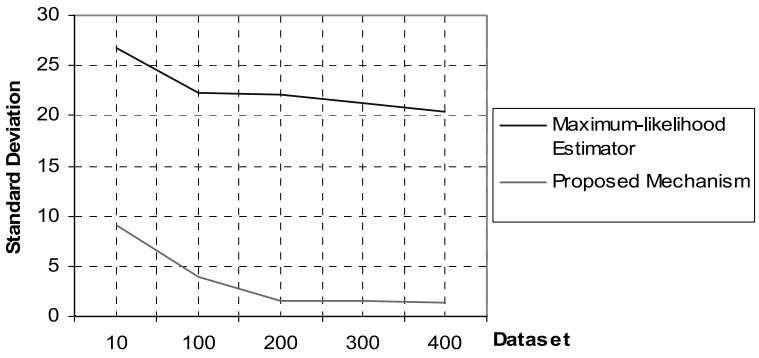

Fig. 14. Standard deviation under disparate inference mechanisms.

from Fig. 14 that our proposed mechanism with the lower standard deviation would satisfy the objective of inference detection. In contrast, maximum-likelihood estimator would not be utilized in WSNs environment.

6. Conclusions and future work

This paper identified the principal faults experienced by WSNs and summarized the existed fault inference mechanisms. This paper presented an intelligent agent – ATLAS with a fault inference problem solving engine as a response to the drawbacks of current inference mechanisms and the features of WSNs. The inference engine is modeled by maximization-likelihood estimation via unobservable data set. In accordance with theoretical computation, the proposed inference engine embedded in ATLAS is abstracted as a nondeterministic finite accepter. This paper provided experimental results that showed the inference engine of ATLAS achieves substantial energy gain by containing the impact of malfunctioning sensor nodes.

In the future, we plan to study and establish the relationship between ATLAS and energy consumption of WSNs. Moreover, we plan to augment our effort to expand our approach to a higher level of fault management, i.e. involving fault diagnosis with minimum exploitation of WSNs resources.

References

[1] E. Shakshuki, H. Ghenniwa and M. Kamel, Agent-based system architecture for dynamic and open environments, *International Journal of Information Technology and Decision Making* **2**(1) (2002), 105–133.

[2] I. Stojmenovic, Localized network layer protocols in sensor networks based on optimizing cost over progress ratio, *IEEE Network* **20**(1) (2006), 21–27.

[3] M. Chen, T. Kwon, Y. Yuan, Y. Choi and V.C.M. Leung, Mobile Agent-Based Directed Diffusion in wireless Sensor Networks, EURASIP Journal on Advances in Signal Processing, Article ID 36871, 2007.

[4] E. Shakshuki, H. Malik and T. Sheltami, Multi-agent Based Clustering Approach to Wireless Sensor Networks, *International Journal of Wireless and Mobile Computing* (*IJWMC*), *Inderscience Publisher* **3**(3) (2009), 165–176.

[5] E. Shakshuki, X. Xing and H. Malik, Mobile Agent for Efficient Routing Among Source Nodes in Wireless Sensor Networks, The 3rd International Conference on Autonomic and Autonomous Systems (ICAS), IEEE Computer Society, June 19–25, Athens, Greece, 2007.

[6] H. Qi, Y. Xu and X. Wang, Mobile-agent-based Collaborative Signal and Information Processing in Sensor Networks, *Proceeding of the IEEE* **91**(8) (2003), 1172–1183.

[7] Q. Wu, N.S.V. Rao, J. Barhen et al., On computing mobile agent routes for data fusion in distributed sensor networks, *IEEE Transactions on Knowledge and Data Engineering* **16**(6) (2004), 740–753.

[8] C.-L. Fok, G.-C. Roman and C. Lu. Mobile Agent Middleware for Sensor Networks: An Application Case Study, In Proceedings of the 4th International Conference on Information Processing in Sensor Networks (IPSN'05), Los Angeles, California, April 25–27, pp. 382–387, 2005.

[9] D. Massaguer, C.-L. Fok, N. Venkatasubramanian, G.-C. Roman and C. Lu, Exploring sensor networks using mobile agents, AAMAS, 2006, 323–325.

[10] M. Coates and R. Nowak, Network Loss Inference Using Unicast End-to-End Measurement, in ITC Seminar on IP Traffic, Measurement and Modelling, September 2000.

[11] N. Duffield, F.L. Presti, V. Paxson and D. Towsley, Inferring Link Loss Using Striped Unicast Probes, in Proceedings of IEEE INFOCOM 2001.

[12] S. Ratnasamy and S. McCanne, Inference of Multicast Routing Trees and Bottleneck Bandwidths using End-to-End Measurements, in Proceedings IEEE INFOCOM 1999.

[13] R. Caceres, N.G. Duffield, J. Horowitz and D. Towsley, Multicast-based inference of network-internal loss characteristics, *IEEE Trans Inform Theory* **45** (1999), 2462–2480.

[14] G. Hartl and B. Li, Loss Inference in Wireless Sensor Networks based on Data Aggregation, in the Proceedings of the Third IEEE/ACM International Symposium on Information Processing in Sensor Networks (IPSN 2004), pp. 396–404, Berkeley, California, April 26–27, 2004.

[15] F. Koushanfar, M. Potkonjak and A. Sangiovanni-Vincentelli, Fault-Tolerance in Sensor Networks, Book chapter, in: *Handbook of Sensor Networks*, I. Mahgoub and M. Ilyas, eds, CRC press, Section VIII, no. 36, 2004.

[16] C. Hsin and M. Liu, A Distributed Monitoring Mechanism for Wireless Sensor Networks, in the 3rd workshop on Wireless Security, ACM Press, 2002.

[17] C. Hsin and M. Liu, Self-monitoring of Wireless Sensor Networks, *Computer Communications* **29** (2005), 462–478.

[18] M. Ding, D. Chen, K. Xing and X. Cheng, Localized Fault-Tolerant Event Boundary Detection in Sensor Networks, in Processing of INFOCOM 2005.

[19] J. Chen, S. Kher and A. Somani, *Distributed Fault Detection of Wireless Sensor Networks*, in Processing of DIWANS'06, Los Angeles, USA, ACM Press, 2006.

[20] A.P. Dempster, N.M. Laird and D.B. Rubin, Maximum-likelihood from incomplete data via the EM algorithm, *J Roy Stat Soc* **39** (1977), 1–38.

[21] M. Yu, H. Mokhtar and M. Merabti, *A Survey of Network Management Architecture in Wireless Sensor Network*, in Proceedings of the Sixth Annual Post-Graduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting, 2006.

[22] A. Perrig, R. Szewczyk, V. Wen, D. Culler and J.D. Tygar, *SPINS: Security Protocols for Sensor Networks*, in Processing of ACM MobiCom'01 Rome, Italy ACM Press, 2001.

[23] J. Staddon, D. Balfanz and G. Durfee, *Efficient Tracing of Failed Nodes in Sensor Networks*, in Processing of First ACM International Workshop on Wireless Sensor Networks and Applications, Altanta, GA, USA, ACM, 2002.

[24] W.-L. Lee, A. Datta and R. Cardell-Oliver, *WinMS: Wireless Sensor Network-Management System, An Adaptive Policy-Based Management for Wireless Sensor Networks*, UWA, aUSTRALIA, 2006.
[25] N. Ramanathan, K. Chang, R. Kapur, L. Girod, E. Kohler and D. Estrin, *Sympathy for the Sensor Network Debugger*, In Processing of the 3rd Embedded networked sensor systems, San Diego, USA, ACM Press, 2005.
[26] F. koushanfar, M. Potkonjak and A. Sangiovanni- Vincentelli, Fault Tolerance Techniques for Wireless Ad Hoc Sensor Networks, 2000.
[27] S. Marti, T.J. Giuli, K. Lai and M. Baker, *Mitigating Routing Misbehavior in Mobile Ad Hoc Networks*, in Processing of the 6th International Conference on Mobile Computing and Networking, Boston, Massachusetts, USA, ACM, 2000.
[28] RF Monolithics Inc. TR1001 868.35 MHz Hybrid Transceiver Data Sheet, Aug. 2001.
[29] J.-H. Huang, S. Amjad and S. Mishra, *CenWits: A Sensor-Based Loosely Coupled Search and Rescue System using Witnesses*, in Proceedings of the 3rd international conference on Embedded networked sensor systems (Sensys'05), San Diego, USA, ACM, 2005.
[30] C. Hartung, C. Seielstad, S. Holbrook and R. Han, *FireWxNet: A Multi- Tiered Portable Wireless System for Monitoring Weather Conditions in Wildland Fire Environments*, Fourth International Conference on Mobile Systems, Applications, and Services (MobiSys), 2006, pp. 28–41.
[31] F. Erik, Y. Golen and S. Nirmala, An Evolutionary Approach to Underwater Sensor Deployment, *IJCIS* **2–3** (2009), 184–201.
[32] Gowrishankar and P.S. Satyanarayana, Neural Network Based Traffic Prediction for Wireless Data Networks, *IJCIS* **1–4** (2008), 379–389.
[33] E. Shakshuki, X. Xing and T. Sheltami, *An Intelligent Agent for Fault Reconnaissance in Sensor Networks*, the 11th International Conference on Information Integration and Web-based Applications & Services, pp. 137–144, December 14–16, Kuala Lumpur, Malaysia, 2009.
[34] T.K. Moon, The expectation-maximization algorithm, Signal Processing Magazine, *IEEE* **13**(6) (November 1996), 47–60,
[35] E. Shurkova, R. Ishak, S. Olariu and S. Salleh, Emergent Behavior in Massively-deployed Sensor Networks, *Mobile Information Systems* **4**(4) (2008), 313–331.
[36] S. Mahfoudh and P. Minet, Maximization of Energy Efficiency in Wireless Ad hoc and Sensor Networks with SERENA, *Mobile Information Systems* **5**(1) (2009), 33–52.
[37] S. Kumar and S. Park, Probability Model for Data Redundancy Detection in Sensor Networks, *Mobile Information Systems* **5**(2) (2009), 195–204.

Elhadi M. Shakshuki is a professor at the Jodrey School of Computer Science at Acadia University, Canada. He is the founder and the head of the Cooperative Intelligent Distributed Systems Group at the Computer Science Department, Acadia University. He received the BSc degree in computer engineering in 1984 from El-Fateh University, and the MSc and PhD degrees in systems design engineering respectively in 1994 and 2000, from the University of Waterloo, Canada. Dr. Shakshuki is an Adjunct Professor at Dalhousie University, Canada. He manages several research projects in his research expertise in the area of intelligent agent technology and its applications. He is a member of IEEE, ACM, AAAI and APENS.

Xinyu Xing is a PhD student at the Department of Computer Science, University of Colorado at Boulderat, USA. He received his MSc degree from Jodrey School of Computer Science, Acadia University, Canada in 2008. His research interests are in the areas of multi-agent systems and wireless sensor network.

Tarek R. Sheltami is currently associate professor at the Computer Engineering Department at King Fahd University of Petroleum and Minerals (KFUPM) Dhahran, Kingdom of Saudi Arabia. He joined the department on September, 2004. Before joining the KFUPM, Dr. Sheltami was a research associate professor at the School of Information Technology and Engineering (SITE), University of Ottawa, Ontario, Canada. He has two years of industrial experience at GamaEng Inc (2002–2004). Dr. Sheltami has been a member of a technical program and organizing committees of several international IEEE/ACM conferences. Dr. Sheltamifls research interests are in the area of wireless communications, wireless ad hoc and sensors networks, mobile infrastructure protocols, network control/mobility management, UMTS, and performance evaluation of wireless communication networks.