Evaluation of Routing Protocols for Internet-Enabled Wireless Sensor Networks

Ion Emilian Radoi
School of Informatics
The University of Edinburgh
Edinburgh, United Kingdom
e-mail: emilian.radoi@gmail.com

Aditi Shenoy
School of Informatics
The University of Edinburgh
Edinburgh, United Kingdom
e-mail: shenoy.aditi@gmail.com

D. K. Arvind
School of Informatics
The University of Edinburgh
Edinburgh, United Kingdom
e-mail: dka@inf.ed.ac.uk

Abstract - **This paper investigates the choice of routing algorithms for a CoAP-UDP stack for a internet-enabled Wireless Sensor Network (WSN) running an application for emergency monitoring and evacuation of people in a building. The routing protocols considered belong to two classes: proactive protocols (CTP, RPL) and reactive protocols (AODV, DSR). The emergency monitoring and evacuation scenario, running on a WSN with a full stack, was modelled and simulated in the SpeckSim behavioural simulator. The results of our study demonstrated that AODV would be the protocol of choice for the chosen application. The methodology advocated is sufficiently general for investigating protocol choices for other applications.**

Keywords – WSN; routing protocols; CoAP; AODV; RPL.

I. INTRODUCTION

Internet-enabled WSNs can be used to bridge the physical world that we inhabit with the virtual world of the Internet. Miniature battery-operated sensors with wireless connectivity and processing capability which are attached to objects can be used to extend the connectivity of the Internet. Information from the sensory data can be used to build web-oriented applications such as smart metering and smart building networks, and a number of bodies have been active in their standardisation.

The Internet Protocol for Smart Objects (IPSO) Alliance [17] has been involved in the interfacing of IP technology with everyday physical devices. In addition, the Internet Engineering Task Force (IETF) has incorporated several Working Groups towards the standardization of IP protocols for these objects. Their first attempt was to compress IPv6 over Low power Wireless Personal Area Networks (6LoWPAN) [1] to enable its use in low-power 802.15.4 radios. The Routing Over Low power and Lossy networks (ROLL) Working Group is promoting a routing protocol called the IPv6 Routing Protocol for Low-power and Lossy networks (RPL) [2].

There is now a progress from concern about network connectivity between physical objects (actuators, sensors, embedded devices) and the Internet, towards building useful web service-oriented applications over this basic layer of connectivity.

Internet-enabled WSNs can be realised by adapting traditional web protocols in ways suitable to different applications, thereby enabling the integration of these sensor-enriched physical objects to the Internet. This can be made possible if the existing REpresentational State Transfer (REST) architectural style can be extended to accommodate new application layer protocols suitable for WSNs over existing transport protocols such as TCP/ UDP.

The IETF Constrained RESTful Environments (CoRE) Working Group [18] is focusing on designing application layer protocols that manipulate sensor data, which overcome the restrictions of their networking environments. The resulting Constrained Application Protocol (CoAP) [3] integrates the different facets of the web service architecture. CoAP includes a subset of the REST features that are available in HTTP, to enable effective Machine-to-Machine (M2M) communication between devices.

The question asked in this study was that for a given choice of application/transport layer protocol (CoAP/UDP), and a data link layer protocol (SpeckMAC-D [16]), what is the appropriate choice of routing protocol for the given application scenario of emergency monitoring and evacuation of people in a building.

We considered two sets of routing protocols classified on the basis of their gathering and maintenance of routing information. Proactive protocols generate routing tables and periodically exchange update information, and reactive ones which do not, but instead trigger a discovery process when routing information is required. We selected RPL and Collection Tree Protocol (CTP) from the proactive class, and Ad hoc On-Demand Distance Vector Routing (AODV) and Dynamic Source Routing (DSR) from the reactive one.

We have implemented CoAP-UDP over each of the chosen routing protocols, and have examined in each case the behaviour of the resulting protocol stack. Based on a selection of evaluation metrics relevant to constrained networks, we determine the suitability of the routing protocols for ensuring effective, reliable communication between resource-constrained devices.

Section II reviews related work in this field; Section III describes the different protocols that were implemented; Section IV describes the emergency monitoring and evacuation application and its implementation in the simulator, along with the implementation of the routing protocols. Section V provides an analysis of the results and Section VI presents the concluding remarks.

II. RELATED WORK

The challenge in achieving WSN interoperability with IP networks has been recognised [10], and so has the need for an open resource-oriented architecture for building web services in sensor networks.

A few research papers have concentrated on the need for a new application protocol such as CoAP, and justify its use

 ISBN: 978-1-61208-203-5

in WSNs. Colitti et al. [4] provide a detailed description of CoAP and compare it with HTTP by running the Contiki and libcoap [20] CoAP versions. They demonstrated that the energy consumption of CoAP-running sensor nodes is significantly lower than those running HTTP. Kovatsch et al. [19] describe an implementation of the IETF CoAP protocol for the Contiki operating system that leverages the ContikiMAC low-power duty cycling mechanism to provide power efficiency.

Kuladinithi et al. [5] describe CoAP's Contiki and TinyOS implementations to integrate CoAP into an existing WSN-oriented logistics system for cargo containers. The focus is on performing CoAP-HTTP comparisons based on certain application-specific evaluation metrics such as data retrieval and access rates/times. In this paper we evaluate the entire protocol stack, based on five evaluation metrics.

Both simulated and real implementations of RPL have been evaluated [6][7] since the protocol was selected as the IETF candidate for standardization in WSNs.

Several papers and dissertations [15] have compared different routing protocols for WSNs, but few comparisons have been made between the protocols that we have considered in this study. In [11], for instance, on-demand routing protocols such as AODV, DSDV and DSR are evaluated using the NS2 simulator which concluded that AODV outperforms the other two protocols in terms of packet delivery ratio. In [14], the same protocols were evaluated with similar results in terms of packet delivery, but higher performance was demonstrated by DSDV as the network was scaled and a radio shadowing model was considered.

A number of simulators have been developed for understanding the behaviour of WSNs [10]. The Cooja simulator [11], for example, is focussed on simulating hardware details of the WSN nodes. TOSSIM [12] is a discrete event application-level simulator that can be used for TinyOS-based WSNs. The former is better suited for analysing the impact of low-level network details at cycle-level accuracies, whereas the latter is better suited for capturing the impact of application-specific issues on performance.

In this study, we used the SpeckSim [13] behavioural-level simulation environment which has been designed to perform evaluation across the different layers of a protocol stack to determine the most efficient set of protocols for a given class of applications.

III. BACKGROUND

This section briefly presents the protocols that were chosen for this study. It is followed by the implementation section which further discusses them in more detail.

A. The Constrained Application Protocol (CoAP)

The interfacing of resource-constrained embedded devices to the Internet requires extensions to its current architecture and new light-weight representations. HTTP is less able to handle M2M interactions efficiently with the additional overhead of heavy-weight resource representation formats such as HTML and XML. There is a need for a compact REST-affiliated architectural style to connect internet-enabled physical objects and access them through universally accepted standards-based methods.

CoAP is a generic web protocol, defined by the IETF CoRE Working Group [3], which aims to enable interoperability between embedded constrained M2M applications. The goal of this protocol is not only to compress HTTP, but to include constraints such as statelessness, cache-ability, layered system, uniform interface common in current web protocols and additional features such as multicast support, built-in device discovery, asynchronous message exchanges and bulk transfer of data.

CoAP web services have been designed for end-to-end constrained devices. A detailed description of CoAP's features is presented in [5], most of which have been implemented in the SpeckSim simulator.

B. Routing Protocols

For the purpose of this study, two classes of protocols have been considered: i) proactive and ii) reactive protocols.

1) Proactive Protocols

Proactive protocols involve the generation and maintenance of routing tables by the nodes in the network. Two protocols that fall under this class are RPL and CTP. Even though CTP nodes do not explicitly maintain routing tables but only a single route towards the root node, it can be classified as a proactive protocol.

a) RPL

RPL has been proposed by the IETF ROLL Working Group as a standard routing protocol for IPv6 routing in WSNs, since existing routing protocols do not satisfy all the requirements for low-power and lossy networks.

RPL organises the network as directed acyclic graphs, starting from the root nodes. It forms a non-transitive, non-broadcast, multiple-access, flexible topology, as described in the IETF draft [1].

b) CTP

CTP is a tree-based collection protocol. When the topology is formed, some of the nodes advertise themselves as root nodes, and the rest of the nodes form routing trees to these roots. CTP is address-free, i.e., a node implicitly chooses a root by choosing a next hop.

2) Reactive Protocols

Reactive protocols do not generate routing tables; instead they build and maintain cache tables based on routing information acquired after route discovery events. Two such protocols are DSR and AODV.

a) DSR

DSR was designed for use in multi-hop wireless networks of mobile nodes. It allows the network to be completely self-organised and self-configuring, without the need for any existing network infrastructure or administration.

The protocol is based on a route discovery and a route maintenance mechanism which operate on demand. It provides loop-free routing, does not send periodic packets of

 ISBN: 978-1-61208-203-5

any kind and supports unidirectional links and asymmetric routes.

b) AODV

AODV is a routing protocol for mobile ad-hoc networks. It uses destination sequence numbers to ensure loop freedom at all times, avoiding problems associated with classical distance vector protocols (such as "counting to infinity").

IV. IMPLEMENTATION DETAILS

Table I summarises the main features of the routing protocols.

TABLE I. ROUTING PROTOCOLS SUMMARY

Characteristics	AODV	DSR	CTP	RPL
Class	Reactive	Reactive	Proactive	Proactive
Tree based	No	No	Yes	Yes
Periodic control messages	Yes (maintains neighbour tables)	No	Yes	Yes
Types of traffic	P2P, P2MP	P2P, P2MP	MP2P, P2MP	P2P, MP2P, P2MP
Types of tables	Routing Table	Cache Table	None (just next hop)	Routing Table
Fault tolerance	Yes	Yes	Yes	No
Communication links	Bidirectional	Unidirectional Bidirectional	Unidirectional	Unidirectional Bidirectional

A. SpeckSim Simulation Framework

The SpeckSim simulation framework [13] is a behavioural level simulator designed for modelling and performance analysis of WSNs. SpeckSim enables modelling and simulation at the different levels of abstraction: devices, networks, layers of the protocol stack, and the application and deployment environment. The simulator incorporates several protocols at the data link and network layers, radio channel models and hardware models for the analysis of power consumption and resource usage.

B. Fire Evacuation from a "Smart" Building

An example of an application explored in this paper is the monitoring and emergency evacuation of a building in the event of a fire, using a internet-enabled WSN attached to the building fabric. The web service identifies the location of the occupants and dynamically computes the safest path [8] towards one of the exits (should such a path exist) and the direction towards this exit is displayed to the occupants in the form of a strobing LED. CoAP (with UDP) is used for this purpose. A reliable transport protocol is not needed due to CoAP's simple retransmission mechanism.

The implementation of the different routing protocols enables effective computation of the hazard times and the safest path from the fire towards the exits. The simulation experiments investigate the impact of the choice of routing protocol. The aim is to examine which one is better suited for this application and the trade-offs in their performance.

C. Choice of SpeckMAC-D as the Data Link Protocol

Our application features low data access rates and the Media Access Protocol (MAC) protocol should be chosen accordingly. The SpeckSim simulator provides a library of MAC protocols from which SpeckMAC-D was chosen as it had outperformed the other protocols in terms of energy consumption and battery lifetime in a previous published study [16]. Unlike other channel probing protocols, SpeckMAC-D performs better for both unicast and broadcast packets which further justifies its selection for this application.

D. CoAP Implementation in SpeckSim

The CoAP implementation in SpeckSim conforms to the description in Draft-8 [3]. CoAP nodes communicate by passing CoAP messages comprising of a fixed 8-byte header. The messages come in different types: Confirmable, Non-confirmable, Acknowledgement, and Reset. Confirmable messages guarantee delivery through the network. They are transmitted in the form of simple retransmissions by increasing the timeout by an order of 2 until the number of maximum retransmissions allowed is reached. For the purpose of the fire evacuation application, all messages are declared to be "Confirmable".

The implementation of the emergency monitoring and evacuation application involves the transmission of CoAP messages at regular intervals between the nodes that detect a person's presence and the exits of the building. The locations of the people within the building are monitored as they traverse the safe paths towards the exits. The paths are continuously updated, taking into account the fire's progress.

E. Implementation of Routing Protocols in SpeckSim

This section briefly describes the implementation of the different routing protocols under study, which are evaluated using CoAP for the fire evacuation application.

1) RPL

The RPL implementation provided in SpeckSim is based on the IETF draft [2]. RPL is optimised for collection networks (ones based on typical traffic of multipoint-to-point (MP2P) and point-to-multipoint (P2MP)), with occasional point-to-point (P2P) traffic.

RPL uses MP2P traffic for data collection and P2MP traffic for configuration purposes. The collection networks have multiple nodes that report periodically to a few collection/sink nodes. Sink nodes rarely choose P2P communication with one of the sender nodes.

2) CTP

CTP is a tree-based collection protocol. When a root node starts up, it broadcasts beacons (routing frames) to generate bidirectional links between the nodes. When a non-root node starts up, it sends routing frames with the "P" bit set (i.e., requesting routing information) until it receives a reply (containing its next hop (parent), the node id, and a metric ETX for evaluating the best parent node). After receiving the reply, it starts broadcasting beacons (similar to the root node) and it can establish connections with new adjacent nodes. Each node holds a parent list as a backup in

 ISBN: 978-1-61208-203-5

case the parent node fails. Should this happen, selection of a new parent node will occur.

The protocol checks for frame duplications and allows up to 32 retransmissions in case of lost data frames or acknowledgements.

The ETX metric was changed to "hop count" due to the ambiguity in the protocol's specification on how to deal with routing loops, thus resulting in loop-free routing.

3) AODV

The implementation of the AODV protocol in the SpeckSim simulator is based on the IETF draft [21].

A node broadcasts a Route Request (RREQ) message when it needs to find a route for a new destination. A route can be obtained when the RREQ either reaches the destination itself or an intermediate node with a 'fresh enough' route to the destination (a 'fresh enough' route is a valid route entry for the destination whose associated sequence number is at least as great as that contained in the RREQ). Each node receiving the request caches a route back to the originator of the request, so that the reply can be unicast from the destination to that originator, or likewise from any intermediate node able to satisfy the request.

Route Error messages are used to propagate link/node failures and changes through the network. The messages may be either broadcasts or unicasts.

4) DSR

The DSR implementation provided in the SpeckSim simulator is based on the paper authored by David B. Johnson et al. [9].

Before the transmission of data packets containing CoAP messages, a node first searches for a route in its cache table. If it finds a route towards the desired destination, it builds the data packet by adding the necessary header information which includes a list/path of nodes that the packet will have to follow in order to reach the destination.

The other nodes in the network will forward the data packet based on the routing information transported in the header of the packet. When a node receives a data packet, it will send an acknowledgement (ACK) message to the node that previously sent the message (one-hop ACKs).

If the node does not find a route towards the desired destination in its cache table, it will initiate the Route Discovery process. The current DSR implementation uses two types of Route Requests: a simple Route Request and a piggyback Route Request (contains the routing information of a Route Reply). This allows it to support unidirectional links and avoid infinite recursion of Route Discoveries.

If the packet is retransmitted for the maximum number of times (15), this will generate Route Error messages which are used to identify the link over which the packet could not be forwarded. The cache table stores only one route for a destination and is populated by the receipt of piggyback Route Requests and Route Replies.

V. RESULTS AND ANALYSIS

We have simulated two building topologies in the SpeckSim simulator: a grid (Manhattan) topology and a less regular topology of a floor in a real building, the Informatics Forum (Figure 1). In both cases each node is within radio range of its immediate neighbours. In Figure 1, the 24-node WSN populates the corridor. Note that the most favourable placement for RPL and CTP root nodes is at the exit nodes of the building, such that the MP2P capability of these protocols can be exploited.

Figure 1. Informatics Forum floor plan and its representation in SpeckSim

A. Test Cases

The fire evacuation application involves simulating people within the building and their passage to safe exits. It also simulates the continuous transmission of CoAP messages between the nodes that detect people and the exit nodes. These messages represent updates on people's location in the building as they move along the paths towards the exits. We have implemented the fire evacuation application on two networks: a 16-node grid-based topology and a 24-node building topology.

The fire evacuation scenario was simulated for the entire protocol stack for the following metrics: delivery ratio, latency, overhead, power consumption, and fault tolerance. Also, scalability studies were performed on grid networks for the following metrics: overhead, packet loss and latency.

B. Results

Each node simulated in SpeckSim has the following characteristics:

- Battery: capacity - 1mAh, voltage- 3V.
- MCU: active current - 0.005mA, sleep current - 0.001mA, off current - 0mA.
- Radio: Perfect Radio Shell, range - 0.35 units.
- Power up delay: Min=0s, Max=1s.

We now present the results that have been gathered by running the fire monitoring and evacuation application in SpeckSim, for the different routing protocols under CoAP-UDP. The results presented are the average of six runs.

AODV exhibits the highest delivery ratio of 100 percent, guaranteeing the transmission of all the messages in the network to the intended destinations.

The latency (Figure 2) is important for the chosen scenario because emergency evacuation requires rapid responsiveness in the network for the short period of time of the evacuation. RPL outperforms the other protocols, as the exit nodes of the building were configured as root nodes, thus leading to effective route selections. The higher latency in AODV and DSR (an average of 11 seconds) can be attributed to their time-consuming route discovery process. However, the 16 second average discrepancy between the two is due to AODV's use of only bidirectional links.

 ISBN: 978-1-61208-203-5

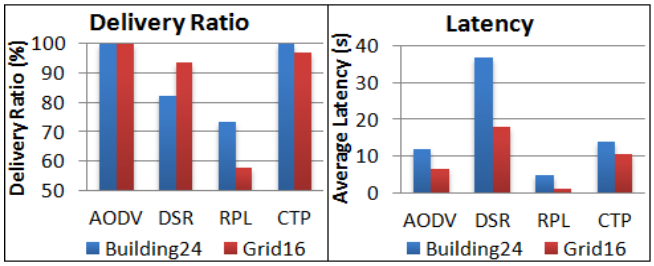

Figure 2. Delivery Ratio and Latency measurements for the two scenarios

The protocol overhead is a useful metric for analysis because it has a direct effect on the average power consumption of the network. It was measured by counting the number of control packets exchanged in the network over a period of time (approximately 650s). We observed a lower overhead for the proactive protocols (Figure 3), owing to the usage of algorithms such as the Trickle timer, which reduce the control packet exchange when the topology is stable.

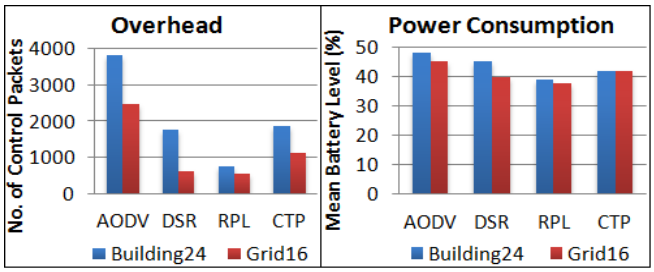

Figure 3. Overhead and Power Consumption results for the two scenarios

AODV needs to maintain neighbour tables in order to use only bidirectional links, but this increases its overhead. Due to the low number of data packets generated (approximately 250 packets) and the short simulation time (approximately 650s), DSR did not have the chance to outclass the other protocols in terms of overhead, even though it does not exchange periodic messages. A more significant difference is shown in Figure 4 of the scalability scenarios, where DSR clearly has lower overhead compared to the other protocols.

Figure 3 also shows the power consumption in terms of the percentage of depleted battery life at the end of the simulation run. This metric is important, as the batteries must last until the evacuation of the building is complete. It was observed that, for all the routing protocols, less than 50% of the batteries' power levels (1 mAh capacity) were drained after running the scenario. Note that any type of battery likely to be used in a real-life deployment is expected to be in order of hundreds of mAh (i.e., CR2032 provides 220 mAh or AA batteries which provide 2500 mAh).

The mean battery consumption was measured when each protocol was run in the simulator for the same type and number of specks. All the protocols displayed similar battery lifetimes/consumption because of the limited run time of the scenario. It may be possible to observe more prominent differences in power consumption if they were simulated on larger topologies for a longer duration.

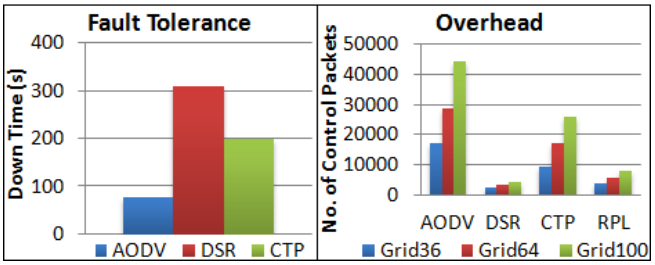

Figure 4. Fault Tolerance (building scenario), Ovehead (grid topologies)

Fault tolerance is another important metric in the context of this application due to the increased chance of the nodes getting damaged. Node failures must be detected and propagated throughout the network so that alternative paths can be found in a timely manner. No fault tolerance mechanisms have been defined in recent RPL drafts because of the overhead that may be created in terms of bandwidth and energy consumption. Therefore, RPL's response to node failures cannot be evaluated.

Figure 4 shows that AODV has the highest tolerance for node failures. This plays an important role in its selection for the CoAP-UDP stack for this particular application.

It can be seen that DSR's down time is higher in comparison to AODV. One plausible explanation is that the DSR implementation in SpeckSim is built to work over both unidirectional and bidirectional links. This implies that the Route Discovery process for this reactive protocol may cause the Route Reply to reach the sender through a different path from that of the Route Request, which causes an additional delay. AODV makes use only of bidirectional links (Route Replies use the backward route of the Route Request), thereby having a reduced down-time. Also, DSR retransmits a packet 15 times before considering a route to be broken, as opposed to AODV which performs only 5 retransmissions.

The protocol drafts do not specify most of the delays and timers used by the protocols, thus making these values implementation specific. Since the intervals for the periodic control packets are implementation specific, in the case of AODV, DSR and CTP when choosing these values, the focus was on reducing the overhead rather than minimising the reconvergence time of the network. This explains the significant down-time of the network when a node fails.

C. *Scalability Results*

The scalability tests have the purpose of validating the results obtained in the case of the fire scenario for the grid and building topologies.

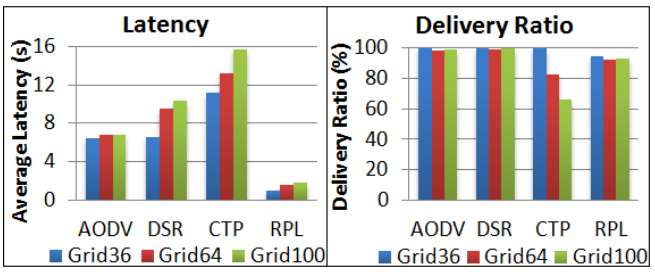

Figure 5. Latency and Delivery Ratio (grid topologies)

 ISBN: 978-1-61208-203-5

The graphs presented in Figures 4 and 5 show that the pattern of the results obtained for the evacuation scenario is maintained as the grid network is scaled to larger topologies.

The only pattern that is not maintained is in the case of the overhead metric for DSR. As previously argued, the protocol is designed not to exchange periodic messages, so it is expected to have a lower overhead over a longer run-time period. For the emergency evacuation scenario DSR did not have the opportunity to outclass the other protocols in terms of overhead. However, a more significant difference is noticed in the scalability studies (Figure 4) where it has a clearly lower overhead than the other protocols.

TABLE II. RESULTS SUMMARY

Characteristics	AODV	DSR	CTP	RPL
Delivery Ratio (%)	100	87.94	96.98	65.62
Average Latency (s)	9.30	27.46	12.21	3.08
Overhead (no. of control packets)	3163	1199	1508	660
Power Consumption (% battery left)	53.32	57.26	58.09	61.50
Fault Tolerance (s) -down time-	78	310	199	-

VI. CONCLUSIONS

This paper has demonstrated an approach for analysing the choice of routing algorithms for a CoAP-UDP protocol stack for internet-enabled WSNs. Table II summarises the performance results for the four routing protocols for the fire evacuation scenario. RPL outperforms the other routing protocols for three out of five metrics. However, it is not fault tolerant and has the lowest delivery ratio.

We can observe that no one protocol outperforms the others for all the metrics which were selected to be relevant to the application. Therefore, the selection of the appropriate protocol to be used with the CoAP-UDP network stack would depend on the weightage accorded to each metric.

In the given scenario, the overhead metric was included to gauge its impact on power consumption. We concluded that power is less of an issue for the time duration simulated. Therefore the overhead metric should not be the prime reason for selecting a routing protocol for this scenario.

Of the five metrics chosen for evaluation one can prioritise three of them: delivery ratio, latency and fault tolerance. One can observe in Table II that AODV and RPL are the two most competitive protocols. Whereas AODV responds well to failures and exhibits a high delivery ratio, RPL has a significantly lower latency.

In case of the fire emergency scenario, the probability of the sensors getting damaged is high. Thus, the network must be able to react to topology changes caused by node failures. Since the current RPL implementation is not fault tolerant, this leaves us to conclude that the most suitable routing protocol (from the ones evaluated) for use in a emergency evacuation scenario is the AODV routing protocol.

ACKNOWLEDGEMENT

The authors wish to thank partial support from ICT FP7 projects HOBNET (257466) and PLANET (257649).

REFERENCES

[1] N. Kushalnagar, G. Montenegro, and C. Schumacher, “IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, Goals”, rfc 4919, 2007.

[2] T. Winter, P. Thubert, and the ROLL Team, “RPL: IPv6 Routing Protocol for Low power and Lossy Networks draft-ietf-roll-rpl-10”, June 28, 2010.

[3] Z. Shelby, K. Hartke, and B. Frank, “Constrained Application Protocol (CoAP) draft-ietf-core-coap-08”, Nov. 1, 2011.

[4] W. Colitti, K. Steenhaut, N. De Caro, B. Buta, and V. Dobrota, “Evaluation of Constrained Application Protocol for Wireless Sensor Networks”, In Proc. of Local & Metropolitan Area Networks (LANMAN), 2011 18th IEEE Workshop on, Oct. 2011.

[5] K. Kuladinithi, O. Bergmann, T. Pötsch, M. Becker, and C. Görg, “Implementation of CoAP and its Application in Transport Logistics”, in Proc. IP+SN, Chicago, IL, USA, 2011.

[6] N. Tsiftes, J. Eriksson, and A. Dunkels, “Low-power wireless IPv6 routing with ContikiRPL”, in Proc. 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, April 12-16, 2010, Stockholm, Sweden.

[7] J. Tripathi, J. C. de Oliveira, and J. P.Vasseur, “A performance evaluation study of RPL: Routing Protocol for Low power and Lossy Networks”, Information Sciences and Systems (CISS), 2010 44th Annual Conference on, March 2010, pp. 1–6.

[8] M. Barnes, H. Leather, and D.K. Arvind “Emergency Evacuation using Wireless Sensor Networks”, in Proc. SenseApp 2007: 2nd IEEE Int. Workshop on Practical Issues in Building Sensor Network Applications, 15 - 18 Oct. 2007, Dublin, Ireland, IEEE.

[9] D.B. Johnson, D.A. Maltz, and J. Broch. “DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks”, Ad Hoc Networking, chapter 5: pp. 139 - 172. Addison-Wesley, 2001.

[10] E. Egea-Lopez, J. Vales-Alonso, A. Martinez-Sala, P. Pavon-Marino, and J. Garcia-Haro, “Simulation tools for wireless sensor networks", in Proc. Int. Symp. on Performance Evaluation of Computer and Telecommunication Systems, 2005, pp. 559-566.

[11] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, “Cross-level sensor network simulation with COOJA," in Proc. 31st IEEE Conf. on Local Computer Networks, 2006, pp. 641-648.

[12] P. Levis, N. Lee, M. Welsh, and D. Culler, “TOSSIM: Accurate and scalable simulation of entire TinyOS applications,", 1st Int. Conf. On Embedded Networked Sensor Systems. ACM, 2003, pp. 126 – 137.

[13] Specksim, http://www.specknet.org/dev/specksim, [April 2012].

[14] T. Yang, M. Ikeda, G. DeMarco, and L. Barolli, “Performance Behavior of AODV, DSR and DSDV Protocols for Different Radio Models in Ad-Hoc Sensor Networks”, in Proc. Int. Conf on Parallel Processing Workshops, Sept. 2007.

[15] I.E. Radoi, “Evaluation of Routing Protocols in WSN”, MSc Dissertation, School of Informatics, University of Edinburgh, 2011.

[16] K.J. Wong and D.K. Arvind, “SpeckMAC: low-power decentralised MAC protocols for low data rate transmissions in specknets.” Proceedings of the 2nd international workshop on Multihop ad hoc networks from theory to reality. ACM, 2006. 71-78.

[17] IPSO Alliance, http://www.ipso-alliance.org/, [April 2012].

[18] https://datatracker.ietf.org/wg/core/charter/, [April 2012].

[19] M. Kovatsch, S. Duquennoy, and A. Dunkels. “A Low-Power CoAP for Contiki.”, in Proc of the Workshop on Internet of Things Technology and Architectures (IEEE IoTech 2011), Spain, Oct 2011.

[20] http://sourceforge.net/projects/libcoap/, [April 2012].

[21] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing”, rfc 3561, July 2003.

 ISBN: 978-1-61208-203-5