

A.MO.R : An Adaptive Routing Mechanism for WSN Health Applications

S. Giannoulis, A. Prayati, C. Antonopoulos, G. Papadopoulos

Research and Innovation Center in Information, Communication, and Knowledge Technologies -Industrial Systems Institute, Greece

sgiannoulis@ee.upatras.gr, {prayati, cantonop, papadopoulos}@isi.gr

Abstract¹ - As Wireless Sensor Network (WSN) applications become more popular in every day life, the importance of flexible communication protocols rises higher. The limited resources and time-critical nature of WSN must be addressed by the network layer, as one of the most mandatory communication components to support routing capabilities in multi-hop mobile sensor nodes' topologies. WSNs application particularities are analyzed in order to form the design framework for an adaptive routing algorithm that can address all WSN needs. Simulation tests have shown that A.MO.R (Adaptive Multi-Objective Routing) can achieve the WSNs objectives with low communication and energy overhead while meeting QoS and congestion requirements.

Keywords-adaptive routing, multiple routing metrics, health WSN applications.

I. INTRODUCTION

Focusing on the network layer of WSNs, even though some routing algorithms have been proposed for WSNs, many routing aspects are either still not addressed at all or are handled at the expense of high computational complexity. Various metrics have been proposed for routes, like hop-count, end-to-end delay, or remaining power. Many of the approaches propose dual metric protocols that can use either one depending on the application demands.

Mesh topologies are harder to handle and most of the times routes break down due to mobility. The use of a fixed infrastructure, where applicable, may improve the route life-time and network performance. The node communication range, the deployment scheme and the connectivity degree of the fixed infrastructure are some of the most important aspects of this approach.

On the other hand, the dynamic nature of WSNs, requires a protocol behavior adaptive to changing network performance. Various factors must be taken into account in order to satisfy the adaptivity requirement whilst meeting network QoS and power-related requirements as well. Cross-layer design

approaches between MAC and routing have managed enhancing network performance, even though no standard interface between MAC and Routing has been established yet.

Another requirement in WSN routing when handling broken links is the distinction between nodes moving out of range and nodes still in range with packet collisions. This distinction is crucial since the first event should trigger link or node deletion from the routing table, while the latter should be handled just by packet retransmission.

In order to treat the dynamic nature of WSNs under the constraint of limited memory and processing resources, hop-by-hop routing is preferred to source routing. The need for effective flooding with minimum overhead can be achieved by loop avoidance algorithms for reducing unnecessary retransmissions of broadcast packets throughout the network. However, no optimal solution exists to minimize bandwidth overhead, end-to-end delay and control packet overhead. Clustering techniques create groups of nodes that can cooperate to form a cluster, imposing even more network overhead in terms of bandwidth, delay and power consumption.

This paper illustrates the needs of state-of-the-art WSN applications with respect to routing technical requirements and proposes a routing algorithm for a given system deployment that meets these constraints. Simulation results show that the algorithm is efficient, with low overhead, performing well in large-scale deployments.

II. WSN APPLICATION SCENARIOS REQUIRING ROUTING

As discussed in [2], there are performance requirements that are common to all WSN application scenarios, with outdoors tracking scenarios being among the most demanding ones in terms of guaranteed real-time network performance under high application traffic rate. Such scenarios refer mostly to health applications from vital-sign monitoring of patient or athletes to people rescue in emergencies.

In most cases, the network spans over a mesh-based topology with nodes deployed randomly or in a grid-

¹ The work reported here was performed as part of the ongoing research Program uSWN FP6-2005 IST-034642 and funded by the European Social Fund (ESF).

like scheme. There are also cases of health applications in hospitals or sanatoria, where some nodes may be placed at fixed points forming a backbone network and thus assisting the WSN communication of mobiles to data sinks, reducing end-to-end delay, which is mandatory in vital signs monitoring applications.

Minimal hop count aims directly at lower end-to-end-delay, while QoS metrics like congestion indication, aim at bypassing areas with high traffic congestion. Hop-by-hop techniques to route packets are more efficient to avoid any source routing information overhead inside the header in every packet, as well as the need to have global knowledge of the network topology. All of the above aim to enhance the scalability of the designed routing algorithm.

Multiple data sinks minimize the average hop count from any node to a sink and may be used in redundancy. Minimizing average hop count by using multiple data sinks ensures higher throughput and lower end-to-end delays.

The tracking algorithm accuracy based on the tracking algorithm used, the optimal deployment must adjust the nodes distances to also ensure network connectivity.

The mobile object maximum speed is taken into consideration when calculating the total throughput needed by the system, adjusting the retransmission period of routing protocol to meet the maximum offered bandwidth by the physical and MAC layer.

The Retransmission Period can be then achieved by $(\text{system resolution} - \text{tracking resolution}) / \text{speed}$. *Maximum needed throughput* calculated by the $\text{AvgHopcount} * (8 * \text{Packet size}) / \text{Retransmission Period}$ is used for MAC performance analysis to set the routing constraints.

The maximum number of moving objects is considered when calculating the total throughput needed by the system, adjusting the retransmission period of routing protocol to meet the maximum offered bandwidth by the physical and MAC layer.

The moving object's tracked position overall error affects the total throughput needed by the system, adjusting the retransmission period of routing protocol to meet the maximum offered bandwidth by the physical and MAC layer as stated above.

The measured data reporting period by the sensor nodes of critical monitoring applications affects the required data transmission rate imposing a constraint to the routing overall latency.

III. A.M.O.R. SPECIFICATION

The routing protocol is designed based on the assumption that a fixed infrastructure of wireless sensor nodes covers 100% of the area at all times, able to act as routers of messages towards the nearest sink. The overlapping of the area covered by each fixed node gives the opportunity of multiple alternatives for routers to send packets to the nearest sink. Data sinks are strategically deployed in order to cover the whole

area with high connectivity degree. Boundaries of these areas are defined by nodes that are of equal hop distance from more than one data sinks. The network topology is shown in Figure 1.

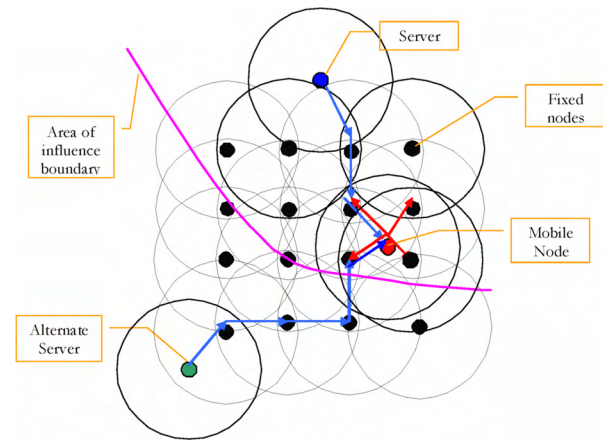


Figure 1. Network topology

Based on the above, the proposed routing protocol maximizes the exploitation of the fixed infrastructure, while minimizing the impact of the mobility patterns of the mobile nodes.

Step 1 – Creation of fixed infrastructure routing tree: Flooding advertisement of server position and fixed routing tree construction based partly on techniques used in [1].

Increase of a hop count per hop: this helps every fixed node to track its maximum distance from the server that sent the discovery packet. Each node retransmitting the advertisement packet from a server increases this field by one.

Detection and prevention of retransmissions/loops: With the use of a continuously increased age identifier, old messages are pinpointed and discarded and thus loops are avoided. In case a node receives a packet from sink A and has a smaller age identifier than the one stored in the node as latest, the packet is considered as old and discarded. This minimizes the overhead of the flooding process in a system-wide manner. Each node that receives an advertisement packet saves the tuple server address, server age, hop count in order to be able to identify future duplicate packets that have been re-broadcasted by the same node.

Only fixed nodes rebroadcast routing messages: Only fixed nodes find routes to the sinks and act as routers, minimizing the disconnection problems due to user mobility. Every mobile node will only have to select a close fixed node as repeater and use it to send messages to the closest data sink.

Step 2 – Local advertisement of node status and creation of neighbourhood information table: A disconnection management component in each fixed node advertises to its 1-hop away neighbours its status

based on the span 1-hop advertisement technique. The attributes advertised are listed below:

Remaining power: helps all nodes to know the power status of their neighbour nodes and hence ensure that routing can take power-aware decisions.

Hop count to sink: ensures that routing can take decisions based on minimum distance, since all nodes try to rebroadcast packets to nodes nearest to the sink.

Congestion status: allows routing to decide based on QoS and to possibly redirect traffic to avoid local congestion. The QoS metric is based on queue load and application data traffic generation rate.

Link Stability: based on the RSSI of each link, this metric is used by routing for ensuring route stability.

Step 3 – Selection of parent fixed node based on selected active metrics: Each mobile node, based on the advertisement of fixed nodes, selects a fixed node as parent and forwards messages through it. Mobile nodes must maintain tables with 1-hop away nodes, including the information of remaining power, hop-count, congestion status and link stability provided by the disconnection management component. Fixed nodes must only maintain a table with possible closer fixed neighbour nodes to all available data sinks.

Step 4 – Communication of packets to the closest data sink: Different types of communication are supported.

Sink flooding to the whole WSN: if the packet to be sent is generated by a sink, then routing will flood it to the network by following the procedure described in Step 1.

Mobile / Fixed node to the closest data sink: Based on the selected metric, each fixed node chooses the next hop and forwards the packet towards the sink. Each fixed node repeats this step until the packet arrives to the data sink. Since the metric will be based on the three aforementioned metrics, adaptability of the final metric is expected to be high on all three axis power, congestion and QoS.

Between nodes in the 1hop neighbourhood: node-to-node communication is supported by setting the destination node address when sending the packet from the application/MW layer. In this case, the packet is sent directly and only to the specific neighbor destination node.

Node broadcast at 1hop: any node (sink/fixed/mobile): may broadcast a packet to its 1-HOP neighbours by setting the destination node address to a default broadcast value when sending the packet from the application/MW layer. In this case, the packet is broadcasted to all nodes in the communication range of the sender.

IV. DISCUSSION OF ROUTING PERFORMANCE

The proposed routing algorithm has been evaluated with respect to the deployment scheme shown in Figure 2, forming a full triangular grid of both fixed and

randomly distributed mobile nodes. The tools used to simulate the protocol were Tossim and TinyViz[3].

The simulated scenario consists of 51 fixed nodes aligned to a hexagonal grid, 50 mobile nodes and three servers playing the role of data sinks, consisted of the main application server and 2 gateways that are strategically deployed in the simulated area so as to minimize the average hop count of every fixed node to the closest data sink.

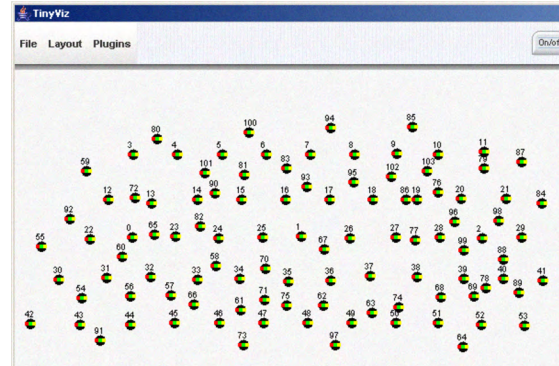


Figure 2. WSN deployment scheme for routing performance evaluation

Application packets are only created by mobile nodes in a predefined period of time per node based on the analysis of a common tracking application. More specifically, assuming, a packet payload of 50 Bytes, a tracking error of 8m, overall error allowed of 10m, mobile maximum speed of 10 Km/h, the bitrate is calculated by $(\text{speed} / \text{routing error}) * \text{payload} * 8$ to be 555bps. Thus, for the full deployment of 50 mobile nodes generating application packets, the traffic is $50 * 555$ bps, around 28 Kbps. Two simulated scenarios are considered:

- 15Kbps overall application packet creation rate representing light network traffic
- 30Kbps overall application packet creation rate as heavy traffic conditions.

TOSSIM uses TinyOs 2.x, and the underlying PHY is CC2420.

For the 15Kbps scenario, the proposed algorithm manages to keep the average hop count at 2 hops in a 100 nodes network, as shown in Figure 3, with a maximum of 4 hops. This is mainly due to the strategic placement of the data sinks so that the average and maximum hop count will remain low. The average end-to-end delay remains at 8 msecs, as presented in Figure 4. This is achieved by using advanced flooding control techniques and introducing multiple data sink support. 96% of packets were delivered successfully with no packets discarded due to queue overflow, or packet lost due to not finding a route to the closest data sink. All lost packets are due to collisions in the physical layer. The routing packets overhead is 13,9% of the application packets routed to the data sinks.

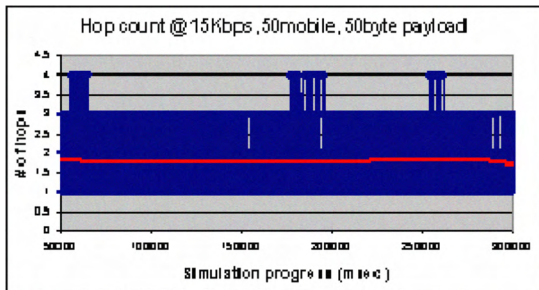


Figure 3. Hop count of packets delivered to data sink at 15Kbps

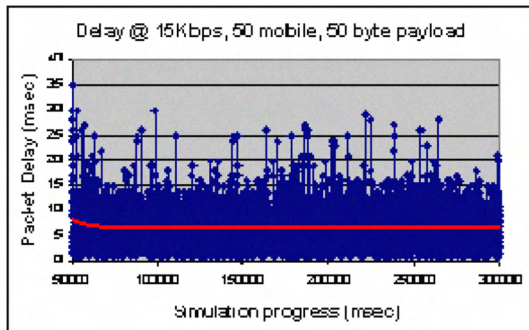


Figure 4. End-to-End delay of successfully received packets at 15 Kbps

In the second scenario, the application traffic load is doubled at 30Kbps. Figure 5 shows that the doubled application load does not affect considerably the routing efficiency with an average of 2 hops and a maximum of 4 for more packets than in the first scenario.

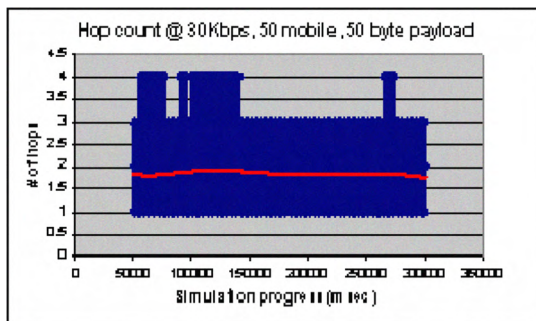


Figure 5. Hop count of packets delivered to data sink at 30Kbps

Although, the average end-to-end delay is only slightly increased at around 11msec, the number of packets received with delays over 15 and 20msec is significantly increased. In the case of the doubled workload scenario, a high number of packets experience delays over 20msec and even reaching 40msec. As far as queue and routing protocol packet

loss is concerned, still 0 packet loss is experienced. It must be noted that que ues have a length of 4 packets per layer, whilst every node supports up to a total of 36 packets buffer for processing

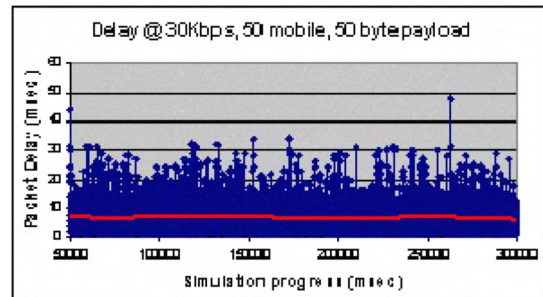


Figure 6. End-to-End delay of successfully received packets at 30 Kbps

The implementation of the proposed routing protocol in TelosB motes [4] has a footprint of 1664 bytes and needs a total of 76 bytes of RAM to run. The network protocol layer of the system needs 5746 bytes of ROM, making it suitable for use in WSN motes since it has very low ROM and RAM requirements.

V. CONCLUSIONS

A new routing protocol is proposed for WSN health applications. Power- and congestion-aware metrics are used to create a real time adaptable protocol to network status. Evaluation results from Tossim and Tinyviz are promising that this protocol can solve major problems in QoS and connectivity in large WSNs, while still remaining simple, with low footprint and memory demands.

REFERENCES

- [1] Ana-Belén, García-Hernando, José-Fernán Martínez-Ortega, Juan-Manuel López-Navarro, Aggeliki Prayati, Luis Redondo-López (Eds). Problem Solving for Wireless Sensor Networks. Springer Verlag London Limited., November 2008, ISBN: 978-1-84800-202-9
- [2] Y. Zhang and Q. Huang, "Adaptive Tree: A Learning-based Meta-Routing Strategy for Sensor Networks", IEEE 3rd Consumer Communications and Networking Conference 2006, Las Vegas, USA, January 2006
- [3] Philip Levis, Nelson Lee, Matt Welsh, and David Culler. "TOSSIM: Accurate and Scalable Simulation of Entire TinyOS Applications" Conference On Embedded Networked Sensor Systems, 2003
- [4] "Telos - Ultra low power IEEE 802.15.4 compliant wireless sensor module", Datasheet, MotelV Corporation