

# Opportunistic RPL for Reliable AMI Mesh Networks

Sedat Gormus · Filippo Tosato · Zhong Fan · Zubeir Bocus · Parag Kulkarni

Received: date / Accepted: date

**Abstract** Driven by the need to improve energy efficiency and reduce environmental impact, we observe a thrust towards enabling a Smart Grid. It is envisaged that to achieve these goals, the Smart Grid will be equipped with communications infrastructure and mechanisms that will enable near real-time control of the grid components. One of the key elements of the Smart Grid is the Advanced Metering Infrastructure (AMI) which is expected to facilitate the transport of meter readings from a smart electricity meter at the customer premises to the utility provider, and control data in the other direction. These communications can be potentially realized by deploying a self-organizing mesh network composed of smart metering nodes connected to concentrator nodes which in turn are connected to the utility provider data management systems.

This paper explores a cooperative communications approach to improve reliability of such mesh networks. The proposed opportunistic forwarding protocol called ‘ORPL’ has been realized as an enhancement on top of the Routing Protocol for Low power and lossy networks (RPL), a connectivity enabling mechanism in AMI mesh networks. In ORPL, smart meter nodes select multiple candidate relays to facilitate reliable transport of smart metering data to the concentrator node. Moreover, it is designed to work in a distributed manner thereby ensuring scalability. We also present a further extension to ORPL, i.e., ORPLx with adaptive medium access control (MAC) retransmit limit, which reduces unnecessary retransmissions. Our protocols have been evaluated and verified with comprehensive experimental results, demonstrating their effectiveness and favorable characteristics.

**Keywords** AMI · smart metering · mesh networks · opportunistic routing · cooperative communications · RPL.

## 1 Introduction and Related Work

Motivated by the need to improve energy efficiency and reduce the environmental impact, we observe a thrust towards enabling a ‘Smart Grid’. One of the key elements of the Smart Grid is the Automated Metering Infrastructure (AMI), envisaged to be a network of meter and concentrator nodes wherein meter readings are automatically sent by smart meters located at the customer premises to the Meter Data Management System (MDMS) through concentrators provisioned by the utility provider [9].

The range of communications technologies that are being considered to realize this is diverse, with a wide mix of public and private, wired and wireless, standard and proprietary networking solutions. In a neighborhood area network (NAN), WiFi, GPRS and WiMAX are some of the example technologies being considered for this segment. With the latest spectrum regulatory developments and recent trials, the TV white space (460-790 MHz) is also being put forward as a candidate. On the other hand, in home area networks (HAN), IEEE 802.15.4 (Zigbee) is the main wireless technology that is being considered, followed by WiFi and power line communications (PLC). At the moment there is no clear consensus yet on what is the best communication technology. It can be envisaged that in the foreseeable future we may

have a combination of different technologies in place to support smart metering. In this paper we focus on a wireless mesh networking solution because it is easy to deploy, reliable, and cost effective.

The topology in such a network would typically take the form of a number of trees, each rooted at the concentrator, with the meter nodes in a locality sending metering data over radio links (through multiple hops if necessary) to a concentrator in their vicinity. The concentrator in turn sends the data over a Wide Area Network (WAN) connection to the MDMS. In an urban deployment, there could be hundreds of thousands of flats/houses with one or more meters (e.g. electricity, water, gas) in each of these. In such a case, the delivery of the metering data may be affected by interference from the neighboring nodes. Although the nodes in a smart metering network are constrained in terms of processing power, it is possible to improve the reliability of the network through efficient protocols that exploit the inherent characteristics of wireless transmission<sup>1</sup>. For example, with lossy wireless links, the probability of receiving a packet at any node in a set of neighbors (anycasting) increases with the number of nodes in this set. In previous work [6], anycasting has been used to opportunistically forward a data frame from source to destination. When introducing opportunistic forwarding in a lossy wireless network scenario, the key issue is how to select the best relay set that reduces the cost of transmission. At the same time, it is also equally important to coordinate the relay set in an optimal manner so as to reduce/eliminate repeated transmissions (to reduce overhead) from the relaying nodes.

RPL, a routing mechanism designed for low power and lossy networks (LLNs) and recently standardized by the IETF ROLL working group, is a good candidate for enabling connectivity in AMI mesh networks [20]. In this paper, we present the details of a reliable cooperative forwarding mechanism for the RPL routing protocol named ORPL, study its overhead analytically, demonstrate its performance on low complexity parent selection and coordination in order to minimize the overall forwarding cost with minimal protocol overhead. In addition, we investigate an adaptive MAC retransmission mechanism to reduce overhead further. Both analytical and experimental results are presented to demonstrate the effectiveness of the proposed protocols.

In previous work (such as [7] and RPL [20]), much effort was put into developing multi-hop routing protocols for wireless networks where each node selects a single next hop neighbor as the forwarding node. However, selecting multiple next hop neighbors has been proven to be advantageous for link layer cooperative relaying mechanisms. Recent opportunistic routing work takes the link level opportunistic forwarding protocols to a new level where, at each hop, the data is forwarded to a set of nodes. Naturally, the chance of a node in this set receiving a transmission is much higher than the case of sending the data to a single next hop node. Hence, we have developed this idea to enable opportunistic forwarding in RPL based networks where the links are inherently lossy and sensor nodes are constrained in terms of transmission power.

In RPL, each node is assigned with at least a default parent and one or more candidate parents, if there exist alternative routes to the sink node (concentrator in this case). This is quite useful in a cooperative opportunistic forwarding scenario where the forwarding node can be opportunistically selected from these parent nodes. In our earlier work [11], we presented preliminary results of applying opportunistic communications (cooperative relaying [14], [4]) extensions to RPL in AMI mesh networks with the aim of enhancing the reliability of communications. In this paper we significantly extend the work in [11] by including more detailed analysis and performance results. Further, we will also describe a protocol enhancement - adaptive MAC retransmit limit, which reduces unnecessary retransmissions. The analytical and experimental results presented here will provide valuable guidelines for AMI mesh network deployment.

Recently there has been work on applying various wireless mesh networking solutions to AMI. For example, the authors of [12] propose a new distributed distance-vector routing protocol for AMI applications, designed to adapt quickly to changing link conditions while minimizing network control overhead. A wireless mesh network system is also studied in [16], where coverage gaps are filled with relay nodes. Both [12] and [16] are based on proprietary systems, while our solution is standard-compliant and can be easily maintained and extended. The authors of [13] have compared geographical routing and RPL for smart meter networks and found they show similar performance for the given network size and traffic scenario. In [17], in parallel to our earlier work [11], a multi-path RPL routing algorithm has been proposed for IEEE 802.15.4 networks, which achieves slightly improved end-to-end packet reliability and delay compared to basic RPL. While their work mainly focuses on the adaptation of 802.15.4 MAC to

<sup>1</sup> Wireless medium is a broadcast medium, therefore it is possible to design protocols that make use of this to improve link reliability.

RPL routing, our work makes use of anycasting and adaptive MAC retransmit limits in RPL. Recently, Dequunoy et al. in [8], published their findings on opportunistic forwarding for duty cycled RPL networks where the client node forwards its traffic to a selected set of parents opportunistically and the first parent that wakes up forwards the incoming traffic to the RPL root. This approach is in principle, quite similar to our earlier work, but the protocol is designed for battery powered devices and aims to improve the delay performance of such a duty cycled RPL network. In our case, the protocol was designed specifically for AMI mesh networks where the smart meters will be mains powered and the sensor radios are expected to be on all the time to enable dense deployment and reliable communication. The constraints in the application scenario considered in our work are therefore relatively relaxed. This provides a degree of freedom to exploit the opportunistic behaviour more aggressively as the main objective here is to improve reliability of AMI networks.

The rest of the paper is organized as follows. In Section 2, the opportunistic RPL protocol (ORPL) is introduced, providing protocol details, mathematical analysis, implementation and performance results. A protocol extension to ORPL (ORPLx) is discussed in Section 3, with mathematical formulation and performance results. Section 4 highlights the conclusions drawn from this study.

## 2 The Opportunistic Communication Approach - ORPL

### 2.1 Brief Overview of RPL

The IETF has developed the RPL protocol [20] to enable connectivity in low power and lossy networks. The RPL specification facilitates creating, at each node in the network, a Directed Acyclic Graph (DAG) rooted at the concentrator to maintain network state information. The edges in this graph form a path from a smart meter node to the concentrator with each node on the path being associated with a rank value such that the rank monotonically decreases at each node en route to the concentrator. The DAG construction begins with the root node advertising a control message called ‘DAG Information Object (DIO)’. Any node that receives this message and wishing to join the DAG, adds the DIO sender to its parent list, computes its rank relative to the parent node and broadcasts the DIO with the updated rank information. A node may receive a DIO message either as a broadcast or in response to a ‘DAG Information Solicitation (DIS)’ message that it sent on realizing that it does not have connectivity (either on startup or upon losing connectivity). Once the DAG is in place, any smart meter node wishing to send metering data will forward it to its parent who will then relay it on to its parent and so on until the data reaches the concentrator. In this paper, we extend RPL to provide support for opportunistic communications with the aim of improving the reliability of communications.

### 2.2 ORPL Protocol

As mentioned earlier, nodes in an AMI mesh network are typically resource constrained and the links connecting these nodes are lossy. Moreover, an increase in the number of nodes in the network exerts additional pressure on the already limited link capacity, e.g. the theoretical maximum data rate for an 802.15.4 link in a point to point configuration with no external interference is 250 Kbps. In the case of nodes using Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) as the channel access protocol at the MAC layer, the achievable throughput may be significantly reduced with an increase in the number of nodes competing to access the medium. This can have an impact on the reliability of communications. The ensuing discussion presents Opportunistic RPL (ORPL), a protocol that shows how the inherent broadcast nature of wireless communications can be exploited to improve reliability.

Fig. 1 shows the packet forwarding process in RPL and ORPL at a high level. As shown in this figure, a node employing RPL would send/forward a frame along the default route, i.e., to its default parent node. Any other node that overhears this transmission, will simply ignore it. If the default parent fails to receive this frame, the client node retransmits the frame. On the other hand, a node employing ORPL will not only address the frame to its default parent but also to other parents called ‘candidate parents’ in its vicinity. Even if the default parent fails to hear this transmission, it may be possible that one or more of the candidate parents may successfully overhear it. In such a case, one of the candidate parents (e.g. the one closest to the destination) will acknowledge receipt of the frame. On overhearing the acknowledgement, the other candidate parents will discard their copy of the frame. The parent who

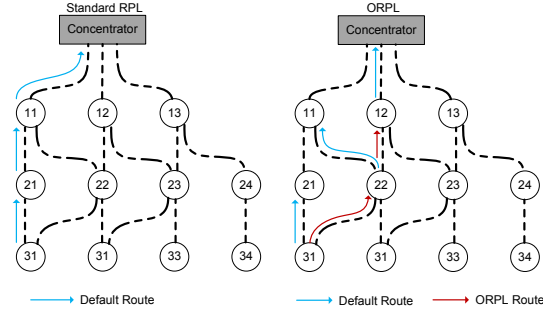


Fig. 1 Forwarding process in RPL and ORPL

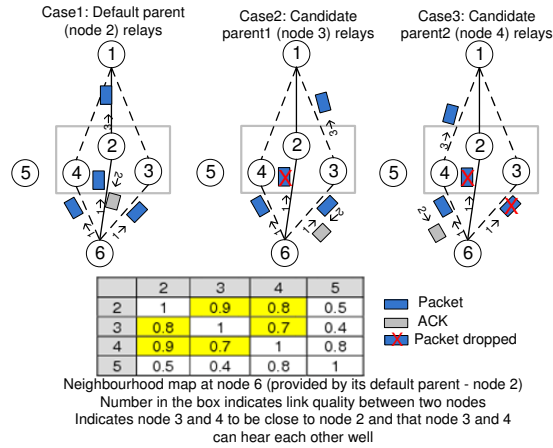


Fig. 2 ORPL - a neighborhood map and operational procedure

sent the acknowledgement will then relay the frame opportunistically on to its parent node en route to the concentrator thereby avoiding the need for a retransmission.

This principle, called anycasting, has been shown in [6] in the context of wireless ad hoc networks to be effective in improving the likelihood of frame delivery. However, in order to realize this in a low power and lossy network, several key issues need to be addressed. In particular, how to select the best set of candidate parents so as to minimize the transmission cost and how to coordinate these to reduce/eliminate duplicate transmissions from these parents thereby reducing the transmission overhead, a critical requirement especially in resource constrained sensor networks.

ORPL exploits the parent structure of the standard RPL protocol and employs overhearing based cooperation to coordinate the nodes involved in the forwarding process. The source node chooses a candidate parent set based on network state information made available to it by its default parent node. This is achieved in a standard compliant manner by carrying the network state information in the optional fields of the DIO control message employed by standard RPL.

Each node in an ORPL network maintains a neighborhood table. The metric used to create this table can be based on link statistics like Received Signal Strength Indicator (RSSI) or expected transmission count (ETX). The size of the neighborhood table depends on the target sensor platform. The propagation of neighborhood table to the client nodes is achieved using the DIO messages of RPL protocol.

ORPL gives different priorities to the nodes in its parent set depending on their distances (in terms of rank [20]) to the sink node. Upon receiving the frame from its child, the default parent forwards the frame towards the sink node. In this case, the candidate parents discard the copy of the forwarded frame when they overhear the default parents transmission. On the other hand, when the default parent fails to receive the frame successfully, the highest priority node in the receiving parent set carries out the forwarding operation on behalf of the default parent. The operation of the ORPL protocol can be visualized with the help of a simple example as shown in Fig. 2.

Neighbour Table Structure	
Field	Memory Requirement
Neighbour Address	8 bytes
Relative Neighbour Rank	2 byte
Neighbour RSSI	1 byte
Last Updated	1 byte

Parent Table Structure	
Field	Memory Requirement
Short Neighbour Address	4 bytes
Parent RSSI	1 byte

**Fig. 3** Memory requirements for parent and neighbor tables

### 2.3 Implementation Details for the Contiki Operating System

ORPL requires modification to both MAC and network layers of the protocol stack. Hence, it is advantageous to have access to an open source development platform where the necessary modules can easily be modified according to the protocol requirements. For this purpose, we used TelosB [2] devices with Contiki operating system as the development platform. Contiki [1] is an open source operating system specifically designed for low power sensor devices. Furthermore, it has a well maintained RPL version which implements a large subset of the RPL specification described in IETF RFC 6550 [20] which we used as a baseline for comparing performance with that of the ORPL protocol. In addition to providing support for running on target hardware, Contiki also provides an emulator called Cooja which facilitates development, testing and debugging of the code before running it on the target platform.

#### 2.3.1 Network Layer Modifications

One of the key components of ORPL is the neighborhood table as shown in Fig. 3. We selected RSSI as the main neighborhood metric since it is readily available at PHY layer of the TelosB hardware. RSSI metric indirectly represents the relative distance between two nodes. However, RSSI information is not sufficient on its own to decide whether a node can be qualified as an ORPL neighbor. In addition to RSSI, the receiving node should know the distance of a node to the sink node. This can be achieved using rank information of the RPL protocol (e.g., hop distance, etc).

If two nodes have a similar rank to the sink node and they are sufficiently close in terms of RSSI metric, each of these nodes adds the other as a neighbor to its local neighborhood table. While RSSI value does not always guarantee a high quality neighbor link due to fading, the average RSSI value can be used as an indication of the link quality. Furthermore, if the RSSI values of the neighbor nodes fall below a minimum expected RSSI value, they do not add each other as neighbor to their neighbor table. The neighborhood table is maintained in a way that a node adds only  $M$  neighbors with largest RSSI value with an acceptable rank range to its neighborhood table. The neighborhood table can be populated in the RPL objective function<sup>2</sup> module where the neighborhood metrics and rank values of the surrounding nodes are easily accessible.

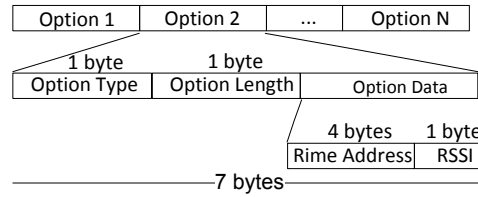
In our implementation, each node updates RSSI entries of its neighbors with the RSSI value of the frames received where the updated RSSI value is an exponential average<sup>3</sup> of the received RSSI values over a period of time. If a node does not hear from a node in its neighborhood table for a predefined period  $T$ , the entry corresponding to this node is removed from the neighborhood table. When a node having a better RSSI value appears, the receiving node updates its neighborhood table with the information of this new node. In the event that the neighborhood table is full, the neighbor with the worst RSSI value is removed from the table to make room for the new neighbor.

The neighborhood information is propagated to the client nodes using DIO option fields as shown in Fig. 4. It is to be noted that due to size limitation of the DIO option messages, only Rime address<sup>4</sup> [1] and RSSI values are propagated in DIO option messages as highlighted in Fig. 4. In this case, each node can reliably propagate information belonging to two of its best ORPL neighbors. While this can be seen

<sup>2</sup> Objective function in RPL protocol is used to calculate the rank of each node relative to the sink node in terms of a chosen link quality metric such as ETX or RSSI[20].

<sup>3</sup>  $RSSI_{avg} = \alpha * RSSI_{current} + (1 - \alpha) * RSSI_{avg}$ ,  $\alpha = 0.25$ .

<sup>4</sup> Rime is a lightweight communication stack designed for low-power radios.



**Fig. 4** DIO options for neighborhood map

as a limitation, it is adequate for propagating local neighbourhood map which will only be used by the client nodes with direct connection to the DIO sender. Here, the number of the neighboring meter nodes is expected to be less than the address space represented by 4-byte rime address.

Upon receiving a DIO from its default parent, each node checks if the reported Rime addresses in the DIO option fields are in its RPL parent list. If any of the reported Rime address can be located in the RPL parent list, it is added to the ORPL parent set. On the other hand, if a reported neighbor is not in the RPL parent list of the node, this means that the link to this particular node is not reliable and hence should not be added to the ORPL parent set. The ORPL parent set is implemented as a linked list where the candidate parents are sorted according to their RSSI values to the default parent. When a new candidate parent with a better RSSI is reported from the default parent, it is inserted to the linked parent list.

When any node changes its default parent, it frees its parent set and requests the neighborhood information of its new parent to form its new parent set. Although this process introduces a control overhead to the network, it makes sure that parent set information is kept up to date when such an event occurs. As each node keeps an up to date neighborhood table at the network layer, the MAC layer is responsible for using this neighborhood information to synchronize and control the opportunistic forwarding operation.

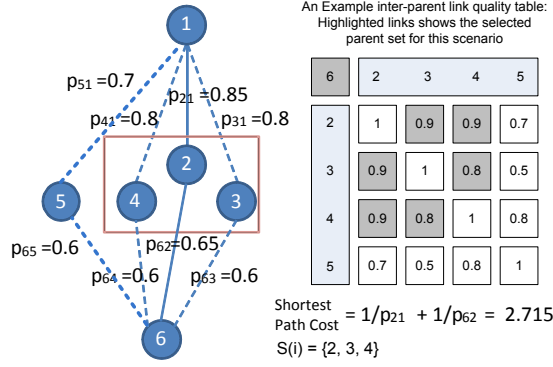
### 2.3.2 MAC Layer Modifications

The first responsibility of the MAC layer within an opportunistic forwarding scenario is to add candidate parent(s) addresses to the MAC header of the frame, if the relevant parent table contains alternative parents. Subsequently, the frame is transferred to the radio for transmission.

When an ORPL-enabled device receives a frame from the network, the receiving node first decodes the MAC header and extracts the address information added on top of the standard header. If the frame is destined for the receiving node, the node sends an ACK frame to the source and processes the frame normally. If the frame is not destined for the receiving node, it checks if the MAC header contains its address in the modified MAC header section. If this is the case, the node checks the location of its address in the header. If its address is in the first address location of the header, it sets an acknowledgement timer to wait for an ACK frame from the default parent. If its address is in the second address location, the waiting period is doubled. In this case, the lower priority node checks the candidate parent transmission as well as the default parent. Of course, doubling of the waiting time will introduce some overhead, but it may still be more advantageous than source node retransmission. When the highest priority node detects an ACK frame failure from the default parent, it sends an ACK frame to the source node, and forwards the stored MAC layer frame to the network layer. Upon receiving an ACK frame from any node in the parent table, all nodes that have stored the frame for transmission drop the acknowledged frame and cancel the ACK timer.

It may be possible to get multiple copies of the same frame at the candidate parents due to communication failures between the parent set and source node. In this case, each parent in the parent set keeps the sequence number of the received frame that any of the node in the parent set has acknowledged. If they receive a frame with the same sequence number, the frame is dropped in order to prevent redundant frame forwarding to the concentrator.

While using multiple candidate parents substantially reduces the loss probability in the first hop transmission, it makes the coordination of the nodes in the parent set increasingly complex which may result in more and more redundant frame forwarding due to cooperation failures. In such a case, it is of interest to evaluate the overhead introduced by ORPL.



**Fig. 5** A simple routing topology and corresponding neighbourhood map for node 6. The shortest path of node 6 corresponds to the route (6-2, 2-1).

## 2.4 ORPL Protocol Analysis

Firstly, let us analyze the overhead of ORPL to show that it is possible to attain significant performance improvement in terms of communication cost as compared to the non-cooperative approach.

### 2.4.1 Path Cost of ORPL

During an opportunistic forwarding process, the client node anycasts a frame to its parent set (PS). In this case, the probability of any node in PS receiving the transmission depends on the set size. We denote this set belonging to the node  $i$  as  $S(i)$ . ETX [7] is used in many routing protocols as a base for calculating the best single path route from a source to a destination. ETX of a link from node  $i$  to node  $j$  is given as

$$E_{ij} = \frac{1}{p_{ij}}, \quad (1)$$

where  $p_{ij}$  represents the probability of node  $j$  receiving a transmission from node  $i$ . The anycast probability,  $p_A$ , of a node  $i$  to its PS  $S(i)$  can then be calculated as

$$p_A = 1 - \prod_{j \in S(i)} (1 - p_{ij}). \quad (2)$$

In this case the anycast cost of node  $i$  to PS  $S(i)$  becomes

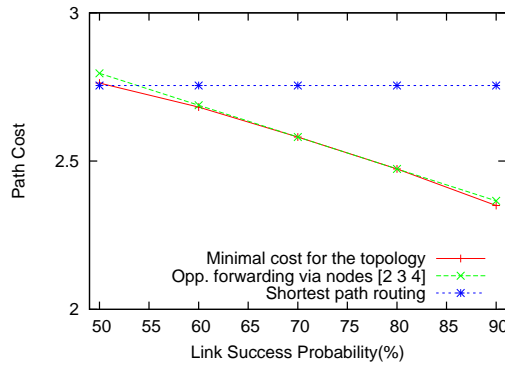
$$E_{iS(i)} = \frac{1}{p_A}. \quad (3)$$

These formulae assume independent reception error probabilities where a node is only affected by channel impairments such as noise and multi-path fading. The cost of an opportunistic route depends not only on the anycast cost, but also the aggregate cost of the remaining hops. The aggregate path cost can be visualised as the expected number of transmission required to reach a destination node from a source node. This has been addressed in [18] for wireless mesh networks where the cost of the remaining path of a node  $i$  is given as

$$R_{iS(i)} = \frac{C_1 p_{i1} + \sum_{j=2}^{|S(i)|} C_j p_{ij} \prod_{k=1}^{j-1} (1 - p_{ik})}{1 - \prod_{k \in S(i)} (1 - p_{ik})}, \quad (4)$$

where it is assumed that the PS are sorted by their costs to the sink node such that  $C_1 < C_2 < \dots < C_{|S(i)|}$ . While the remaining cost to the sink node for the proposed algorithm in this paper can be calculated using (4), the total cost of sending a frame from a node  $i$  to the sink node over the shortest anycast route can be expressed as

$$C_i^a = E_{iS(i)} + CO_{iS(i)} + R_{iS(i)}, \quad (5)$$



**Fig. 6** Impact of inter-parent link quality on the cost of the opportunistic route

where  $CO_{iS(i)}$  represents the cooperation overhead introduced by the ORPL protocol. The cooperation overhead can simply be defined as the probability of a node in PS retransmitting a frame when any other node in PS has already forwarded the frame to the next hop which can be approximated as

$$CO_{iS(i)} = \begin{cases} p_{ij}RC_j, & |S(i)| = 1 \\ \sum_{j=2}^{|S(i)|-1} p_{ij}RC_j \prod_{m=1}^{j-1} (1 - p_{im}), & |S(i)| > 1 \end{cases} \quad (6)$$

where the cooperation overhead is a weighted average of the erroneous forwarding probabilities of the nodes in the parent set. For example, when the default parent receives the forwarded frame, the probability of repeating the transmission is dominated by the node that has the highest chance of not overhearing the transmission by the default parent. Hence  $RC_j$  is given as

$$RC_j = R_{rd} \times (1 - p_{jr}), \quad (7)$$

where

$$r = \arg \max(1 - p_{jr}), \quad \text{for } r \in (S(i) \setminus \{1, \dots, j\}) \quad (8)$$

and  $R_{rd}$  represents the path cost from node  $r$  to the destination.  $S(i) \setminus \{1, \dots, j\}$  represents the set of nodes when the highest priority node which receives the transmission from the source node is  $j$ . Note that the nodes in the parent set are ordered according to their costs to the sink node. Therefore, we can assume that when a node  $j$  forwards a frame, all the nodes with a lower cost (higher priority) to the destination fail to receive the transmission from the source. For example, if node 3 in Fig. 5 forwards the frame, then this means that node 2 has failed to receive the transmission from node 6. In this case, only repetition may come from node 4 and this depends on the quality of the link between nodes 3 and 4.

Fig. 6 shows the total path cost for the shortest anycast route for the topology given in Fig. 5. The x-axis represent the inter-parent success probabilities which are assumed to be equal for the entire parent set. As evident from this figure, it is possible to attain a significant gain when the inter-parent success probability is high. However, the opportunistic forwarding gain becomes less significant with a higher inter-parent loss probability to the point that it provides no benefit for the given topology when the inter-parent success probability is less than 50%. This clearly indicates that the parent set should be selected in a way that the inter-parent link quality is above a certain threshold. Of course, it may not be beneficial at all to use cooperation in the case that the shortest path route has a high reliability.

#### 2.4.2 Optimal Parent Set Selection

In the previous section, we have shown that ORPL can lead to an overall reduced cost compared to non-cooperative systems when the inter-link success probability is reasonable. Moreover, we also observed that the final cost depends on the choice of the parent set. On that account, the optimal parent set that



would minimize the overall cost of the ORPL system can be obtained by solving the given optimization problem:

$$\begin{aligned} & \underset{S(i)}{\text{minimize}} && C_i^a \\ & \text{subject to} && |S(i)| \geq 1, \end{aligned} \quad (9)$$

where  $C_i^a$  is given in equation (5) and the first constraint ensures that at least one node is chosen as the default node. Given the non-linearity of the terms making up (5) and the combinatorial nature of the optimization problem, such an optimization problem would normally be classed as an NP-hard problem [18]. However, considering the memory constraint at each node, it is expected that the size of the parent set is upper bounded by a relatively small number. Consequently, an exhaustive search through all possible sets of nodes to determine the one leading to the minimal cost can be performed. As an example, we consider the system illustrated in Fig. 5, where the maximum parent set size is 4. By performing an exhaustive search through all possible subsets using Matlab, it was observed that the parent set leading to minimal cost consists of nodes [2, 3, 4, 5] when the inter-parent success probabilities were set to 0.9.

### 2.4.3 Protocol Efficiency

As elaborated in the protocol description, ORPL uses a single ACK message from the highest priority successful parent to realize the cooperation between the nodes in the parent set. While this feature aims to reduce protocol overhead, it may potentially introduce cooperation errors. An alternative approach could be to use an acknowledgement scheme where each node in the parent set needs to send an ACK message one after the other. However, such a protocol would use a fixed cooperative frame period which can be very inefficient when link quality is high.

Here, we compare the link efficiency of the ORPL protocol to the Extremely Opportunistic Routing Protocol (ExOR) introduced in [4] where, unlike ORPL, all the nodes in the parent set acknowledge a received frame regardless of their priority in the parent set. In particular, we want to analyze the time required, on average, for a successful frame transmission from a source node to any of the parent set nodes, by using ExOR versus ORPL and show that the latter offers a significantly better time efficiency.

*ExOR* In the ExOR protocol, the time between packet retransmissions is constant, say  $T_p$ , and equal to the delivery time for a packet,  $T$ , plus the waiting time for ACKs from all the parent nodes. If  $S(i) = \{1, \dots, K\}$  is the parent set, with the node numbered in order of decreasing priority, then  $T_p = T + KT_{\text{ack}}$ , where  $T_{\text{ack}}$  is the time needed to deliver a single acknowledgement.

Let us indicate with  $P_{\text{fail},j}$  the failure probability of parent node  $j$  to return an ACK

$$P_{\text{fail},j} = 1 - p_{ij}p_{ji}, \quad j = 1, \dots, K \quad (10)$$

where  $p_{ij}$  the success probability in the forward link from source  $i$  to node  $j$ , and  $p_{ji}$  is the success probability in the reverse link. Let us also introduce the probability,  $P_{\text{fail}}$ , that the source node does not receive ACK from any of the nodes in the parent set, which is given by

$$P_{\text{fail}} = \prod_{j=1}^K P_{\text{fail},j} = \prod_{j=1}^K (1 - p_{ij}p_{ji}). \quad (11)$$

The average time for a successful transmission,  $\bar{T}_{\text{suc}}$ , can be calculated as the product of the constant time  $T_p$  by the average number of transmissions required to deliver the packet successfully, i.e.,

$$\bar{T}_{\text{suc}} = \sum_{i=0}^{\infty} (i+1)T_p(1 - P_{\text{fail}})^i P_{\text{fail}} = \frac{T + KT_{\text{ack}}}{1 - P_{\text{fail}}}. \quad (12)$$

*ORPL* In the ORPL protocol the time between one unsuccessful transmission and the following retransmission is the same as for ExOR and equals  $T_p$ . However, the time required to complete the last (successful) transmission is variable as the source waits for the first ACK to arrive before sending the next packet. Note that with this protocol the first transmission of a packet may collide with a late ACK for the previous packet sent by some low priority node. This event may occur when a low priority parent node does not detect that an ACK was sent by a higher priority node whilst the source correctly receives that ACK. Hence the low priority node may send an ACK at the same time as the source is transmitting a new packet. If we neglect this small collision probability, the time required for a successful packet transmission is given by

$$T_{\text{suc}} = \eta T_p + (T + \beta T_{\text{ack}}), \quad \eta = 0, 1, 2, \dots, \beta = 1, \dots, K \quad (13)$$

where  $\eta$  denotes the number of unsuccessful retransmission and  $\beta$  the index of the parent node whose ACK has been correctly received after the last (successful) transmission. Because the two events,  $\eta = i$  and  $\beta = k$ , are independent, we obtain the following joint probability

$$P[T_{\text{suc}} = iT_p + (T + kT_{\text{ack}})] = P[\eta = i] \cdot P[\beta = k] = P_{\text{fail}}^i (1 - P_{\text{fail},k}) \prod_{j < k} P_{\text{fail},j}. \quad (14)$$

Therefore, the average time required for a successful transmission can be calculated as follows

$$\begin{aligned} \bar{T}_{\text{suc}} &= \sum_{i=0}^{\infty} \sum_{k=1}^K [iT_p + (T + kT_{\text{ack}})] P[T_{\text{suc}} = iT_p + (T + kT_{\text{ack}})] \\ &= \sum_{i=0}^{\infty} \left\{ (T + iT_p) P_{\text{fail}}^i \sum_{k=1}^K P[\beta = k] + T_{\text{ack}} P_{\text{fail}}^i \sum_{k=1}^K k P[\beta = k] \right\} \end{aligned} \quad (15)$$

We can expand the first sum in  $k$  as follows

$$\sum_{k=1}^K P[\beta = k] = 1 - \prod_{j \leq K} P_{\text{fail},j} = 1 - P_{\text{fail}}. \quad (16)$$

Moreover, if we assume that all  $P_{\text{fail},j}$  are equal to  $P_f$ , we can expand the second sum in  $k$  as follows

$$\begin{aligned} C &\triangleq \sum_{k=1}^K k P[\beta = k] = \sum_{k=1}^K k (1 - P_f) P_f^{k-1} \\ &= (1 - P_f) \left[ \frac{1 - P_f^K}{(1 - P_f)^2} - \frac{K P_f^K}{1 - P_f} \right]. \end{aligned} \quad (17)$$

Therefore, from (15), after some simple passages we obtain

$$\bar{T}_{\text{suc}} = \frac{T + K T_{\text{ack}} (P_{\text{fail}} + \frac{C}{K})}{1 - P_{\text{fail}}}. \quad (18)$$

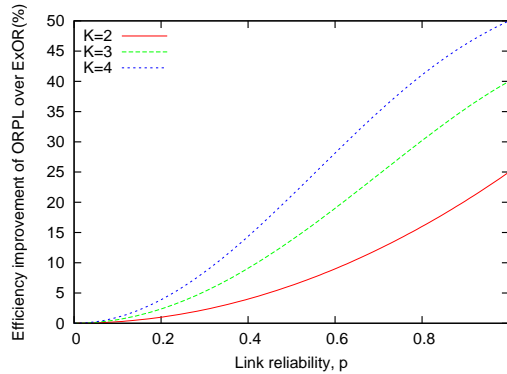
By comparing (12) and (18) we see that for small failure probability, i.e., for  $P_{\text{fail}} \rightarrow 0$ , the term  $C$  of (17) tends to 1 and the expressions (12) and (18) tend to

$$\bar{T}_{\text{suc}}(\text{ExOR}) \xrightarrow{P_{\text{fail}} \rightarrow 0} T + K T_{\text{ack}}, \quad (19)$$

$$\bar{T}_{\text{suc}}(\text{ORPL}) \xrightarrow{P_{\text{fail}} \rightarrow 0} T + T_{\text{ack}}. \quad (20)$$

Therefore we can conclude that ORPL saves on average  $(K - 1)T_{\text{ack}}$  in time compared to ExOR for highly reliable links.

Fig. 7 shows the time efficiency improvement in percentage of ORPL over ExOR, i.e.,  $1 - \bar{T}_{\text{suc}}(\text{ORPL}) / \bar{T}_{\text{suc}}(\text{ExOR})$ , as a function of the success probability on each link, in the fully symmetrical case with  $p_{ij} = p_{ji} = p$  for all  $j \in S(i)$ .



**Fig. 7** Time efficiency improvement of ORPL over ExOR protocol for different parent set size  $K = 2, 3, 4$ .  $T = 1ms$ ,  $T_{ack} = 0.5ms$ .

## 2.5 Performance Evaluation of ORPL

### 2.5.1 Evaluation Scenarios

The performance of ORPL protocol has been evaluated against standard RPL using three different scenarios.

*Scenario 1:* The first scenario consists of 30 nodes manually placed (*planned deployment*) in a 100 meter by 100 meter area where each node has at least two or more candidate parent nodes and the furthest client nodes are five hops away from the concentrator. The purpose of this scenario is to highlight the benefit of ORPL in a planned deployment scenario.

*Scenario 2:* In this scenario, 70 nodes are distributed uniformly (*unplanned deployment*) in a 200 meter by 200 meter area. The objective in this case is to get a better understanding of the ORPL protocol in a realistic (likely to be ad hoc) deployment. Here, the nodes may or may not have multiple candidate parents. In such a scenario, ORPL may not provide as much benefit as compared to the previous scenario.

*Scenario 3:* This scenario comprises of a planned deployment of 20 nodes where all of the client nodes at second and third hops have 2 or 3 ideally placed parent nodes. In this scenario, the furthest client nodes are 3 hops away from the concentrator. The objective of this scenario is to show the packet delivery ratio performance of ORPL protocol against RPL protocol when MAC layer retransmissions are controlled by the client node depending on the forward channel quality.

All of the simulations have been carried out for different link success probabilities where the link success depends on the distance as given by the free-space path loss formula [3]. For example, a scenario with link success rate of a link being 50% can be visualized as one in which two nodes are at the maximum distance where they can just about communicate with each other. On the other hand, when the nodes are just besides each other (less than 1 meter apart), the actual link success rate becomes around 100% for the same scenario. The results presented show the aggregate values of the performance metrics (considered in this study) for the whole network after 100 frame transmissions.

The following metrics have been used in this study to compare the performance of ORPL with that of standard RPL:

*Packet delivery ratio (PDR) with and without MAC layer retransmissions* - the ratio of the number of packets received to the total number of packets sent. This metric aims to capture what fraction of packets sent by the sources are actually delivered to the destination.

*Number of retransmissions* - If a transmission is unsuccessful in ORPL, other nodes may opportunistically try to send the same frame upon sensing this. This should reduce the need for retransmissions in comparison to standard RPL wherein unsuccessful transmissions will lead to retransmissions. This metric represents aggregate number of retransmissions in the network averaged over the number of experiments run. Furthermore, when ORPL is used, number of retransmissions represents the transmissions due to packet failures plus the retransmissions due to cooperation errors.

*Cooperation overhead* - the ratio of number of duplicate frames to the total number of frames received at the destination. It will be appreciated that in practice it may be difficult to achieve perfect cooperation, e.g., even though the default parent is successful in forwarding the frame, the candidate parent may still transmit the same frame. This could happen for instance if the candidate parent fails to hear an ACK

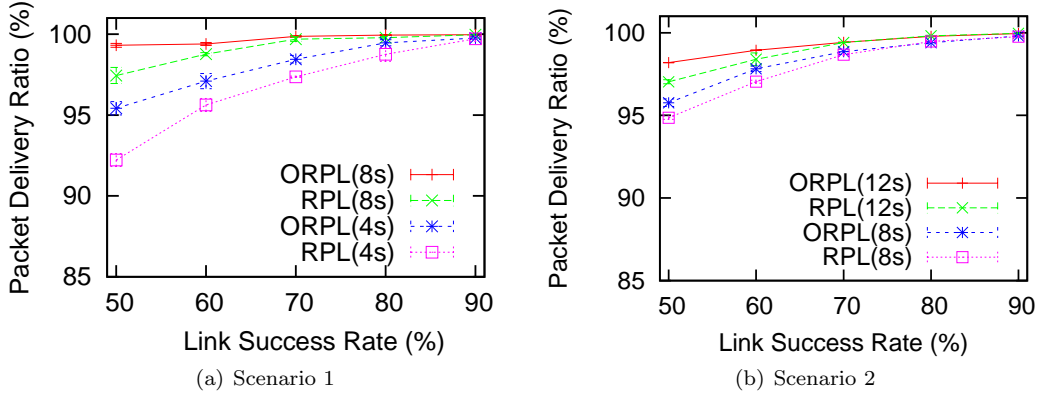


Fig. 8 PDR for different link success rates

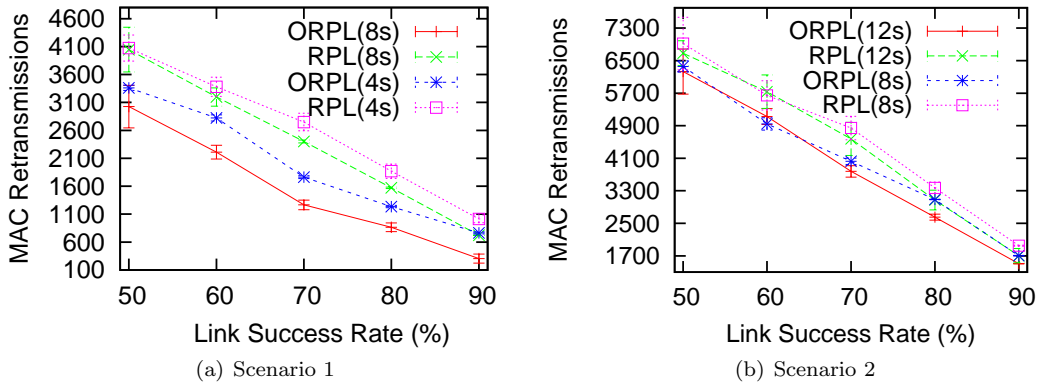


Fig. 9 Retransmissions for different link success rates

from the default parent. Transmission of such duplicate frames leads to a reduced network throughput. This metric tries to quantitatively capture such overhead.

### 2.5.2 Performance Results

Fig. 8(a) and 8(b) show the PDR performance attained by ORPL in comparison to RPL in scenarios 1 and 2 respectively where the nodes are configured to transmit meter readings for high contention and low contention traffic load conditions. In high contention conditions, depicted by the 4s metering interval experiments for the 30 node network and 8s metering interval experiments for the 70 node network, nodes contend for the scarce radio resource. Under low contention case, depicted by the 8s metering interval experiments for the 30 node network and 12s metering interval experiments for the 70 node network, the channel access delay is significantly reduced. When the network is in high contention, packet drops happen due to both interference and errors in the wireless channel. On the other hand, packet drops typically occur due to channel errors in the low contention traffic scenarios.

It is evident from Fig. 8(a) and 8(b) that ORPL achieves better PDR in comparison to RPL. The performance benefit is significant under low contention scenario with low link success rate. In such a scenario, ORPL achieves diversity over the number of receiving parent nodes (through opportunistic transmissions) which not only improves the delivery probability but also provides the same with much lesser number of MAC layer retransmissions as shown in Fig. 9(a) and (b) (20-50% lower than that of RPL for planned and random deployment scenarios). On the other hand, ORPL shows a less impressive MAC layer retransmission performance under high contention where it achieves 10 to 40% lower MAC retransmissions as compared to RPL. In this case, the MAC layer retransmissions and coordination errors caused by simultaneous transmissions in the wireless medium limit the ORPL performance. This in turn increases the duplicate frame transmissions. This behavior is confirmed by the cooperation overhead figures presented in Fig. 10(a) and 10(b).

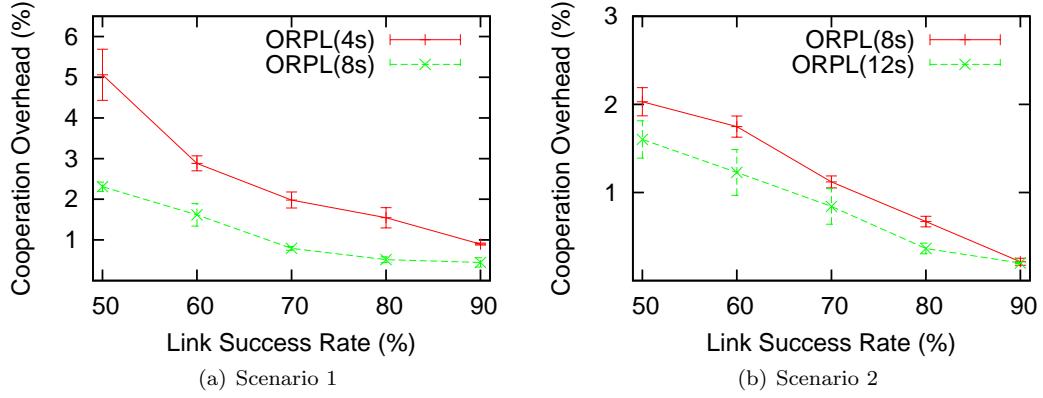


Fig. 10 Cooperation overhead of ORPL

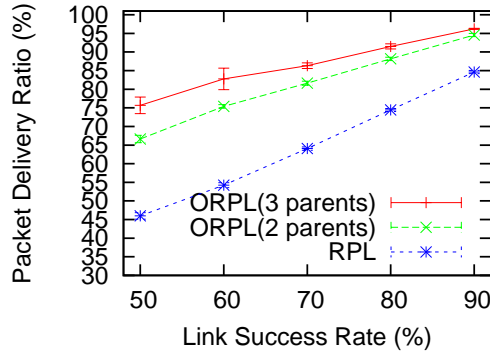


Fig. 11 PDR in scenario 3 (20 nodes, retransmissions disabled)

Finally, we also investigate the implications of disabling retransmissions at the MAC layer (i.e. no retransmission subsequent to a loss) on the performance of both RPL and ORPL. Fig. 11 shows the PDR attained by both RPL and ORPL in scenario 3 with a 10 seconds metering interval. We observe that in the absence of MAC layer retransmissions, ORPL performs significantly better in comparison to RPL for both 2 and 3 parents where ORPL performance improves with increasing number of parent nodes. This is due to the fact that the more the number of parent nodes is, the less is the anycast loss probability. We observe that nodes taking advantage of ORPL attains on average 25-35% better PDR compared to RPL. Furthermore, a PDR of around 90% is achievable for a link success rate of 80% when ORPL (3 parents) is employed. The latter result indicates that it is possible to reduce the network traffic further by employing an adaptive MAC layer protocol.

### 3 ORPL<sub>x</sub> : ORPL with Adaptive MAC Layer Retransmit Limits

ORPL greatly improves reliability and scalability of an AMI mesh network by reducing the MAC layer transmissions. It is clear from the results in section 2.5.2 that ORPL achieves a reduction of up to 50% in MAC layer traffic. However, the results for cooperation overhead indicates a relatively high number of duplicate frame reception at the cooperating parent nodes, especially in poor wireless channel conditions. The ORPL protocol aims to reduce protocol overhead by a single ACK frame from the highest priority successful parent as explained in Section 2. This creates asymmetric reception probabilities of the data and acknowledgement frames. As a result, it may be possible that a frame is retransmitted unnecessarily due to ACK failures. Here, we propose a novel MAC layer extension for ORPL to overcome this problem. We call this new protocol 'ORPL<sub>x</sub>'.

### 3.1 Adaptive MAC Layer Retransmit Limits

The ideal MAC retransmit limit in an ORPLx configuration depends on several factors such as the number of nodes in the parent set, the quality of the links from the source to parents and the desired level of reliability (i.e. link success rate). An example scenario could be such that reliability of high frequency metering data may be sacrificed in order to reduce the traffic in the network. In this case, it may be possible to classify the sensor data to different priority levels and adjust MAC layer retransmit limits according to these priority levels. This approach is similar to the methods proposed in [15] and [5] where the MAC layer retransmit limit is modified to meet the quality of service requirements of UDP video traffic. For ORPLx, we need to enable such a mechanism through which the MAC layer should be able to calculate the number of retransmissions for a target link success rate (i.e. PDR) by taking into account the anycast channel quality.

To this end we consider a source node  $i$  sending a message to a number of nodes included in the set  $S(i)$ . For each node  $j \in S(i)$ ,  $p_{ij}$  is the probability of correctly receiving the message from the source node. We want to calculate the minimum number of retransmissions required to guarantee that at least one node in the set receives the message correctly with probability not smaller than a target  $p_t$ .

The anycast probability, i.e., the probability of at least one node receiving the message correctly after a single transmission was introduced in (2). Therefore, the probability of at least one node receiving the message correctly after  $k \geq 1$  transmissions<sup>5</sup> is given by

$$\sum_{n=1}^k \binom{k}{n} p_A^n (1 - p_A)^{k-n} = 1 - (1 - p_A)^k \geq p_t, \quad (21)$$

where we impose the constraint that the probability is not smaller than the predefined target. By taking the logarithmic at both sides of the inequality we obtain the minimum number of retransmissions required to meet the target success probability as a function of the anycast probability:

$$k_{\min} = \max \left\{ 1, \left\lceil \frac{\log(1 - p_t)}{\log(1 - p_A)} \right\rceil \right\}. \quad (22)$$

Naturally, implementing a logarithmic function on a low power sensor platform is not feasible. By truncating the power series expansion of the logarithmic functions to the second term we can approximate (22) as

$$k_{\min} \approx \max \left\{ 1, \left\lceil \frac{p_t + \frac{p_t^2}{2}}{p_A + \frac{p_A^2}{2}} \right\rceil \right\}. \quad (23)$$

In our implementation we have used a slightly modified version of (23), as reported in the next section.

Reliably calculating the number of retransmission is a challenging task in a real deployment. In order to achieve this, the MAC layer should be able to estimate/calculate individual link success probabilities of the anycast links  $p_{ij}$ . Here, it should be pointed out that calculating forward link error probability for low power and lossy networks is on its own another research topic. Therefore, it is not covered in this paper. But, interested reader is referred to our recent related work in [10]. On the other hand, a straight-forward<sup>6</sup> way to do this is to collect MAC layer PDR statistics for different receive power levels for the target propagation environment.

#### 3.1.1 Empirical Calculation of MAC Layer Retries

We used two TelosB motes configured as a client-server pair to calculate the link level statistics empirically. The MAC layer acknowledgement mechanism of the receiver was disabled to calculate the PDR of the link for frames with 60 bytes payload. In this configuration, the client node was configured to send sequential frames marked with a counter periodically. The last received counter value was then divided by the total number of received frames. Consequently, the PDR results were averaged for different receive SNR levels which were used to estimate the individual link success rates.

Table 1 presents the PDR levels ( $p_{ij}$  in Eq. (2)) for the experimental and the Cooja based simulation studies. In the experimental study, nodes were placed far apart in our office environment and the transmit

<sup>5</sup> i.e., one initial transmission and  $k - 1$  retransmissions.

<sup>6</sup> But, this approach is not very accurate due to interference effects and dynamic nature of wireless channel.

**Table 1** Receive power levels vs. link success probabilities

Office Experiments					
P(dBm)	[-90 -85]	[-85 -80]	[-80 -75]	[-75 -70]	[-70 -55]
$p_{ij}$ (%)	75	85	95	98	99
Cooja Experiments					
P(dBm)	[-95 -87]	[-87 -70]	[-70 -55]	[-55 -45]	[-45 -35]
$p_{ij}$ (%)	70	75	83	93	98

power of the sender and receiver nodes was changed for empirically calculating PDR statistics for the receive power levels ranging from -90 dBm to -70 dBm. The average PDR results for the experimental study were calculated by running each experiment<sup>7</sup> during the office hours on an interference free channel (i.e., channel 26 of 802.15.4). On the other hand, Cooja uses a free-space path loss model [3] where link success depends on the distance between the source and destination. In this case, the average PDR statistics were calculated by running experiments for different transmitter receiver separations and results were averaged to get PDR levels of Table 1.

### 3.1.2 MAC Layer Modifications

In order to estimate the quality of the links between the client node and its parents, each client node in an RPL network is required to collect received power levels for all the nodes in its parent set. To achieve this, a linked list is used to store average receive power levels for each link from a client node to its parents where an exponential averaging process<sup>8</sup> is used.

$$k_t = \begin{cases} 1, & \text{for } \theta < 1 \\ \lceil \theta + 1 \rceil, & \text{for } 1 \leq \theta < 1.5, \\ \lceil \theta + 1 \rceil, & \text{otherwise} \end{cases} \quad (24)$$

where

$$\theta = \frac{p_t + \frac{p_t^2}{2}}{p_A + \frac{p_A^2}{2}}. \quad (25)$$

When a client node needs to send a data frame to the concentrator node, it needs to calculate the retransmit limit for the target link success threshold  $p_t$ . Firstly, the client node retrieves stored received power levels of the the parent set. These values are used to look up the empirical link statistics for all the links from the client node to the parent nodes. Consequently, the client node approximates the retransmission limit for the target link success rate  $p_t$  using a slightly modified version of Eq. (23) as given in Eq. (24). The reason for this modification is the fact that the retransmission value  $k_{min}$  in Eq. (23) is a coarse approximation of the retransmission limit. Furthermore, other factors such as interference affects the success of the delivery. Therefore, in ORPLx, we use Eq. (24) to cater for approximation errors introduced by Eq. (23).

As shown in Eq. (24), we use a retransmission limit of 1 only when the estimated number of retransmissions is less than 1. This means that the transmission of the frame will be successful with a probability greater than the target success probability  $p_t$ . On the other hand, when the estimated retransmit limit is greater than 1, an extra transmission is performed upon an ACK failure.

## 3.2 Performance Evaluation of the ORPLx Protocol

The performance of the ORPLx protocol is compared against ORPL and RPL protocols using the following scenarios:

<sup>7</sup> For a particular transmit power level.

<sup>8</sup>  $RSSI_{avg} = \alpha * RSSI_{current} + (1 - \alpha) * RSSI_{avg}$ ,  $\alpha = 0.25$

*Scenario 1:* This scenario is realized by emulating real TelosB hardware in Contiki Cooja emulator where the network topology of Scenario 3 of Section 2.5 was used for evaluating the performance of the proposed protocol extensions.

*Scenario 2:* This scenario comprises of a planned deployment of 120 client nodes that utilises the map data of a neighborhood located in city of Bristol. It is assumed that each house is equipped with a smart electricity meter. The objective of this scenario is to analyse the performance of the proposed extensions in a realistic deployment scenario.

*Scenario 3:* This scenario consists of two experimental test-bed topologies consisting of 5 and 32 TelosB motes respectively.

The aim of the 5 node topology is to highlight the impact of channel variations on the performance of the proposed extensions without any outside interference where main cause of packet drops is expected to the link variation caused by movement of people in the office floor. Furthermore, this simple and controllable topology is used to vary the transmit power of the nodes to see the impact that has on the diversity gain of the proposed protocol. In this scenario, four of the motes are configured as client and one mote is used as the root node. The nodes are placed in different sections of our office environment. The server node is connected to a laptop at one end of the 30 meter long 'L' shaped office floor. One client node is placed in the furthest location from the server node and the remaining client nodes are placed in the middle 3 meters apart on a horizontal line in order to make sure that these 3 nodes act as the parent set for the client node.

The main objective of the 32 node scenario is to analyse the performance of the ORPL and ORPLx protocols as compared to RPL for a realistic deployment case. Here, the packet drops in the intermediate links can be caused by channel variations and contention in the network. In this case, the client nodes are evenly distributed over two floors of our office environment and the root node is placed at the same position as the 5 node topology.

The test-bed experiments are run for a whole day during office hours on 802.15.4 channel 26. While one may argue that using a channel isolated from WiFi interference may not be realistic, this gives us the opportunity to analyse the performance of the system in relative isolation of the interference caused by WiFi networks. It is expected that the performance of opportunistic routing schemes are negatively affected by WiFi interference since the links losses are likely to be correlated as highlighted in [19].

In our experiments, ORPL protocol is configured to use a maximum of three parent nodes for forwarding the traffic originating from the client nodes. Each client node periodically sends a frame of 60 bytes to the concentrator representing a meter reading. The scheduling of the readings is done by dividing the meter reading intervals into time slots that are mapped to sensor node IDs<sup>9</sup>. When a client node fails to transmit a frame due to an ACK failure, the MAC layer schedules the frame with a linear back off timer.

The simulation scenarios are run for 10 times with different random seeds where the simulations last until all of the clients have transmitted 100 frames. The results for experimental study represent the average performance for the day normalized to 100 frame transmissions. All of the reported results include only the nodes using cooperative communication where the results for the non-cooperating last hop nodes (one hop away from the concentrator) are not considered. The target link success rate  $p_t$  is set to 99%.

### 3.2.1 Performance Results

The simulation studies aim to analyze the contention and interference performance of the ORPLx protocol in a controlled environment. On the other hand, the main objective of the experimental scenarios are to analyze the performance of ORPLx in a realistic channel realisation. In this case, communication errors are likely to be caused by the wireless channel variations and the contention in the network.

Fig. 12(a) and 12(b) show the PDR performance of ORPLx in Scenario 1 as compared to RPL and ORPL for 5 and 15 seconds metering intervals. The results for 5 seconds metering interval show that the performance of ORPLx suffers as compared to ORPL and RPL. This is mainly due to interference caused within the network where concurrent transmissions of non-neighboring nodes (i.e, hidden terminal problem) cause ORPLx to achieve a PDR 3% less than the minimum target link success rate of 99%. In fact, the link success probabilities used to calculate the anycast probability  $p_A$  in (24) are estimated by collecting PDR statistics for different receive power level in the absence of interfering transmissions.

<sup>9</sup> Each node has a unique ID



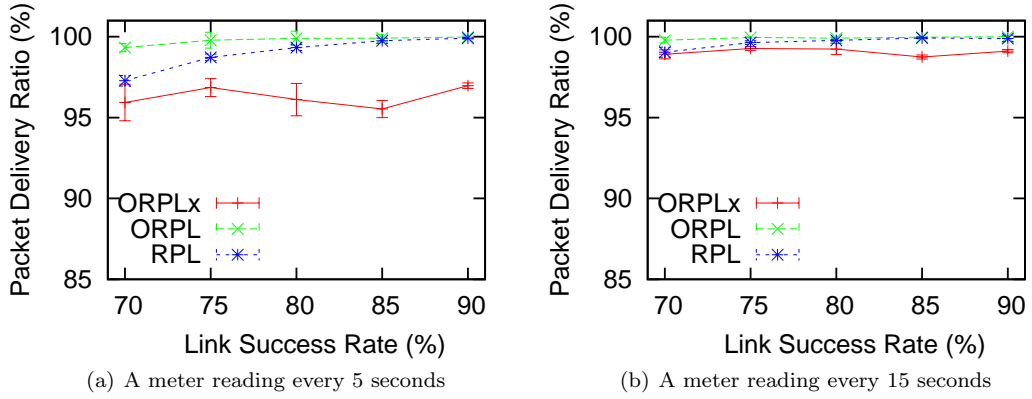


Fig. 12 PDR performances of RPL, ORPL and ORPLx for scenario 1

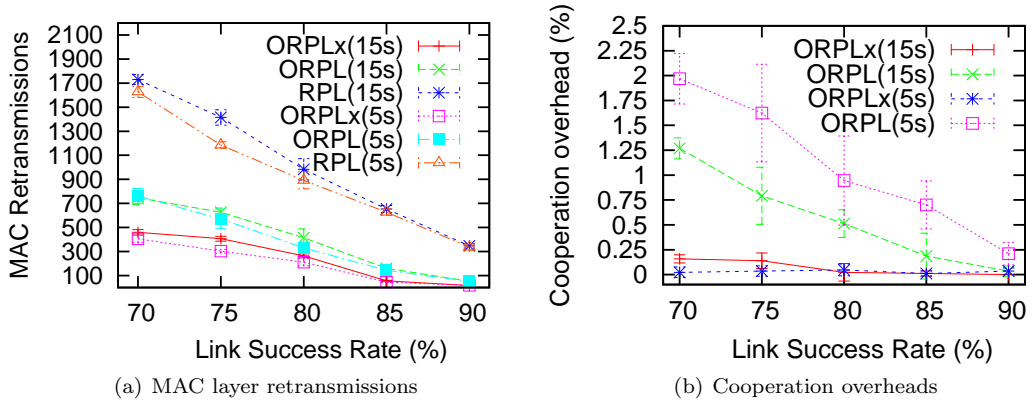


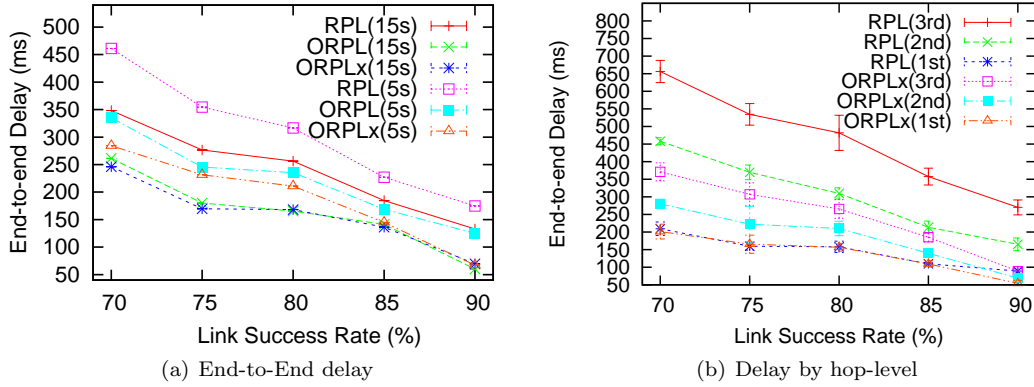
Fig. 13 Retransmissions and Cooperation overheads for saturated and unsaturated traffic loads

Therefore, these estimates tend to be optimistic when a large number of transmissions (5 seconds interval) are taking place. Nevertheless, even for such a scenario with relatively higher contention, ORPLx is able to achieve an average PDR of around 96% which can still be used for low priority metering data. On the other hand, ORPLx achieves a much higher average PDR for the scenario where the contention is lower. In this case, PDR results are around 99% for every link quality level which is in line with the minimum  $p_t$  of 99%. Here, we should highlight that the performance of ORPLx depends on accuracy of the forward link quality estimation which is tightly related to the performance of the estimation process [10].

The MAC layer retransmission results for Scenario 1 are given in Fig. 13(a) for both 5 and 15 seconds metering intervals. The results show that while ORPL provides around 50% reduction in terms of MAC layer retransmissions as compared to RPL, ORPLx provides a further 50% reduction as compared to ORPL. Hence, ORPLx achieves four to five fold reduction in number of MAC retransmissions on average as compared to RPL. On the other hand, ORPL performance is close to that of ORPLx for high quality links. This confirms our initial observation for retransmissions due to MAC layer ACK failures. Furthermore, we can see that the number of retransmissions for ORPLx approaches zero at 85% link success rate for both high and low contention scenarios, which indicates that the cooperative link from the client to the parent nodes is able to support 99% reliability at 85% link success rate with a single transmission with the assumption that there is no contention in the wireless medium.

Fig. 13(b) gives the cooperation overhead of ORPLx and ORPL for high and low contention scenarios. It is evident from the figure that ORPLx is able to eliminate the majority of duplicate frame transmissions. This means a more efficient network operation with fewer number of MAC layer frame transmissions.

The Fig. 14(a) presents end-to-end application layer delay performance of the network given in Scenario 1. And Fig. 14(b) gives the delay results the client nodes experience at different hop levels for the same scenario. Please note that the implementation of the proposed protocol extensions are software based and this impacts the end-to-end performance results depending on the complexity of the protocol. For example, RPL has the most basic MAC layer implementation with no modification to network



**Fig. 14** Delay performances of RPL, ORPL and ORPLx for scenario 3

**Table 2** PDR and MAC Retransmissions Results for the Example Deployment Scenario

	RPL	STDEV(RPL)	ORPL	STDEV(ORPL)	ORPLx	STDEV(ORPLx)
PDR(%)-15s	94.34	0.8	98.07	0.7	94.04	0.8
MAC Tx-15s	9230	208	6220	343	5317	383
PDR(%)-30s	98.89	0.2	99.25	0.4	96.73	0.24
MAC Tx-30s	8972	72	6625	276	5134	98

layer. On the other hand, ORPL and ORPLx requires modifications to MAC and Network Layers of the protocol stack to enable cooperative forwarding. Especially, ORPLx needs to calculate the forward link quality for each and every data packet transmitted. This introduces some extra delay to the link. Therefore, the reported delays here for ORPL and ORPLx are expected to be better when these protocols are implemented in a dedicated hardware.

It is apparent from the figures that ORPL and ORPLx reduce the delay experienced in the network substantially. Furthermore, according to the delay results of Fig. 14(b), the end-to-end delay is reduced in the hop levels where the nodes can take advantage of cooperative communications (i.e. second and third hops). These results confirm the predictions of the theoretical analysis presented in Section 2.4.

Fig. 15 gives an example scenario based on the geographical data belonging to a neighborhood of the city of Bristol. There are 120 client nodes (in green) and a concentrator node (in orange) in this scenario. And the distances between the client nodes are designed such that they reflect the real distance between the houses for this particular neighborhood.



**Fig. 15** A neighborhood deployment scenario for the city of Bristol

The radio range of the used hardware is adequate to enable a fully connected network. The nodes in this scenario have a PDR level of 70% when they are at the furthest distance that they can communicate. Table 2 gives the PDR results of RPL, ORPL and ORPLx protocols for 15 and 30 seconds metering intervals. ORPL and ORPLx reduce the number of transmissions in the network for both 15 and 30 seconds metering intervals. Furthermore, ORPL achieves around 4% PDR improvement in 15 seconds scenario as compared to RPL. This increase can be attributed to the reduced network traffic achieved by

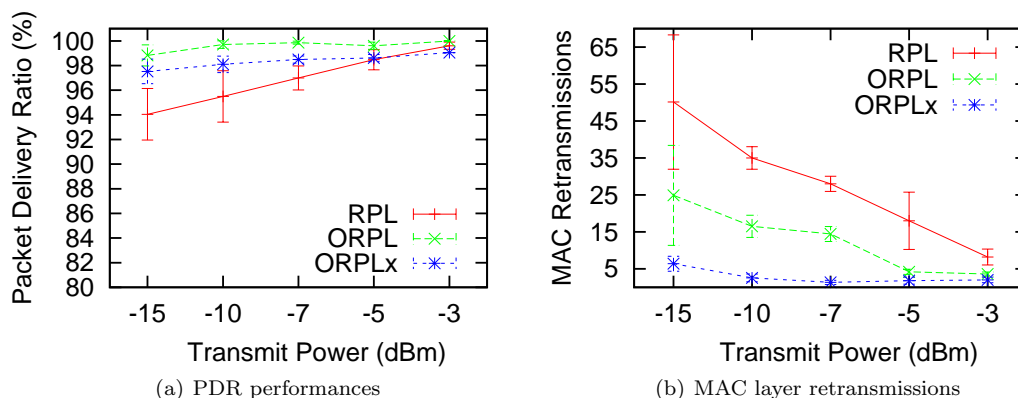


Fig. 16 5-node experimental results for 5 seconds meter reading interval

Table 3 PDR and MAC Retransmissions Results for the 32 Node Experimental Scenario

	RPL	STDEV(RPL)	ORPL	STDEV(ORPL)	ORPLx	STDEV(ORPLx)
PDR(%)-10s	97.80	0.98	99.40	0.32	96.80	1.05
MAC Tx-10s	257	74.68	131	22	75	6
PDR(%)-20s	99.2	0.3	99.56	0.1	99.1	0.24
MAC Tx-20s	155	50	108	15	61	9

ORPL protocol. On the other hand, ORPLx shows a slight drop in PDR relative to RPL in this scenario. This is due to the packet drops caused by interference which is not incorporated into the forward error probability calculation in this study. Nevertheless, the results here show that ORPL and ORPLx can be utilized to create a low overhead and reliable AMI network.

While simulation based experiments provide a good insight to the performance of the proposed extensions, it is of interest to look into real life performance of the algorithms in a test-bed deployment. Fig. 16(a) shows the PDR performances of RPL, ORPL and ORPLx for Scenario 3. We observe from the figure that ORPLx achieves a PDR performance of around 98 to 99% for all the transmit power levels (i.e. different receive SNR levels). RPL, on the other hand, suffers for the most power settings and gives a lower PDR compared to ORPL and ORPLx. This can be explained by the fact that during the office hours people move around the office and cause significant fading in the propagation medium. This also explains high deviation in the PDR results where the movement patterns of people vary greatly depending on the time of day. As expected, ORPL achieves the best performance results out of the three at the expense of higher number of MAC layer transmissions as highlighted in Fig. 16(b). On the other hand, ORPLx achieves a significant reductions in number of retransmitted frames as compared to ORPL and RPL.

Table 3 shows the PDR and MAC layer retransmission results for the 32 node experimental scenario for 10 and 20 seconds metering intervals. The results show that ORPL and ORPLx achieves significant reduction in network traffic for a real life deployment. On the other hand, when the contention in the network is high, ORPLx shows a slight drop in PDR. In this case, it has to be noted that ORPLx performance heavily depends on the reliability of the forward link quality estimation. Hence, the drop in performance can be attributed to this fact.

## 4 Conclusions

Self-organizing wireless mesh networks are low cost, easy to set up and maintain, and therefore attractive for realizing a practical AMI deployment. In this paper, we have proposed an opportunistic communication mechanism called Opportunistic RPL (ORPL) to improve the reliability of data transport in AMI mesh networks. We have studied the performance of ORPL through mathematical analysis, implemented it as part of Contiki operating system and investigated its behavior through emulation and deployment based experiments. Results from this study have demonstrated the effectiveness of ORPL in terms of superior packet delivery ratio and lesser number of retransmissions in comparison to RPL while achieving this

with low protocol overhead. We have also proposed an extension to ORPL which aims to reduce number of MAC layer retransmissions further to improve the efficiency of the protocol. ORPLx provides two and four fold reductions in terms of MAC layer traffic as compared to ORPL and RPL respectively. In the meantime, it achieves the desired link layer delivery targets when the contention in the network is low.

We conclude that using ORPL can be a way forward to improving communication reliability in scenarios where nodes experience high losses. In particular, for scenarios involving low priority data that may not require high reliability, it is possible to accommodate more smart metering nodes in a single concentrator's network by using ORPLx. Future work could look at the performance of ORPL in duty cycled mesh networks. It is also of interest to look at scheduling mechanisms that can reduce contention and interference within the AMI mesh networks. It should be noted that apart from smart metering, the protocol enhancement proposed in this paper can also be used in other M2M (machine to machine) type applications, such as e-health, intelligent transport systems, and smart infrastructure monitoring.

## References

1. Contiki: The Operating System for Connecting the Next Billion Devices - the Internet of Things. <http://www.sics.se/contiki/> (2010)
2. TELOSB Mote Platform. [http://www.willow.co.uk/html/telosb\\_mote\\_platform.html](http://www.willow.co.uk/html/telosb_mote_platform.html) (2010)
3. Balanis, C.A.: Antenna Theory: Analysis and Design, Third edn. John Wiley and Sons Inc. (2005)
4. Biswas, S., Morris, R.: Opportunistic routing in multi-hop wireless networks. *SIGCOMM Comput. Commun. Rev.* **34**(1), 69–74 (2004). DOI 10.1145/972374.972387
5. Chen, C., Lin, C., Chen, Y.: Cross-layer packet retry limit adaptation for video transport over wireless LANs. *IEEE Trans. CAS for Video Technology* **20**(11) (2010)
6. Choudhury, R., Vaidya, N.: MAC-layer anycasting in ad hoc networks. *ACM SIGCOMM CCR* **34**(1) (2004)
7. Couto, D., Aguayo, D., Bicket, J., Morris, R.: A High-Throughput Path Metric for Multi-Hop Wireless Routing. In: *ACM MOBICOM* (2003)
8. Duquenooy, S., Landsiedel, O.: Poster abstract: Opportunistic rpl. In: *Proceedings of the 10th European Conference on Wireless Sensor Networks* (2013)
9. Fan, Z., Kulkarni, P., Gormus, S., Efthymiou, C., Kalogridis, G., Sooriyabandara, M., Zhu, Z., Lambbotharan, S., Chin, W.H.: Smart grid communications: Overview of research challenges, solutions, and standardization activities. *IEEE Communications Surveys and Tutorials* **15**, 21–38 (2013)
10. Gormus, S., Bocus, M.Z.: Efficient cooperative anycasting for ami mesh networks. In: *10th IEEE Global Communications Conference, Atlanta, GA, USA* (2013)
11. Gormus, S., Fan, Z., Bocus, Z., Kulkarni, P.: Opportunistic Communications to Improve Reliability of AMI Mesh Networks. In: *IEEE Innovative Smart Grid Technologies Europe (ISGT EU) conference, Manchester, UK* (2011)
12. Iwao, T., Yamada, K., Yura, M., Nakaya, Y., Cardenas, A., Lee, S., Masuoka, R.: Dynamic data forwarding in wireless mesh networks. In: *IEEE Smart Grid Communications Conference, Gaithersburg, MD, USA* (2010)
13. Iyer, G., Agrawal, P., Monnerie, E., Salazar, R.: Performance Analysis of Wireless Mesh Routing Protocols for Smart Utility Networks. In: *IEEE Smart Grid Communications Conference, Brussels, Belgium* (2011)
14. Laneman, J.N., Wornell, G.W., Tse, D.N.C.: An Efficient Protocol for Realizing Cooperative Diversity in Wireless Networks. In: *IEEE International Symposium on Information Theory (ISIT)* (2001)
15. Li, Q., van der Schaar, M.: Providing adaptive QoS to layered video over wireless local area networks through real-time retry limit adaptation. *IEEE Trans. Multimedia* **6**(2) (2004)
16. Lichtensteiger, B., Bjelajac, B., Muller, C., Wietfeld, C.: RF mesh systems for smart metering: System architecture and performance. In: *IEEE Smart Grid Communications Conference, Gaithersburg, MD, USA* (2010)
17. Pavkovic, B., Theoleyre, F., Duda, A.: Multipath Opportunistic RPL Routing over IEEE 802.15.4. In: *ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Miami Beach, FL, USA* (2011)
18. Schaefer, G., Ingelrest, F., Vetterli, M.: Potentials of opportunistic routing in energy-constrained wireless sensor networks. In: *European Conference on Wireless Sensor Networks* (2009)
19. Srinivasany, K., Jainy, M., Choi, J.I., Azim, T.: The k Factor: Inferring Protocol Performance Using Inter-link Reception Correlation. In: *In the Proceedings of The 16th Annual International Conference on Mobile Computing and Networking* (2010)
20. Winter, T., Thubert, P., et al.: RPL: IPv6 Routing Protocol for Low Power and Lossy Networks, RFC 6550. Tech. rep., IETF ROLL WG (2012)