Support for Hybrid Network in RPL

François Lemercier, Nicolas Montavont, Laurent Toutain, Kumaran Vijayasankar, Ramanuja Vedantham, Philippe Chiummiento

Support for Hybrid Network in RPL

F. Lemercier, N. Montavont, L. Toutain
IMT/Telecom Bretagne/IRISA
2 rue de la Châtaigneraie
35576 Cesson Sevigné, France
{francois.lemercier, nicolas.montavont, laurent.toutain}
@telecom-bretagne.eu

K. Vijayasankar, R. Vedantham
Texas Instruments
13532 N Central Expwy, MS 3810
Dallas TX 75243, USA
kumaran@ti.com
r-vedantham@ti.com

P. Chiummiento
Itron
CTO office
52 rue Camille Desmoulins
92130 Issy-les-Moulineaux, France
philippe.chiummiento@itron.com

***Abstract*—The Smart Grid is based on Advanced Metering Infrastructure that mostly relies on Narrow Band Power Line Communication (PLC). In such network, using a single communication interface does not fulfill the primary requirement of 99.99% reading rates and coverage. Hybrid communication, by adding an additional radio interface, is a solution to provide the quality of service required by Smart Grid applications. However, dedicated routing protocols usually operate with a single communication technology. In this paper we present three solutions to enhance the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) - a well-known routing protocol for smart grid application - for handling multi-interface devices called the Multiple RPL Instances, the Interface Oriented and the Parent Oriented. By means of simulation we show how the network can provide a higher quality of service and a better resilience to failure.**

I. Introduction

Fluctuations in the energy consumption and the integration of renewable energy production into the grid impose a modern electric network. The term smart grid describes an electric power distribution network that relies on an Advanced Metering Infrastructure (AMI) that can handle the increase of multiple and fluctuating production sites. The mutation of the electric production field modifies the equilibrium of the electric power distribution network and its exploitation has to be adapted to move from a centralized to a distributed power generation. Automatic Meter Reading (AMR) is a first step for the smart grid and it is widely used by suppliers to collect data from electric and gas meters. AMI is a smart grid component that allows providing a two-way communication between providers and consumers, for example, to inform customers on electricity prices or perform utility management. AMI networks mostly relies on a multitude of devices using multiple heterogeneous technologies, such as Radio Frequency (RF) or Power Line Communication (PLC). Those technologies are considered as Low power and Lossy Networks because of their sensibility to interference and the constrained characteristics of the devices used, especially in term of energy and computing resources.

In those networks, two kinds of routing protocols exist, traditionally named proactive and reactive protocols. Reactive routing protocols builds routes on demand and maintains them only if needed. Routes are deleted when there is no traffic on them. However, as a route does not exist before its use, an additional latency has to be taken into account for the construction. AODV [1] is a well-known example of such a reactive protocol. In proactive routing protocols, every nodes in the network must know a route to all destination at any time, and the routes are computed before they are needed. In this way, a node can transmit data to a destination without additional delay, just by looking in its routing table. To be up-to-date, the routing table is maintained by periodic messages. The transmission frequency of those signaling messages has to be high enough to take into account the network topology changes, but in the same time it should not add too much overhead in the network.

RPL [2] is a proactive routing protocol based on distance vector and it is the most common protocol used in the Internet Of Things (IoT) community. It operates at the network layer, and thus is link-layer agnostic, i.e. it can operate on RF as well as on PLC networks. RPL allows discovering neighbors in a given topology, and carefully selectsome of these neighbors to build a tree rooted at a sink. The selection is based on an objective function, which allows ranking nodes according to some metrics. Most of today RPL usage is however in an homogeneous network, where nodes are all equipped with a single network interface. However, a substantial number of limitations come with homogeneous network, such as deployment issues, network evolution or coverage aspects. Using multiple and heterogeneous interfaces could unlock the real possibilities in smart grids by allowing more applications and communications between devices. For example, it could be used to struggle against interference by using technology diversity. Or it can allow for more flexible deployment depending on the node density. In this paper we investigate how multiple interfaces can be managed within RPL, and what options can be considered to take advantage of this heterogeneity. We propose three alternatives, called Multiple RPL Instances (MI), Parent Oriented (PO) and Interface Oriented (IO) solutions.

The MI solution is based on several RPL instances, one for each technology. A RPL instance is defined in the RPL standard and allows isolating different logical graphs from a given topology. Using this feature, it is straightforward to manipulate one instance per technology, which also allows defining a different objective function for each technology. The PO and IO solutions take a different approach by merging the characteristics of the heterogeneous interfaces. The PO solution combines multiple links into a single virtual link while the IO solution sees each interface from a neighbor node as a potential parent in the RPL tree. An additional and independent interface management policy is also introduced to choose which technology to use when forwarding a data packet toward the root. This policy can dictate the way the interfaces can be used, for example, by imposing a given technology for a given data flow, or on the contrary by letting intermediate forwarder to choose their best interface. To evaluate these approaches, we developed a RPL DODAG Simulator that computes a DODAG from a predefined or random generated graph, according to the three solutions described in this paper.

Our contributions can be summarized as follows:

1) We extend RPL to support multiple hybrid interfaces by proposing three solutions (MI, PO and IO)
2) We propose a new metric to symbolize the ability of a node to communicate over two interfaces.
3) We introduce a configurable RPL DODAG Simulator (RDSim) that computes a DODAG from random or predefined graphs, for the three solutions presented in this paper.
4) Using RDSim, we observe that dealing with multiple interfaces give better performance than using a single interface. We also show that the PO solution appears to be more stable than the other solutions.

The remainder of the paper is organized as follows. We describe the routing protocol RPL and related works in Section II, before presenting our interface management methods in Section III. We present our evaluation platform and results in Section IV, before concluding in Section V.

II. Related works

In RPL [2], the topology is organized in a Directed Acyclic Graphs (DAG), where the connections between nodes have a direction and a "non-circular" property. Due to the acyclic nature of a DAG, the graph comprises at least one node with no outgoing edge, called the root. To construct topologies, RPL uses a typical kind of DAG: the destination oriented DAG (DODAG), which is a DAG with only one DAG root. Figure 1d shows a DODAG made of five nodes with one root.

The position in the DODAG is given by the rank of a node. All nodes start with an infinite rank, except for the root. When the graph construction begins, the root sends a multicast DIO message to its neighbors. After receiving this DIO message, and because the DIO message comes from a node with a lower rank, nodes in range of the root will process the message and select the root as its DODAG parent. Then the nodes will compute their rank according to the objective function (OF) and, in turn, send a multicast DIO message. In [3], a set of routing metrics are proposed and the OF defines how to convert those metrics into a rank value. Note that during the lifetime of the network, a node will receive many DIO messages, including from its sons. DIO messages that come from nodes with higher rank are ignored.

A RPLInstanceID identifies at least one DODAG, and several RPLInstanceID are possible in a network. As multiple DODAGs can belong to a RPLInstanceID, RPL uses a DODAGID to identify a particular DODAG in a RPLInstanceID. Finally, RPL uses a DODAGVersionNumber to define the version of a DODAG, this value is incremented by the root to validate the integrity of the DODAG, a router with an old DODAGVersionNumber will not be chosen as a parent. A node can belong to multiple RPL instances in a network. Over a physical topology, it is possible to construct multiple DODAGs and a RPL node may join more than one instance. However, dealing with two RPL instances needs to take into account the possibility of loops when passing from one instance to the other. For example, one node could be parent of a node in an instance, and child of the same node in another instance at the same time.

To the best of our knowledge, only few works have been done to integrate hybrid network in RPL. These works especially use the multiple RPL instance feature. Indeed Long et al. [4] already proposed an interesting multi-instance usage of RPL in a homogeneous network. They studied how to prioritize traffic in a wireless sensor network using cross-layer mechanism based on CSMA and RPL. They defined two instances, one for priority packets and another one for the other packets. Their Cooja simulation showed that the end-to-end delivery latency is decreased. In the same spirit, but in a hybrid network this time, Pignolet et al. [5] propose an extension of the contiki network stack to handle multiple interfaces. One instance per technology is created for the routing operation but the interface management is not addressed. A smart grid simulation scenario is also studied in Cooja simulator to test the repair mechanism benefits of two interfaces nodes. Balmau et al. [6] extend this work by evaluating how much a PLC network can be degraded before having an effect on the smart grid application performance.

Chauvenet et al. [7] propose a cooperation between PLC and RF networks that shows the suitability of RPL for such networks but highlight the importance of an adapted OF to suit PLC networks. Also, as the cooperation of the PLC and RF networks is based on single interface routers and gateways, the multiple interfaces case is not addressed. H. Sawada et al. [8] present a communication protocol to handle the construction of a network infrastructure by following a DODAG routing for multi-interface communication. This study shows that DODAG routing performs well with multi-interfaces architecture (RF and PLC), especially when the DODAG is altered by jamming links.

These works show usage examples of the multiple RPL instance feature, as well as some preliminary multiple interfaces management. In the next section, we go a step further, and introduce more integrated hybrid network management, especially with the PO and IO solutions.

III. How to manage multiple interface in RPL

In this section, we present three design alternatives for RPL to manage multi-interfaces nodes. We assume that nodes have two different interfaces, for example one 802.15.4 RF interface and one 1901.2 PLC interface.

A. Multiple RPL instances

Because nodes have two interfaces, we could easily think of two RPL instances: one instance for each interface. Each instance could be governed by its own OF so we could consider different ways to create the route to the sink. Each instance optimizes the path according to the requirements of its interface. Long et al. [4] already take benefit from the multiple RPL instances by defining two kinds of nodes (regular nodes and alarm nodes) in order to support priority traffic in the network. We could also assume that another RPL instance could be used when one instance fails. For example, the RF DAG could be a backup in case of an unreliability issue in the PLC DAG. Because loop-free graph can not be guaranteed if packets are switching from one instance to another several times, we only authorize a single transition from one instance to another. So we keep two distinct instances, and a packet stays in its DODAG instance as long as there is no failure. Upon failure, the packet can switch instance, only once.

To enable this mode of operation, one has to decide which is the primary technology, and which is the backup one. The primary technology should use a smaller RPLInstanceID because RPL only authorizes a packet to move to a higher RPLInstanceID. For example, let us assume that the PLC interface is chosen as the default communication technology and the RF is only used as a backup interface. At the time of the initial DODAG configuration, the PLC interface of each node joins a specific DODAG for PLC in a specific RPLInstance (e.g. RPLInstanceID 1). The same goes with the RF interface (e.g. the RF interface of a node joins the DODAG RPLInstanceID 2). These DODAGs are rooted to one node, called the sink, via its two interfaces. Because the main advantage of a multiple RPL instances architecture is to have a specific OF for each instance, choosing the appropriate OF according to PLC or RF networks is a key point.

Figure 1a depicts a physical topology of five hybrid nodes, randomly generated by RDSim. Solid links and dashed links represents respectively PLC links and RF links. Nodes are vertices in the graph and an edge between two vertices represent a link between two nodes. Each edge is associated with an integer between 1 and 8, which represents the link quality (this can be viewed as the Expected Transmission Count for example).

This first proposal is not intrusive in the RPL code because it exclusively relies on existing RPL mechanisms. Only the interface policy needs to be coded in order to set up the behavior described above. Note that the RF interface can also be used from a data source node, but in this case, no failure recovery can be provided from the PLC RPL instance, since it is forbidden to switch to a smaller RPLInstanceID.

B. Parent oriented design

The goal of the Parent Oriented solution (PO) is somehow to merge the communication facilities of a node into a single metric in order to compare all interfaces of a node at the same time. Thus a node chooses its best parent according to the cumulated link qualities of both interfaces. Note that in this case, once a node is chosen as parent, any of the two interfaces can be used. Contrary to the MI solution, the purpose of the PO solution is to build a single RPL instance that takes into account the multi-interfaces feature of the nodes. Because the goal is to always choose the best parent according to its ability to communicate on two technologies, during the parent selection phase, nodes must send a DODAG Information object (DIO) on both radio and PLC interface. While being sent on different media, these messages contain the same (and unique) RPL rank of the sender. By using a link evaluation metric and the rank given in the DIO message, a node is able to choose its preferred parent among the parent set. However, if a neighbor is only seen through a single technology (either because it is not a hybrid node, or because it is out of range), it will be "penalized" because it cannot offer the benefit of a heterogeneous communication.

Concerning the rank computation, a new metric is introduced to symbolize the ability of a node to communicate on two interfaces. For a given node having its two interfaces working with a parent, the calculation of the symbolized communication is basically the average value of the two metrics, and could be formulated with:

$$W(e) = \frac{W(e1) + W(e2)}{2} \quad (1)$$

Where W(e) is the calculated weight of edge e, W(e1) W(e2) are the weights of the two communication links. If only one link exists between two given nodes, the link is penalized and the value associated to the link is given by the formula:

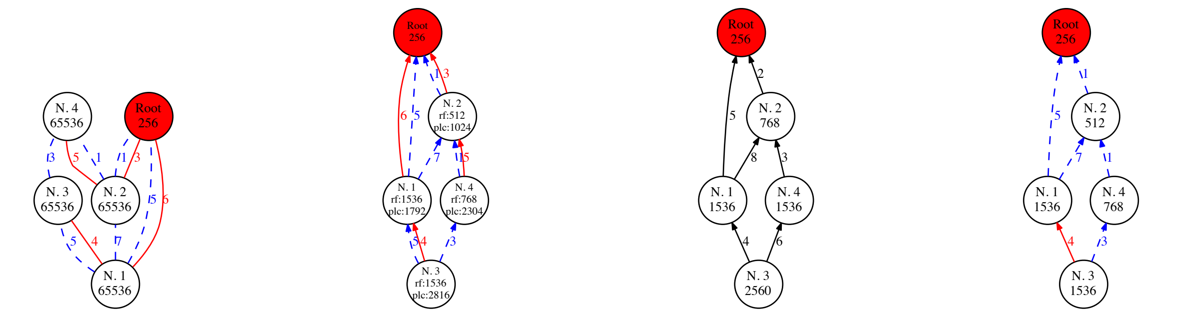
$$W(e) = \frac{W(e1) + 9}{2} \quad (2)$$

Figure 1c depicts the DAG constructed by RDSim with the given physical topology shown in figure 1a. Each couple of links between two nodes are symbolized as a single link with a specific value (obtained by the formula given in eq. 1).

C. Interface oriented design

The Interface Oriented solution (IO) considers each link with a neighbor in a given technology as an independent potential parent. So instead of a having a list of neighbors, we consider tuples (nodeID, interfaceID). By doing this way, we will always choose the best link as the preferred parent. Because each node chooses a given technology toward the root, a packet may change technology at each hop. In this solution, only one RPL instance is used and a single RPL DODAG is created so a node has only one rank. During the parent selection phase, a node evaluates each link. If only one interface is available on a potential parent, contrary to the PO, this parent is not penalized.

After sending DIO on both interfaces of the root, every neighbors in the coverage area of the root will choose as a parent the interface which has the best metric and compute a



(a) Physical topology

(b) Multiple RPL Instances (MI) DAG

(c) Parent Oriented (PO) DAG

(d) Interface Oriented (IO) DAG

Fig. 1. Graphs generated by RDSim

Solution	MI	PO	IO
Load balancing	-	++	+
Stability	+	++	-
Implementation	++	-	-

TABLE I
SOLUTION COMPARISON

rank. Neighbors of the root will send a DIO message on both interfaces, and nodes in the coverage area will receive it and also choose the best interface. Their other interface will not be in the parent set of those neighbors. For the hybrid physical topology given in figure 1a, each node choosing a parent will select the interface of a potential parent with the best metric. As a result, figure 1d depicts the resulting DAG constructed by RDSim.

D. *Summary*

MI, PO and IO present different ways of managing heterogeneous interfaces in RPL. They particularly define how to build a DODAG, and how to combine hybrid interfaces into one or several trees. Then, an interface management policy can further be considered to use these interfaces. In MI, there is not much choice: sources nodes (nodes which injects data packets destined to the root) can choose between the two RPL instances, given that if the technology with the higher RPLInstanceID is chosen, no backup will be possible. Forwarding nodes on the path will generally forward these data packets using the same RPLInstanceID (i.e. technology). In IO, there is no choice at all, because each node will likely choose their preferred parent, which also specifies which technology to use with this node. In the opposite, PO allows for more flexibility. Any technology can be chosen at any time.

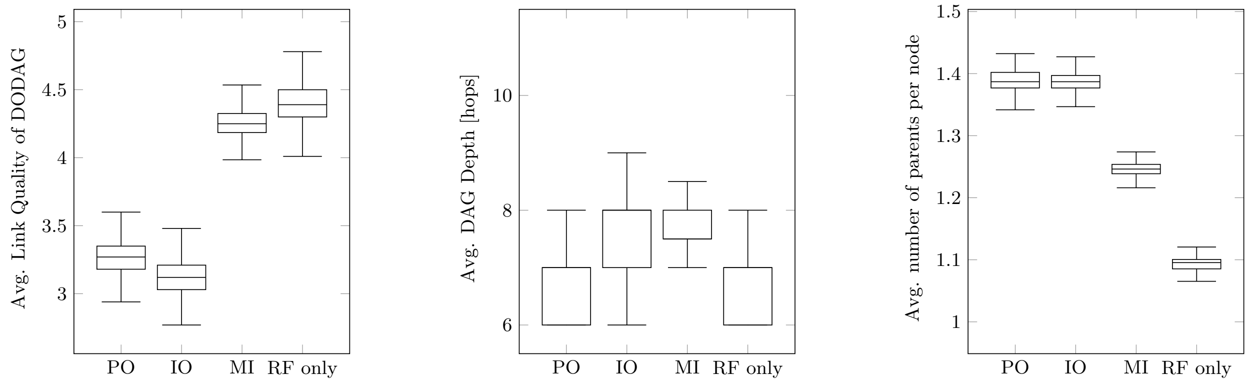
As shown in Table I, each of the three presented solutions have different characteristics. PO and IO solutions require more modifications to RPL than the MI solution because of the need to build the DODAG according to the two interfaces available on the nodes. On the other hand, the MI solution let RPL only see one interface per RPL instance. Concerning the stability of the DAG, the PO solution should be more stable than the MI and IO solutions, because the OF chooses a preferred parent based on the two interfaces. In the PO solution, the failure of one link will not trigger a parent change if there is another technology available between the nodes. For the MI solution, because each instance has its own OF, the stability will depend on the usual RPL parameters. Regarding the load balancing possibility, a drawback of the MI solution is the risk of holes in the physical topology that can cause an instance not to be used.

IV. PERFORMANCE EVALUATION

A. *Evaluation platform and methodology*

To compare the three presented solutions, we developed a RPL DODAG Simulator called RDSim. This command line tool helps user to create a node topology and gives statistics over the computed DODAGs for each solution. As an option, the program can also produce a graphical representation of each solution.

RDSim allows to generate a random topology with a specific number of nodes and a specific proportion of hybrid/single interface node. It can also load a predefined topology. Nodes in the topology are the vertices of the graph and all edges have a random computed weight (i.e which could represents an ETX metric between 1 and 8, where 8 is the worst value). All nodes except the root have a rank attribute set to 65535 during the creation of the graph. The rank attribute of the root is fixed to 256 for both technologies, because in the MI solution, a node has two ranks. The resulting graph, considered as the physical representation of the hybrid network is analyzed iteratively from the root to the next hops to construct the DAG. During this processing, node ranks are calculated. It has to be noted that RDSim does not handle message exchanges yet, its purpose is graph generation and performance analysis on these generated topologies / DODAGs. For the MI, PO and IO solutions, the computation of the rank is given by the formula below, based on the rank calculation given in [9]:



(a) Link Quality

(b) DAG depth

(c) Number of parents per node

Fig. 2. Metrics for 200 nodes topology, PO, IO and MI with 100% of nodes hybrid + RF Only

Number of nodes	100, 200			
Proposal	MI, PO, IO			RF only
Hybrid nodes	100%	70%	50%	0%
RF Nodes	0%	15%	25%	100%
PLC Nodes	0%	15%	25%	0%

TABLE II
SIMULATION PARAMETERS

$$R(N) = R(P) + Link\_Quality * MinHopRankIncrease \quad (3)$$

Where R(N) is the calculated rank. R(P) is the rank of the parent. Link_Quality is the edge weight, a value between 1 and 8, between parent P and node N. MinHopRankIncrease is fixed at 256 as recommended in [2].

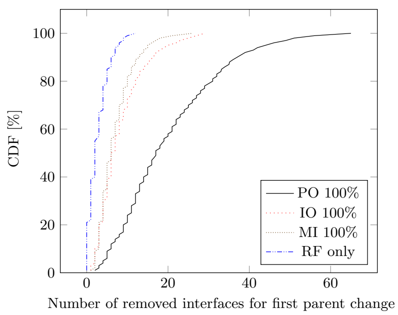
The two DODAGS generated by the MI solution are just superposed and are a good way to highlight the potential issues of loops or hole problems in this kind of architecture. When the PO solution is applied on the hybrid graph, all edges are analyzed and each couple of edges between two nodes are symbolized in single ”average” edge. The weight value of the single edge is calculated with the formula given in eq. 1. It has to be noted that if only one edge exists between two nodes, a penalization is introduced to the single ”average” edge with the formula given in eq. 2. The result is a DAG with a single edge between nodes. Regarding the IO solution, all edges are analyzed and each couple of edges between two nodes are compared. The edge with the best metric is kept in the graph, the others are deleted. If only one edge exists between two nodes, it is kept. The result is a DAG with single edges between nodes.

B. Results

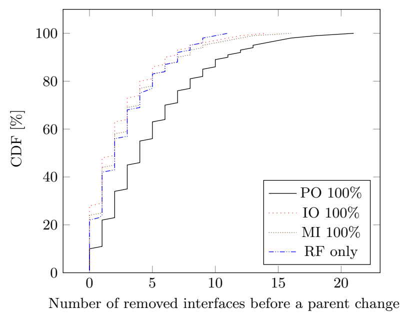
We used RDSim to measure the potential gain obtained when nodes have multiple interfaces and to compare the three management proposals. We varied the size of the network (100 and 200 nodes) and the hybrid node proportion as indicated in table II. Each simulation set was performed 1000 times on random topologies. For space reasons, we only present the 200 node topology results for full hybrid and RF only scenario because results for a different proportion of hybrid nodes give the same tendencies.

For each scenario we measured the average link quality, the maximum number of hops and the average number of parents per node. The average link quality helps to identify which solution builds the route with the best link quality. As the number of hops is critical in smart grid networks, the maximum number of hops in the DODAG helps us to show which solution gives the minimum depth of the DODAG. The average number of parent per node shows the overall possibility of nodes to change parent if necessary. Then, as we expect a hybrid network to be more resilient to failure, we removed one technology from a node (chosen randomly) and measured how the DODAG evolved in all tested proposal. In particular, we counted the number of interfaces that can be removed and the number of parent changes before 10% of the nodes became orphans.

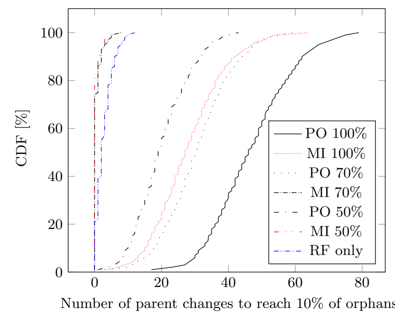
Fig. 2 shows the metrics of the DODAG generated by each proposal. We can observe that PO and IO show equivalent results and are generally better than MI and RF Only, especially in the link quality metric and the size of the parent set. This is because PO and IO take full advantage of the hybrid nature of the nodes, and are able to switch from one technology to another. This makes the average link quality to the parent better, and the size of the parent set larger. Because no jump from one instance to another is allowed, MI has an average link quality worst than PO and IO, and still slightly better than a network where nodes have a single interface. This is because hybrid nodes can select the best tree among the RF and PLC ones. The depth of the tree is quite equivalent in each solution. Fig 3 shows the results when we consider failure in the network. Fig 3a shows the CDF of the number of removed interfaces from the initial topology before the first parent change in the DODAG. The first parent change can occur after several interface removals because in a hybrid network, nodes may switch interface upon one technology failure. IO and MI shows close results, and most of the times there is a parent change before 16 interfaces were removed.



(a) Number of removed interfaces before the first parent change



(b) Number of removed interfaces between two consecutive parent changes



(c) Number of parent changes when the topology has 10% orphans

Fig. 3. CDF for 200 nodes topology

In RF only, if the removed link was the one with the parent node, the node must change its parent, so we see that there are very few removed interfaces in the topology before observing an effect on the DODAG. Fig 3b shows the same metric but not only before the first parent change in the topology, but in-between all parent changes that occur when we consecutively remove interfaces (until reaching 10% of orphans nodes).

The same observations can be made, but with a smaller number of interfaces, because the DODAG on which a failure occurs is less dense. Fig 3c shows the cumulated number of parent changes once we removed so many interfaces that there are 10% of the nodes that become orphans (no link to the DODAG). For a better readability, we omitted to plot IO because it was very close to PO. The first observation is that the more the network is hybrid, the more interfaces can be removed before 10% of nodes are orphans, meaning that the network is more resilient to failure. This observation stands for MI, but in a limited manner.

V. Conclusion

In this paper, we proposed three alternatives to manage hybrid network in RPL for smart grids. Using Multiple RPL Instances (MI) appear to be a straight forward solution to manage heterogeneous interfaces, because it comes as a standard feature in RPL. However, in order to avoid loops, nodes can bearly switch from one instance (technology) to another. This limitation greatly penalizes the gain we could expect from having multiple interfaces. Interface Oriented (IO) offers a simple and easy management of multiple interfaces by considering each tuple (node, interface) as an independent node. Thus IO will only takes the best links to build the DODAG, but at the cost of an unstable DAG. Upon link failure, IO reacts quickly and will immediately change the selected parent. Parent Oriented (PO) proposes a novel metric to measure the quality of a neighbor, taking into account its heterogeneity degree. This solution seems to be more promising in term of reliability.

We performed several simulations using an ad hoc simulator called RDSim, which generate a random topology and calculate the DODAG for each solution. We tested different parameters set (size of the network, proportion of hybrid nodes) and we showed that whatever the way multiple interfaces are managed, there is a gain in performance (better link quality, better resilience to failure, more stability). IO and PO show the best results, especially PO that seems to take full advantage of the heterogeneity in case of failure. When one interface fails, a node will first try the second one before changing the DODAG. MI proposes a smart management considering its integration in RPL (by using default RPL features) but suffer from its mechanisms for loop avoidance. In our future work, we will study what rules can be defined to allow the transition between instances without creating a loop to see if better performance can be obtained.

References

[1] Vibha Kathuria, Gokulraj Mohanasundaram, and Samir R Das. A Simulation Study of Routing Protocols for Smart Meter Networks. pages 384–389, 2013.

[2] T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, JP. Vasseur, and R. Alexander. RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550 (Proposed Standard), March 2012.

[3] JP. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel. Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks. RFC 6551 (Proposed Standard), March 2012.

[4] Nguyen Thanh Long, Marie-Paule Uwase, Jacques Tiberghien, and Kris Steenhaut. QoS-aware cross-layer mechanism for multiple instances RPL. *2013 International Conference on Advanced Technologies for Communications (ATC 2013)*, pages 44–49, 2013.

[5] Yvonne-Anne Pignolet; Ilias Rinis; Dacfey Dzung; Abdulkadir Karaagac. Heterogeneous Multi-interface Routing: Networking Stack and Simulator Extensions. *2012 IEEE 9th International Conference on Mobile Adhoc and Sensor Systems (MASS)*, Supplement:1–6, 2012.

[6] Oana Balmau, Dacfey Dzung, Abdulkadir Karaagac, Vukasin Nesovic, Aleksandar Paunovic, Yvonne Anne Pignolet, and Niloufar Alipour Tehrani. Evaluation of RPL for medium voltage power line communication. *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 446–451, 2014.

[7] C. Chauvenet, B. Tourancheau, D. Genon-Catalot, P.-E. Goudet, and M. Pouillot. A Communication Stack over PLC for Multi Physical Layer IPv6 Networking. *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pages 0–5, 2010.

[8] Hisashi Sawada, Hiroshi Kuriyamat, Naoki Yusat, Tadanori Mizuno, and Hiroshi Mineno. Mutually Complementary Communication Protocol based on Destination Oriented Directed Acyclic Graph. pages 275–279, 2012.

[9] P. Thubert. Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL). RFC 6552 (Proposed Standard), March 2012.