



HAL
open science

Exploiter plusieurs parents avec RPL pour améliorer la stabilité

Oana Iova, Fabrice Theoleyre, Thomas Noel

► **To cite this version:**

Oana Iova, Fabrice Theoleyre, Thomas Noel. Exploiter plusieurs parents avec RPL pour améliorer la stabilité. ALGOTEL 2014 – 16èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, Jun 2014, Le Bois-Plage-en-Ré, France. pp.1-4. hal-00985047v1

HAL Id: hal-00985047

<https://hal.science/hal-00985047v1>

Submitted on 29 Apr 2014 (v1), last revised 4 Jul 2014 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Efficient Energy-Balancing in Multipath RPL

Oana Iova¹, Fabrice Theoleyre¹ and Thomas Noel¹

¹ICube, CNRS, Université de Strasbourg, Email: {otiova, theoleyre, noel}@unistra.fr

RPL has emerged as the *de facto* routing standard in low-power and lossy networks. While most of the proposals focus on minimizing the global energy consumption, we aim here at designing an energy-balancing routing protocol: each node should consume the same quantity of energy to improve the network lifetime. We propose the *Expected Lifetime* routing metric, denoting the time until the node will run out of energy. We present mechanisms to detect energy-bottleneck nodes and to spread the traffic load uniformly among them, using the DAG structure of RPL. Simulations highlight we improve both the routing reliability and the network lifetime.

Keywords: RPL, multipath, energy efficiency, energy-balancing, WSN, network lifetime

1 Introduction

Routing in Wireless Sensor Networks has been extensively studied in the last decade. RPL has emerged as the *de facto* standard for the Internet of Things [W⁺12] and reflects the current evolution of this research area: introducing redundancy in the routing structure to be fault-tolerant. However, in our opinion, the current version of RPL presents two ways of improvement.

First, the Roll working group has focused on providing efficient routing mechanisms. We have now to provide metrics and mechanisms to make RPL energy-efficient: the topology (i.e., the Destination Oriented Directed Acyclic Graph - DODAG) should be constructed based on energy criteria. Chang *et al.* combined linearly the residual energy and the ETX [C⁺13], but the weight is not related to the real lifetime. Second, a node selects **one** preferred parent to construct the DODAG without loops. Still, only this preferred parent is used for routing: the other ones have only a *backup* purpose. Hong *et al.* proposed to choose the preferred parent using the hop count and then select as the forwarding node the parent offering the best link quality [H⁺11]. However, a collection of nodes may still forward most of the traffic, depleting their energy.

We propose here a new routing metric and we exploit the diversity of the topology constructed by RPL. We improve the network lifetime with energy-balanced paths and an accurate load-balancing scheme.

2 Routing Metric: Expected Lifetime (ELT)

In order to construct an energy balanced topology, we propose a new routing metric: the Expected Lifetime (ELT). ELT estimates the expected lifetime, i.e., the time before a node dies if it keeps on forwarding the same quantity of traffic. The metric only accounts energy drained by transmissions. Reception energy consumption is assumed to be included in the transmission energy, since all the packets received will be further transmitted until they reach the sink. To compute its ELT, a node needs (cf. Table 1):

- the volume of traffic it has to forward ($T_{total}(N)$);
- the ratio of traffic sent to each parent P (α_P), where $\sum_{P \in Parent(N)} \alpha_P = 1$;
- the average number of retransmissions to each parent ($ETX(N, P)$);
- its residual energy ($E_{res}(N)$) and the energy drained per transmitted bit ($P_{Tx}/DATA_RATE$).

Finally, a node N estimates its ELT as following:

$$ELT(N) = \frac{E_{res}(N)}{\sum_{P \in Parents(N)} \left(\frac{\alpha_P \times T_{total}(N) \times ETX(N, P)}{DATA_RATE} \right) \times P_{Tx}(N)} \quad (1)$$

$ELT(X)$	Expected lifetime of X	$E_{res}(X)$	Residual energy of X (in Joule)
α_P	Ratio of traffic sent to parent P	$T_{total}(X)$	Throughput (bits/s) of X
$ETX(A,B)$	ETX of the link $A \rightarrow B$	Parents(X)	Parents set of node X
$r_{X,B}$	Ratio of traffic forwarded by X to bottleneck B	$P_{Tx}(X)$	Radio power in transmission mode (in Watt or Joule/s)
DATA_RATE	The rate at which the data is sent (bits/s)	Bottlenecks(X)	Bottlenecks set of node X

Tab. 1: Notations used in the article

Let us consider now that a node N that has to associate with the DODAG. Since the bottleneck is most likely to be the first node to die, the new node has to estimate the impact of its own traffic on the bottleneck's lifetime. Besides the information needed by a node to compute its own ELT, we have to take into account that a node sends its traffic to several parents. Hence, only a part of its total traffic will arrive at a specific bottleneck. Let $r_{N,B}$ be the ratio of traffic that N forwards to the bottleneck B . Then: $r_{N,B} = \sum_{P \in \text{Parent}(N)} (\alpha_P \times r_{P,B})$. Using equation 1, a node N can now estimate the ELT of a bottleneck B , by adding the ratio of its own traffic:

$$ELT(B) = \frac{E_{res}(B)}{\sum_{P \in \text{Parents}(B)} \left(\frac{\alpha_P \times (r_{N,B} \times T_{total}(N) + T_{total}(B)) \times ETX(B,P)}{DATA_RATE} \right) \times P_{Tx}(B)} \quad (2)$$

The information about the bottleneck (e.g. existing traffic forwarded by the bottleneck, the residual energy of the bottleneck, etc.) that a node needs in order to estimate its ELT like in Eq. 2 is sent along the paths in a compact mode in the DIOs. The ELT of a node is updated every time a DIO is received. Thus, each node maintains up to date information.

3 Multipath Construction

In order to construct the routing topology, a node needs to choose its next hop (i.e., the preferred parent in RPL) to route the packets. When choosing its preferred parent, a node must consider both its own lifetime and the lifetime of the bottlenecks, in order to estimate which of them becomes the new bottleneck. However, it is not possible to know the ratio of traffic that will be sent to each of the parents before actually choosing the set of parents. Hence, we cannot accurately estimate neither the lifetime of the node, nor the lifetime of the bottlenecks.

We propose that during the preferred parent selection we assume a node will send all its traffic to one single parent. Even if we underestimate the lifetime of the bottlenecks, we are sure to choose as the preferred parent the node maximizing the lifetime of the bottlenecks. In other words, we prefer considering the worst case to balance more efficiently the energy consumption.

We propose the algorithm 1 to select the preferred parent. For each possible parent a node N will:

1. compute the ELT of all the bottlenecks advertised by a parent P , as if it will send all its traffic to that parent and save the minimum value among all (line 4);
2. compute its own lifetime when choosing this parent and verify if the node did not become the new bottleneck (line 5);
3. remove the traffic to this parent to test the other ones: we have to test all the parents before taking a decision (line 10);
4. choose as preferred parent the node that maximizes the lifetime of the bottleneck with the minimum ELT, itself included (lines 6, 7, 8).

Algorithm 1: Preferred parent selection

```

Data:  $N$ 
Result: preferred_parent of  $N$ 
1  $max\_elt \leftarrow 0$ ;
2 for  $P \in Parents(N)$  do
   // all the traffic is sent to P
3    $\alpha_P \leftarrow 1$ ;

   // track the minimum ELT (all
   // bottlenecks & myself)
4    $min\_elt \leftarrow \min_{B \in Bottlenecks(P)} \{ELT(B)\}$ ;
5    $min\_elt \leftarrow \min\{min\_elt, ELT(N)\}$ ;

   // is this parent the best one?
6   if  $max\_elt < min\_elt$  then
7      $max\_elt \leftarrow min\_elt$ ;
8     preferred_parent  $\leftarrow P$ ;
9   end

   // test now the other parents
10   $\alpha_P \leftarrow 0$ ;
11 end
12 return preferred_parent;

```

Algorithm 2: Load balancing

```

Data:  $N, load\_step$ 
Result: compute  $\{\alpha_P\}_{P \in Parents(N)}$  — the ratio of traffic
   to send to each parent;
1 for  $i = 1$  to  $load\_step^{-1}$  do
2    $max\_elt \leftarrow 0$ ;  $\alpha_P \leftarrow 0$ ;
3   for  $P \in Parents(N)$  do
4      $\alpha_P \leftarrow \alpha_P + load\_step$ ;

     // track the min ELT with this
     // new weight
5      $min\_elt \leftarrow \min_{B \in Bottlenecks(P)} \{ELT(B)\}$ ;
6      $min\_elt \leftarrow \min\{min\_elt, ELT(N)\}$ ;

7     if  $max\_elt < min\_elt$  then
8        $max\_elt \leftarrow min\_elt$ ;
9       parent_max  $\leftarrow P$ ;
10    end

    // test each parent before
    // taking a decision
11     $\alpha_P \leftarrow \alpha_P - load\_step$ ;
12  end
13   $\alpha_{parent\_max} \leftarrow \alpha_{parent\_max} + load\_step$ ;
14 end

```

After choosing its preferred parent a node has to compute its relative distance from the border router, i.e., its *rank*. In order to avoid the formation of loops, the rank of the nodes in the DODAG must strictly monotonically increase from the border router towards the leaves. Since the *Expected Lifetime* represents a minimum metric along a path, its value cannot be used to compute the rank: all the nodes in the sub-DODAG would have the same value. We propose that a node computes its rank by adding a constant step value to the rank of its preferred parent: $Rank(N) = Rank(P_N) + Step \times \text{MinHopRankIncrease}$, where *Step* is a scalar value and *MinHopRankIncrease* the RPL parameter [W⁺12]. A node may have consequently more alternative parents, while avoiding the formation of loops, as proven in [I⁺14].

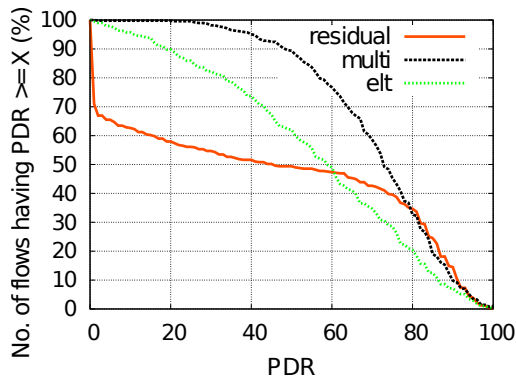
4 Energy Balancing by Exploiting Multiple Paths

A node must split its traffic among all the available paths, taking into account the lifetime of each bottleneck. We present here a greedy algorithm, that leads to an *acceptable* solution in a reasonable time. A node N has to distribute the load to each parent so that it balances the expected lifetime of the corresponding bottlenecks. A node divides its traffic in $\frac{1}{load_step}$ equal fractions, and assigns sequentially each fraction to the parent which maximizes the minimum lifetime among all its bottlenecks.

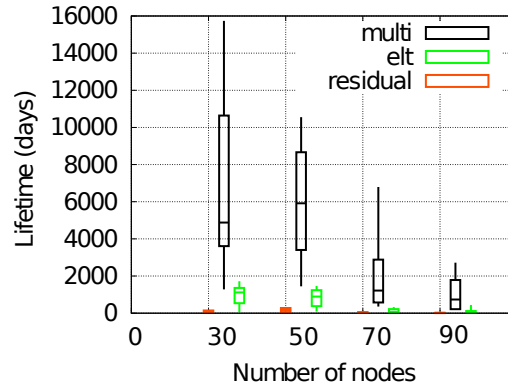
Algorithm 2 defines more formally the heuristic:

1. N considers iteratively each parent (line 1);
2. it computes the minimum ELT that would be obtained by increasing the weight of this parent by *load_step* (line 4). It considers the lifetime of each bottleneck (line 5) and itself (line 6);
3. If this minimum value maximizes the network lifetime, it saves the current parent as the best one (line 7-10);
4. N has to test the other parents before definitively setting the new weight (line 11);
5. Finally, N assigns *load_step* to the best parent (line 13).

A small *load_step* balances more finely the energy in the network but increases the computation complexity, since a node has to execute the assignment $load_step^{-1}$ times. Some optimizations are also possible in the implementation. In particular, for $i > 1$, a node has to recompute the minimum ELT (lines 4-11) only for the parent which was the best one at the previous iteration ($i - 1$). Indeed, the possible weight of all the other parents has already been considered in the previous step.



(a) Complementary CDF of the end-to-end PDR



(b) Network Lifetime in function of the density

This greedy assignment is a $1 + load_step$ -approximation. We may prove by contradiction that the assignment cannot distribute the last $1 + load_step$ fraction of the traffic to a non-optimal parent.

5 Simulation Results

We simulated RPL using WSNet (<http://wsnet.gforge.inria.fr/>) with 50 nodes on a 300m x 300m area. The results are averaged over 10 simulations with different random topologies. For the traffic, we considered usual CBR convergecast flows with 1 packet/min. We compared our multipath proposal against the standard RPL, where the DODAG is constructed using both the residual energy and ELT as routing metrics.

We first evaluated the reliability as the ratio of packets received by the sink. Fig. 1a illustrates the complementary cumulative distribution function (CCDF) of the Packet Delivery Ratio (PDR) for all the flows. In the standard RPL, ELT obtains the best PDR. The residual energy has the worst reliability, since it tends to privilege nodes with energy, without taking into account the link quality. We can also notice how the multipath protocol takes advantage of the load balancing to offer the best reliability.

Second, we evaluated the network lifetime (i.e, the time before the first node dies) in function of the density. We increased the number of nodes within the same simulation area. We can observe in Fig. 1b that our proposal clearly outperforms the standard RPL, even when ELT is used as the routing metric. Multipath routing helps balancing more accurately the energy: routing decisions are not binary, and the traffic is spread to all the bottlenecks. The weights accurately *smooth* the traffic redirections.

6 Conclusion and Perspectives

We designed a new routing metric to prolong the network lifetime: the *Expected Lifetime*. This energy-balanced RPL exploits all the parents and balances well the energy consumption in the whole network.

We are currently investigating how RPL should integrate inaccuracies in the metric estimation. Indeed, the radio link quality is stochastic, and the routes constructed by RPL should not change if the radio link quality has not *significantly* changed. We plan also to experimentally evaluate this new multipath energy-balancing version of RPL, to verify it operates efficiently *in vivo*.

References

- [C⁺13] L.-H. Chang et al. Energy-efficient oriented routing algorithm in wireless sensor networks. In *SMC*. IEEE, 2013.
- [H⁺11] K.-S. Hong et al. DAG-based multipath routing for mobile sensor networks. In *ICTC*, 2011.
- [I⁺14] O. Iova et al. Improving Network Lifetime with Energy-Balancing Routing: Application to RPL. <http://clarinet.u-strasbg.fr/~otiova/papers/elt.pdf>, 2014.
- [W⁺12] T. Winter et al. RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks. RFC 6550, Internet Engineering Task Force, 2012.