



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-00985047v2

HAL Id: hal-00985047

<https://hal.science/hal-00985047v2>

Submitted on 4 Jul 2014

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.

Exploiter plusieurs parents avec RPL pour améliorer la stabilité

Oana Iova¹, Fabrice Theoleyre¹ et Thomas Noel¹

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

RPL est considéré comme le standard *de facto* pour le routage dans les réseaux sans-fil basse consommation avec pertes. Tandis que beaucoup de propositions se focalisent sur la minimisation de la consommation globale d'énergie, nous proposons ici de concevoir un protocole de routage équilibrant la charge et donc la consommation énergétique. Nous proposons la métrique de routage *Expected Lifetime*, reflétant le temps restant pour un noeud avant qu'il épuise son énergie. Nous présentons également des mécanismes afin de détecter les goulots d'étranglement pour répartir la charge uniformément entre eux. Les simulations que nous présentons valide la pertinence d'une telle approche.

Keywords: RPL, multi parent, énergie, équilibrage, WSN, durée de vie

1 Introduction

Le problème du routage dans les réseaux de capteurs a été intensivement étudié durant la dernière décennie. RPL est maintenant le standard *de facto* pour l'Internet des Objets [W⁺12] et reflète l'évolution actuelle : introduire plus de redondance dans la structure de routage afin d'en améliorer la tolérance aux fautes. Cependant, la version actuelle de RPL présente selon nous deux voies d'améliorations.

Tout d'abord, le groupe de travail Roll s'est focalisé jusqu'à maintenant sur la création de mécanismes de routage efficaces. Nous avons maintenant à proposer des métriques et mécanismes afin de rendre RPL efficace en énergie : la topologie (i.e., le graphe dirigé acyclique - DODAG) doit être construit à partir d'un critère d'énergie. Chang *et al.* ont proposé de combiner l'énergie résiduelle et ETX [C⁺13], cependant une telle métrique ne reflète pas la durée de vie réelle. Par ailleurs, un noeud doit sélectionner **un** père préféré pour construire le DODAG sans boucle. Ainsi, les pères secondaires n'ont qu'un rôle de secours, lorsque le préféré disparaît. Hong *et al.* ont proposé de sélectionner le père préféré en exploitant le nombre de sauts et de choisir comme père relayant le trafic celui qui offre la meilleure qualité de lien [H⁺11]. Cependant, un petit nombre de noeuds relaiera la majorité du trafic, épuisant son énergie.

Nous proposons ici une nouvelle métrique exploitant la diversité de la topologie créée par RPL. Nous augmentons la durée de vie en équilibrant la charge entre les chemins.

2 Métrique de routage : Expected Lifetime (ELT)

Afin de construire une topologie équilibrée, nous proposons d'utiliser une nouvelle métrique : l'Expected Lifetime (ELT). ELT estime la durée de vie restante, i.e. temps avant qu'un noeud meurt s'il continue à relayer la même quantité de trafic. La métrique ne tient compte que de l'énergie consommée par les transmissions. Nous considérons que l'énergie consommée par les réceptions de paquets peut être incluse dans celle utilisée pour transmettre (un paquet reçu est relayé). Afin de calculer ELT, nous avons donc (cf. Tableau 1) :

1. le volume de trafic que le noeud N doit relayer ($T_{total}(N)$);
2. le ratio du trafic envoyé à chaque père P (α_P), où $\sum_{P \in Parent(N)} \alpha_P = 1$;
3. le nombre moyen de retransmissions vers chaque père ($ETX(N, P)$);
4. l'énergie résiduelle ($E_{res}(N)$) et l'énergie consommé par chaque bit transmis ($P_{Tx}/DATA_RATE$).

$ELT(X)$	Expected lifetime de X	$E_{res}(X)$	énergie résiduelle de X (en Joules)
α_P	Ratio du trafic envoyé à P	$T_{total}(X)$	Débit (bits/s) de X
$ETX(A,B)$	ETX du lien $A \rightarrow B$	Parents(X)	parents de X
$r_{X,B}$	Ratio de trafic envoyé par X au goulot d'étranglement B	$P_{Tx}(X)$	Puissance radio en mode transmission (en Watt ou Joules/s)
DATA_RATE	Le débit de transmission (bits/s)	Bottlenecks(X)	Goulots d'étranglement de X

TABLE 1: Notation

Finalement, un noeud N estime son ELT :

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

Considérons le que le noeud N doit s'associer au DODAG. Comme le goulot d'étranglement est le maillon faible (premier noeud à mourir), le noeud doit estimer l'impact de son trafic sur la durée de vie du goulot d'étranglement. Nous devons tenir compte le fait qu'un noeud envoie des paquets à plusieurs parents. Ainsi, il envoie un ratio de son trafic à chacun des noeuds goulots d'étranglement. Soit $r_{N,B}$ le ratio de trafic que N relaie au goulot d'étranglement B . Ainsi : $r_{N,B} = \sum_{P \in Parent(N)} (\alpha_P \times r_{P,B})$. Si nous exploitons l'équation 1, un noeud N peut maintenant estimer l'ELT d'un goulot d'étranglement B , en ajoutant son trafic :

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

L'information sur un goulot d'étranglement (e.g. trafic existant relayé par un goulot d'étranglement, son énergie résiduelle, etc.) est envoyée dans les paquets DIOs. L'ELT est ensuite mis à jour à chaque réception d'un DIO.

3 Construction de plusieurs chemins

Afin de construire une topologie de routage, un noeud a besoin de choisir son prochain saut (i.e. père préféré) pour envoyer les paquets. En sélectionnant son père préféré, un noeud doit considérer à la fois sa durée de vie et celle de ses goulots d'étranglement, afin d'estimer si le noeud lui-même deviendra un nouveau goulot d'étranglement énergétique. Cependant, il n'est pas possible de savoir a priori quel père sera choisi pour relayer les paquets afin de choisir tous les pères. Ainsi, nous ne pouvons pas estimer finement la durée de vie du noeuds et de ses goulots d'étranglement.

Nous proposons que durant la sélection du père préféré, un noeud suppose que tout le trafic sera envoyé à ce père préféré. Nous sous-estimons donc ainsi la durée de vie des goulots d'étranglement correspondant, mais maximisons la durée de vie du goulot d'étranglement le plus contraint. En d'autres mots, nous considérons le cas pire de l'équilibrage énergétique.

Nous proposons l'algorithme 1 pour sélectionner le père préféré. Pour chaque père possible, un noeud N va :

1. calculer ELT pour tous les noeuds les plus contraints annoncés par son père P , comme si N envoyait tout son trafic à ce père, et sauvegarder la valeur minimale (ligne 4) ;
2. calculer sa propre durée de vie et vérifier la plus grosse contrainte (ligne 5) ;
3. enlever le trafic vers ce père pour testers les autres pères possibles (ligne 10) ;
4. choisir le père préféré qui maximise les durées de vie, la sienne incluse (lignes 6, 7, 8).

Exploiter plusieurs parents avec RPL pour en améliorer la stabilité

Algorithm 1: Sélection du père préféré

```

Data:  $N$ 
Result: preferred_parent of  $N$ 
1  $max\_elt \leftarrow 0$ ;
2 for  $P \in Parents(N)$  do
   // tout le trafic est envoyé à P
3    $\alpha_P \leftarrow 1$ ;

   // chercher l'ELT min (le plus
   // contraint & moi)
4    $min\_elt \leftarrow \min_{B \in Bottlenecks(P)} \{ELT(B)\}$ ;
5    $min\_elt \leftarrow \min\{min\_elt, ELT(N)\}$ ;

   // est-ce le meilleur père?
6   if  $max\_elt < min\_elt$  then
7      $max\_elt \leftarrow min\_elt$ ;
8     preferred_parent  $\leftarrow P$ ;
9   end

   // tester les autres pères
10   $\alpha_P \leftarrow 0$ ;
11 end
12 return preferred_parent;

```

Algorithm 2: Recherche des poids

```

Data:  $N, load\_step$ 
Result: compute  $\{\alpha_P\}_{P \in Parents(N)}$  — ratio à envoyer à
   chacun des pères;
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      $\alpha_P \leftarrow \alpha_P + load\_step$ ;

     // chercher l'ELT minimum avec
     // ce nouveau poids
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

    // tester chaque père
11     $\alpha_P \leftarrow \alpha_P - load\_step$ ;
12  end
13   $\alpha_{parent\_max} \leftarrow \alpha_{parent\_max} + load\_step$ ;
14 end

```

Après avoir choisi le père préféré, un noeud doit calculer sa distance relative au routeur de bordure, i.e., son *rang*. Afin d'éviter la création de boucles, le rang dans le DODAG doit être monotone croissant vers les feuilles. Puisque l'ELT représente une métrique minimale le long d'un chemin, elle ne peut pas être utilisée directement pour créer les routes. Nous proposons qu'un noeud calcule son rang en ajoutant une constante au rang de son père préféré : $Rank(N) = Rank(P_N) + Step \times \text{MinHopRankIncrease}$, où *Step* est une constant et *MinHopRankIncrease* le paramètre de RPL [W⁺12].

4 Equilibrer la charge

Un noeud doit répartir ensuite son trafic sur tous les chemins. Nous proposons donc ici un algorithme glouton. Un noeud N distribue une fraction de son trafic à chaque itération pour maintenir le meilleur équilibre énergétique entre tous les goulots d'étranglement qu'il a. Un noeud divise son trafic global en $\frac{1}{load_step}$ parties égales et assigne de façon séquentielle chaque partie au père le moins contraint.

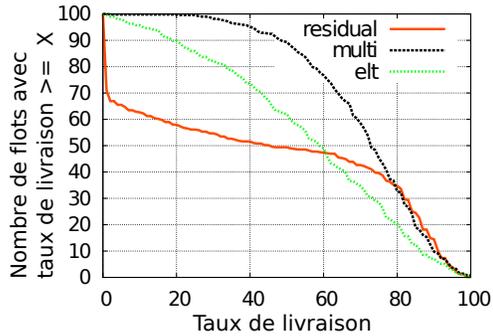
L'algorithme 2 définit plus formellement l'approche :

1. N considère chaque père (ligne 1) ;
2. il calcule l'ELT minimum qu'il obtiendrait en augmentant son poids de *load_step* (ligne 4). Il considère l'ELT des goulots d'étranglement (ligne 5) et son propre ELT (ligne 6) ;
3. Si la valeur minimale maximise la durée de vie, il mémorise ce père (ligne 7-10) ;
4. N teste itérativement chaque père en changeant le poids (ligne 11) ;
5. Finalement, N assigne *load_step* au meilleur père (ligne 13).

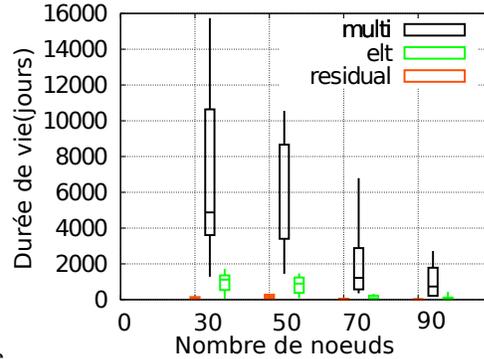
Nous notons qu'une petite valeur de *load_step* équilibre plus finement la charge mais requiert naturellement plus d'étapes pour le calcul, augmentant la complexité pour des noeuds contraints.

5 Résultats de simulation

Nous avons simulé RPL avec WSNNet (<http://wsnet.gforge.inria.fr/>) avec 50 noeuds sur une surface de 300m par 300m. Les résultats sont moyennés sur 10 simulations obtenues avec des topologies aléatoires. Pour le trafic, nous considérons des flux CBR de 1 paquet par minute. Nous avons comparé notre solution avec RPL standard, avec un DODAG construit sur l'énergie résiduelle et sur ELT.



(a) Fonction de distribution cumulative complémentaire du taux de livraison



(b) Durée de vie en fonction de la densité

Nous avons tout d'abord évalué la fiabilité en mesurant le ratio des paquets reçus par le puits. La figure 1a illustre la fonction de distribution cumulative complémentaire du taux de livraison. Avec RPL standard, ELT permet d'atteindre la meilleure fiabilité. L'énergie résiduelle impacte la fiabilité, puisqu'elle tend à privilégier les noeuds à forte énergie, sans tenir compte de la qualité des liens radio. Nous pouvons aussi noter comment le multichemin tire parti de l'exploitation de plusieurs pères.

Dans un deuxième temps, nous avons mesuré la durée de vie (i.e. temps avant la première mort) en fonction de la densité (i.e. surface constante, nombre de noeuds croissant). Nous pouvons observer dans la figure 1b que notre proposition permet bien d'améliorer la performance par rapport à un RPL mono-parent. Les routes équilibrent mieux la charge : les décisions de routage ne sont plus binaires.

6 Conclusion et Perspectives

Nous avons ici proposé une nouvelle métrique prolongeant la durée de vie : l'*Expected Lifetime*. Cette version modifiée de RPL exploite tous les pères et équilibre efficacement la consommation en énergie du réseau entier.

Nous investiguons actuellement comment RPL peut prendre en compte des incertitudes de mesure dans l'estimation de la qualité d'un lien. En effet, le lien radio est stochastique par nature, et les routes construites par RPL ne devaient changer que quand la qualité varie significativement. Nous souhaitons aussi mesurer expérimentalement les performances de notre proposition.

Références

- [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.