



HAL
open science

WURST: routage pour réseaux IoT multi-technologies purement asynchrones

Léon Niess, Julien Montavont, Pascal Mérindol

► **To cite this version:**

Léon Niess, Julien Montavont, Pascal Mérindol. WURST: routage pour réseaux IoT multi-technologies purement asynchrones. CoRes 2023 - 8èmes Rencontres Francophones sur la Conception de protocoles, l'évaluation de performances et l'expérimentation de Réseaux de communication, May 2023, Cargese, France. hal-04085522

HAL Id: hal-04085522

<https://hal.science/hal-04085522>

Submitted on 29 Apr 2023

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.

WURST: routage pour réseaux IoT multi-technologies purement asynchrones

Léon Niess, Julien Montavont et Pascal Mérindol

ICube, Université de Strasbourg, France

Les wake-up radios (WUR) révolutionnent les réseaux sans fil en permettant de déployer des réseaux purement asynchrones efficaces en énergie. Néanmoins, le déploiement des routes sur ce type de réseaux est complexe en raison de la dualité entre le voisinage WUR et celui de la radio principale. Nous résolvons ce défi en proposant le protocole de routage WURST qui fusionne les graphes formés par chaque technologie radio en un nouveau graphe bi-valué avec une contrainte d'existence sur ses arcs. Nos simulations indiquent que WURST dispose d'un meilleur temps de convergence et d'un meilleur ratio de tailles de chemins entre les deux radios en comparaison d'un protocole comme CTP-WUR.

Mots-clefs : IoT, RCSF, Routage, Wake-Up Radio

1 Introduction

L'un des défis de l'IoT consiste à réduire la consommation énergétique des nœuds afin de maximiser la durée de vie du réseau. Pour cela, le transmetteur radio est soumis à un cycle de travail alternant les périodes actives et de veille, cependant cette stratégie impose un compromis entre performances et consommation énergétique. Des communications purement asynchrones et frugales sont toutefois possibles grâce à l'émergence des Wake-Up Radios (WUR). Elles se présentent sous la forme d'un récepteur radio à très basse consommation (quelques μW contre quelques dizaines de mW habituellement [MJS⁺16]) ce qui permet de le maintenir actif sans incidence significative sur la consommation énergétique. Néanmoins, le débit proposé est très limité (quelques kb/s [MJS⁺16]) ce qui restreint son utilisation au plan de contrôle et nécessite un transmetteur radio traditionnel pour assurer le plan de données. De nombreux protocoles MAC utilisant cette dualité ont été proposés dans la littérature [EHDSM⁺22] : l'émetteur réveille la radio principale de son destinataire via le canal WUR avant toute émission sur la radio principale. L'utilisation des WUR est envisagé dans de nombreuses technologies radio, notamment en Wi-Fi via la norme IEEE 802.11ba.

Néanmoins, une seconde limitation des WUR réside dans la faible sensibilité des récepteurs, qui se traduit par des portées radio très faible : entre 2,5 et 25 fois plus faible qu'avec un transmetteur 802.15.4 [MJS⁺16]. Le voisinage d'un nœud varie grandement en qualité en fonction du transmetteur utilisé. Pour assurer une bonne connectivité via les *meilleurs* chemins de la radio principale, il est nécessaire de correctement relayer les messages de contrôle transmis sur le canal WUR. Ce relayage articulé sur les WUR introduit un problème de routage à deux niveaux qui constitue le cœur de notre travail : comment calculer et déployer des chemins efficaces sur une telle architecture duale tenant compte des spécificités de chaque radio ? Nous proposons de résoudre ce problème en fusionnant les graphes formés par chaque technologie radio et en imposant une contrainte sur la *qualité* des chemins WUR (notamment leur longueur). Notre protocole, WURST pour WUR Shortest Tree, est capable d'articuler la métrique WUR au service de la radio principale et vice-versa. Alors que la radio WUR contrôle la principale, cette dernière permet d'échanger des informations sur la dynamique locale du réseau WUR. L'objectif de WURST est d'alléger la charge des tables de commutation et le coût de contrôle de la topologie, en fiabilisant les sous connexions WUR mais au prix d'une potentielle perte d'optimalité sur le réseau de la radio principale. WURST est évalué avec le simulateur Omnet++ et comparé à une solution de la littérature. Nos résultats indiquent que WURST réduit le temps de convergence et la taille des tables de routage, sans pénaliser considérablement la qualité des chemins : la longueur des chemins sur la radio principale est seulement doublée au pire, alors que le canal WUR bénéficie quant à lui d'une réduction d'un facteur 4 au mieux.

2 Formalisation du problème et état de l'art

Considérons un graphe $G_1 = (V_1, E_1, w_1)$ formé par la radio principale et un graphe $G_2 = (V_2, E_2, w_2)$ formé par la WUR comme illustré sur la figure 1. Nous faisons l'hypothèse que chaque graphe est connexe et que tous les nœuds du réseau sont munis des deux types de radio, d'où $V_1 = V_2$. Notre hypothèse de base est que G_2 est connexe : pour chaque arc dans E_1 , il existe un chemin dans G_2 entre ses extrémités, et si nous ajoutons que $E_2 \subset E_1$, alors G_1 est aussi connexe et n'importe quel puits est atteignable dans G_1 depuis n'importe quel nœud avec des sous chemins dans G_2 entre chaque paire de sommets de chaque arc du chemin dans G_1 . Plus précisément, nous faisons l'hypothèse suivante : il existe un arbre $A = (V_1, A_1)$ dans G_1 enraciné en un sommet puits tel qu'entre chacune des extrémités de chaque arc de A_1 , il existe un chemin p dans G_2 tel que la distance de p vérifie une contrainte C^\dagger . Enfin, $G_i, i \in \{1, 2\}$ possède une fonction de valuation $w_i : E_i \rightarrow \mathbb{N}_+$ décrivant la pondération des liens pour définir une métrique arbitraire (l'exemple le plus naïf étant la minimisation du nombre de sauts pour chaque graphe).

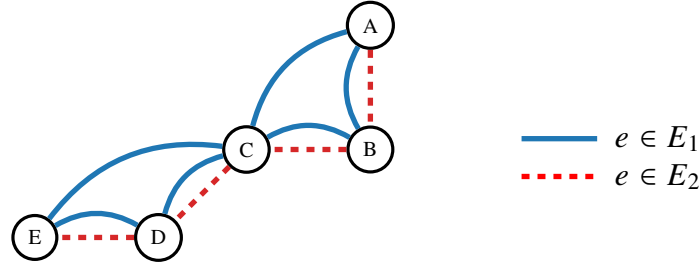


FIGURE 1 : Graphes formés par la radio principale ($G_1 = (V_1, E_1)$) et par la WUR ($G_2 = (V_2, E_2)$)

Une stratégie possible pour déployer des chemins sur une telle architecture à deux niveaux consiste à considérer exclusivement G_2 , tel que proposé dans [PSTT14]. Mais cette solution augmente le nombre de transmissions sur la radio principale, de même que la consommation énergétique, au risque d'annihiler les gains apportés par les WUR. Avec une telle stratégie appliquée à la figure 1, l'envoi de données entre A et E nécessiterait 4 transmissions sur G_1 alors qu'il est possible de réduire ce nombre à 2 avec le chemin $p\{(A, C), (C, E)\}$. Une meilleure option consiste à calculer les meilleurs chemins sur G_1 et relayer les messages de réveil WUR jusqu'au prochain saut ; avec l'exemple précédent, B doit relayer le message WUR de A à C afin d'utiliser le chemin $p\{(A, C), (C, E)\}$ de G_1 . T-ROME [KSRS17] utilise une telle approche mais se base sur un routage statique sur G_2 , avec l'hypothèse forte de liens radios stables dans le temps. Enfin, CTP-WUR [SB19] permet de construire une table de routage complète qui indique, pour chaque destination, le prochain saut et le coût pour joindre cette destination via la radio principale et via la WUR. Cette table est initialisée avec le 1-voisinage de chaque nœud découvert via des diffusions sur chaque radio, puis complétée par le k -voisinage de chaque nœud via l'échange de vecteurs de distance sur la radio principale. Néanmoins, le calcul se fait uniquement sur G_1 indépendamment de G_2 , au risque d'utiliser des chemins peu fiables sur G_2 . Or nous avons montré que la longueur des chemins dans G_2 influence fortement les performances [SMPN18], au risque de réduire à néant l'intérêt des WUR. Fort de ce constat, nous proposons d'améliorer CTP-WUR en fusionnant G_1 et G_2 afin de conserver les dépendances entre les deux graphes et ainsi d'imposer une contrainte entre les deux dans le calcul des plus courts chemins sur G_1 .

3 Contribution : un arbre de sous-chemins WUR fiables

Notre premier objectif est la construction d'un graphe dirigé et bi-valué \mathcal{G} dont l'ensemble des sommets est simplement $\mathcal{V} = V_1 = V_2$ mais où l'ensemble des arcs, noté $\mathcal{E} \subset E_1$, est doublement valué et soumis à une contrainte d'existence tel que :

- $e := (x, y) \in \mathcal{E}$ si et seulement si il existe un chemin entre x et y dans G_2 vérifiant certaines propriétés, typiquement sur sa longueur (par ex., inférieur à un certain nombre de sauts pour la fiabilité) ;

[†]. C restreint la connectivité aux sous chemins considérés fiables et efficaces dans G_2 mais induit une potentielle perte de connectivité si l'hypothèse de départ n'est pas renforcée.

- la fonction de bi-valuation $w(e)$ définie tel que $w(x, y) = (a, b)$ avec a renvoyant $w_1(e)$ (si $e \in \mathcal{E}$), et b indiquant la meilleure distance entre x et y dans G_2 soumis à la contrainte C .

Sur base de ce graphe, l'objectif est alors de calculer les plus courts chemins depuis chaque nœud vers une racine donnée, le puits : tant qu'il n'existe pas de sous chemins WUR pour connecter les deux extrémités d'un arc dans E_1 , la (meilleure) distance qu'il offre vers le puits n'est pas relayée pour ce lien. La fonction de comparaison des distances des chemins est définie lexicographiquement, c'est-à-dire qu'un chemin p est dit (strictement) plus court qu'un autre, p' , soit parce que sa distance sur la première valuation (radio principale) est strictement inférieure à celle de p' , soit elle est égale mais alors (strictement) plus petite sur la seconde valuation (distance WUR). En cas d'égalité sur les deux, nous considérons les chemins égaux (routage multi-chemins) mais il est possible d'utiliser un élément arbitraire pour les départager.

En pratique, pour obtenir cet arbre des plus courts chemins enraciné sur le puits (ou graphe acyclique avec chemins de coûts égaux), l'ensemble du graphe \mathcal{G} n'a pas besoin d'être connu de chaque nœud. Nous avons développé un nouvel algorithme de routage dont le fonctionnement à haut niveau est le suivant :

1. tous les nœuds émettent un signal WUR afin de connaître leurs voisins directs ;
2. cette information (ensemble des liens WUR dirigés vers un nœud) est alors diffusée via la radio principale (voire relayée à une certaine profondeur pour vérifier C) ;
3. le puits s'annonce via la radio principale et, afin de définir son parent préféré de manière itérative, chaque nœud applique alors un protocole de type gradient mais avec une valuation lexicographique.

Il suffit ainsi à chaque nœud de connaître le sous-graphe incluant ses meilleurs chemins WUR, selon notre définition spécifique et contrainte, vers le puits. Notre hypothèse est que ce sous-graphe est local par construction pour des raisons de fiabilité. La métrique distance sur la radio principale est manipulée avec des vecteurs de distances standards pour le puits alors que la métrique WUR est exploitable différemment. Chaque nœud dispose, grâce à la seconde étape, d'une carte partielle de son voisinage proche, en fonction de la profondeur préétablie et de la portée de radio principale rapportée à celle de la WUR. Cette information leur permet de déterminer leurs plus courts chemins WUR vers leurs voisinages proches (ceux atteignables sous la contrainte de fiabilité) et ainsi d'appliquer la comparaison de chemins adéquate sur les deux critères définis. Chaque nœud dispose finalement d'une table de routage secondaire WUR vers ses voisins proches, avec le meilleur prochain saut et la meilleure distance connus vers ceux-ci, et d'une entrée principale sous la forme d'un quadruplet vers le puits. Cette entrée principale contient un prochain saut principal (le voisin direct sur cette radio), un prochain saut secondaire (le voisin direct WUR) ainsi que leurs meilleures distances respectives pour les comparer aux annonces de routage.

4 Simulations et résultats préliminaires

Nous avons implémenté le protocole WURST dans le simulateur OMNeT++ en complément d'un protocole de routage naïf proche de CTP-WUR dans lequel le voisinage lié à une radio est exclusivement communiqué via cette même radio. Le réseau est organisé en une grille régulière dans laquelle le puits est situé dans le coin haut à gauche. Les nœuds disposent d'une WUR (simulant le prototype présenté dans [MJS⁺16]) et d'une radio de type IEEE 802.15.4. La portée WUR permet aux nœuds d'avoir entre 2 et 4 voisins alors que la portée de la radio principale (4 fois supérieure à celle de la WUR) forme des réseaux de diamètre 1 à 4. La contrainte sur la taille des chemins sur G_2 est ici arbitrairement fixée à 3. Les nœuds démarrent aléatoirement après 1 à 5 sec. de simulation et transmettent leur voisinage sur chaque canal toutes les 2 sec. Les résultats présentés ci-après sont des moyennes issues de 100 simulations pour chaque configuration avec des intervalles de confiance à 95%.

La figure 2a montre que WURST permet une convergence rapide, de l'ordre de 11,59 sec. pour une grille 12x12 soit 6 cycles d'échanges de messages de moins que la solution naïve. Cette performance est obtenue par la fusion des deux graphes sous contrainte, ce qui limite la taille des tables de routage finales (144 entrées pour la solution naïve vs. 18,59 entrées en moyenne pour WURST sur la grille 12x12). De plus, WURST transmet le k-voisinage exclusivement via la radio principale dont le débit et la stabilité sont largement supérieurs au canal WUR (la solution naïve présente des temps de transmission supérieurs à 200 ms sur la WUR). Enfin la figure 2b présente la taille normalisée par rapport à la solution naïve des chemins construits

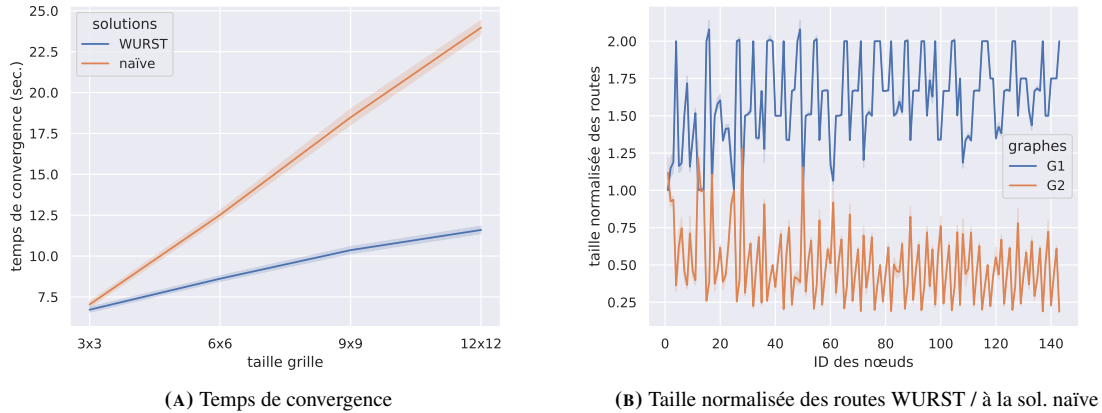


FIGURE 2 : Résultats préliminaires pour 100 simulations de chaque configuration

sur chaque graphe par WURST. La contrainte imposée lors de la fusion des graphes permet d'établir des routes environ 2 fois plus courtes sur G_2 , contre des routes environ 1,6 fois plus longues sur G_1 . WURST augmente légèrement la consommation énergétique, mais la solution naïve utilise de très longues routes sur G_2 (jusqu'à 16 sauts) soit des routes dont le taux de livraison pourrait tendre vers 0 et faire exploser la consommation énergétique en raison des nombreuses retransmissions [SMPN18].

5 Conclusion et perspectives

Notre solution de routage à deux niveaux offre un compromis entre la fiabilité des sous connexions WUR et la longueur des chemins de la radio principale. Notre protocole peut articuler les deux mécanismes de diffusion de plusieurs manières. Par exemple en combinant plusieurs types de plan de contrôle : état des liens partiel pour WUR et vecteur de chemins lexicographique pour la radio principale. Nous pensons qu'il existe un fort gain potentiel pour gérer la dynamique de notre graphe doublement valué. Notamment en ce qui concerne le coût énergétique de la mise en œuvre de la nouvelle structure de routage sous-jacente à chaque changement. Nous vérifierons ces intuitions dans nos futurs travaux. WURST est une solution originale mais préliminaire à ce que nous souhaitons expérimenter.

Références

- [EHDSM⁺22] N. El Hoda Djidi, S. Sampayo, J. Montavont, A. Courtay, M. Gautier, O. Berder, and T. Noël. The revenge of asynchronous protocols : Wake-up Radio-based Multi-hop Multi-channel MAC protocol for WSN. In *proc. of IEEE WCNC*, 2022.
- [KSRS17] T. Kumberg, M. Schink, L.M. Reindl, and C. Schindelbauer. T-ROME : A simple and energy efficient tree routing protocol for low-power wake-up receivers. *Elsevier Ad Hoc Networks*, 59, 2017.
- [MJS⁺16] M. Magno, V. Jelicic, B. Srbinovski, V. Bilas, E. Popovici, and L. Benini. Design, implementation, and performance evaluation of a flexible low-latency nanowatt wake-up radio receiver. *IEEE Transactions on Industrial Informatics*, 12(2), 2016.
- [PSTT14] C. Petrioli, D. Spenza, P. Tommasino, and A. Trifiletti. A Novel Wake-Up Receiver with Addressing Capability for Wireless Sensor Nodes. In *proc. of the IEEE DCOSS*, 2014.
- [SB19] A.V. Sheshashayee and S. Basagni. Multi-hop Wake-up Radio Relaying for the Collection Tree Protocol. In *proc. of the IEEE VTC*, 2019.
- [SMPN18] S. L. Sampayo, J. Montavont, F. Prégaldiny, and T. Noël. Is Wake-Up Radio the Ultimate Solution to the Latency-Energy Tradeoff in Multi-hop Wireless Sensor Networks ? In *proc. of IEEE WiMob*, 2018.