



HAL
open science

De la Survie Énergétique des Réseaux de Capteurs

Quentin Bramas, Wilfried Dron, Mariem Ben Fadhl, Khalil Hachicha, Patrick Garda, Sébastien Tixeuil

► **To cite this version:**

Quentin Bramas, Wilfried Dron, Mariem Ben Fadhl, Khalil Hachicha, Patrick Garda, et al.. De la Survie Énergétique des Réseaux de Capteurs. ALGOTEL 2015 - 17èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, Jun 2015, Beaune, France. hal-01144309

HAL Id: hal-01144309

<https://hal.science/hal-01144309>

Submitted on 21 Apr 2015

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.

De la Survie Énergétique des Réseaux de Capteurs[†]

Quentin Bramas^{1,2}, Wilfried Dron^{1,2}, Mariem Ben Fadhl^{1,2},
Khalil Hachicha^{1,2}, Patrick Garda^{1,2} et Sébastien Tixeuil^{1,2,3}

¹Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, F-75005, Paris, France

²CNRS, UMR 7606, LIP6, F-75005, Paris, France

³Institut Universitaire de France

Les applications récentes des réseaux de capteurs sans fil requièrent des appareils de petite taille, autonomes en énergie, et possédant une longue durée de vie. Il est donc crucial de simuler efficacement et fidèlement la durée de vie des capteurs au sein de ces réseaux. Les simulateurs de réseaux existants implémentent souvent des modèles simplifiés de consommation et de batterie. Par ailleurs, les modèles très réalistes de batterie, implémentés dans des simulateurs dédiés, sont trop complexes pour être utilisés pour la simulation de réseaux de capteurs de grande taille, et dont la durée à simuler peut atteindre des mois, voire des années.

Dans ce papier nous présentons WiSeBat, un modèle de batterie et de consommation d'énergie optimisé pour les réseaux de capteurs, et son implémentation dans le simulateur WSNET. Nos résultats montrent que pour un surcoût de simulation minimale, ce modèle permet d'obtenir une précision de 86 – 96% (contre une erreur supérieure à 2600% avec le modèle par défaut de WSNET) par rapport à un capteur réel. Dans un deuxième temps, nous utilisons WiSeBat pour comparer l'efficacité énergétique de X-MAC, ContikiMAC et 802.15.4.

Keywords: Réseau de Capteurs, Simulateur, Consommation Énergétique

1 Introduction

De nombreuses applications des réseaux de capteurs, notamment dans le domaine médical, requièrent des appareils capables de fonctionner sur de longues périodes, typiquement, de plusieurs jours à plusieurs années. Cependant, la taille des capteurs étant limitée, il est crucial de simuler fidèlement la batterie et la consommation des composants pour choisir les paramètres (notamment, le ratio entre la durée d'éveil et la durée de sommeil) qui répondront aux exigences de l'application concernant la durée de vie du réseau.

Travaux Connexes Il existe de nombreux simulateurs de réseaux orientés vers des types de réseaux spécifiques ou bien généralistes (mais l'étude précise de chaque simulateur n'est pas le but de cet article). Même si la définition de la durée de vie d'un réseau dépend de l'application [DD09], elle est déterminée à partir de la durée de vie de chacun des capteurs. L'estimation de la durée de vie d'un appareil repose à la fois sur le modèle de batterie et le modèle de consommation [DDV⁺14].

De manière simplifiée, les simulateurs de réseaux populaires comme NS2/3 ou WSNET prennent en compte la consommation de la radio uniquement, couplée à un modèle de batterie linéaire. Cela revient à multiplier le nombre de bits transmis et de bits reçus par la consommation de la radio en mode TX et RX respectivement. Ce modèle très simple, utilisé par défaut, n'est cependant pas du tout réaliste, comme nous le montrons ci-après.

D'autres solutions comme SENSE ou PowerTOSSIMz prennent en compte chaque composant de l'appareil, couplé à un modèle de batterie non linéaire. Même si cette approche peut paraître plus réaliste, elle ne tient pas compte des variations de voltage de la batterie, qui elles non plus ne sont pas linéaires. Cela entraîne des imprécisions, notamment pour déterminer à quel moment un nœud s'arrête de fonctionner.

[†]Ce travail a été effectué dans le cadre du Labex SMART soutenu par les financements de l'état Français gérés par l'ANR sous le programme Investissements d'Avenir sous la référence ANR-11-IDEX-0004-02. Aussi, il est soutenu en partie par le LINCS.

Notre Contribution Dans ce papier nous adaptons un modèle existant de batterie pour les réseaux de capteurs sans fil. Nous le couplons avec un modèle de consommation prenant en compte chaque composant du capteur pour former le module complet que nous appelons WiSeBat, dont l’implémentation de départ est intégrée au simulateur WSNET. Nous le validons par des mesures sur des capteurs réels qui montrent que le module WiSeBat (faisant référence à la fois au modèle de batterie et de consommation) surpasse le modèle par défaut de WSNET. Pour finir, nous utilisons WiSeBat pour étudier la durée de vie des capteurs dans plusieurs scénarios. Nous retrouvons ainsi des résultats bien connus, comme la supériorité de Contiki-MAC sur X-MAC, mais nous sommes également en mesure de quantifier précisément les performances de ContikiMAC et de 802.15.4 MAC (en mode beacon avec duty-cycle) dans une configuration plus complexe.

2 Le Module WiSeBat

Modèle de Consommation. Dans les réseaux de capteurs sans fil, la radio qui n’est pas constamment allumée, n’est pas nécessairement le composant qui utilise le plus la batterie. C’est pourquoi WiSeBat offre la possibilité lors de l’initialisation de la simulation d’enregistrer un ensemble de composants \mathcal{X} (par exemple : les capteurs, le micro-contrôleur, les LEDs, etc). Chaque composant $x \in \mathcal{X}$ est associé à une fonction qui donne sa consommation $i_x(m)$ en mA en fonction du mode de fonctionnement m dans lequel il se trouve. Lors de la simulation, la couche application peut mettre à jour le mode de fonctionnement de chaque composant enregistré (le mode de la radio étant directement modifié par la couche MAC). Si on l’associe à une batterie au comportement linéaire, la quantité d’énergie consommée en mAh par les composants est donnée par la formule suivante (où \mathcal{M}_x est l’ensemble des états possibles du composant x et $T_x(m)$ est le temps, en heures, passé dans l’état m) :

$$C = \sum_{x \in \mathcal{X}} \sum_{m \in \mathcal{M}_x} i_x(m) T_x(m)$$

Modèle de Batterie. Le premier comportement non linéaire de la batterie pris en compte par WiSeBat est que lorsque le courant instantané augmente, la batterie se comporte comme si la capacité totale diminuait. En effet, on peut extraire des données constructeur une fonction décroissante $C_{eq} : i \mapsto C_{eq}(i)$ telle que si le courant est constant égal à i alors la capacité mesurée est $C_{eq}(i)$. La capacité nominale $C_{nominal}$ de la batterie est en fait la limite en 0 de cette fonction (elle est généralement mesurée pour un courant faible appelé courant nominal). Ce comportement est capturé en utilisant la définition donnée par Dron *et al.* [DDV⁺14] de courant équivalent $i_{eq}(t)$ au courant total $i(t)$ fourni par la batterie au temps t :

$$i_{eq}(i(t)) = \frac{C_{nominal}}{C_{eq}(i(t))} i(t) \quad (1)$$

En effet, si le courant consommé par les composants est constant égal à i , la batterie sera épuisée (*i.e.*, sera incapable de fournir du courant) après un temps T tel que $i \times T = C_{eq}(i)$.

Le second comportement non linéaire de la batterie modélisé par WiSeBat est la variation de voltage au cours de sa décharge, en fonction du courant fourni. Les données du constructeur permettent de déduire les fonctions E_{mf} et R qui donnent la force électromotrice en Volt et la résistance interne en Ohm de la batterie en fonction de sa capacité résiduelle. Si au temps t la batterie a une capacité résiduelle $C(t)$ et fournit un courant $i(t)$, alors le voltage $V(t)$ à ses bornes est donné par la formule suivante :

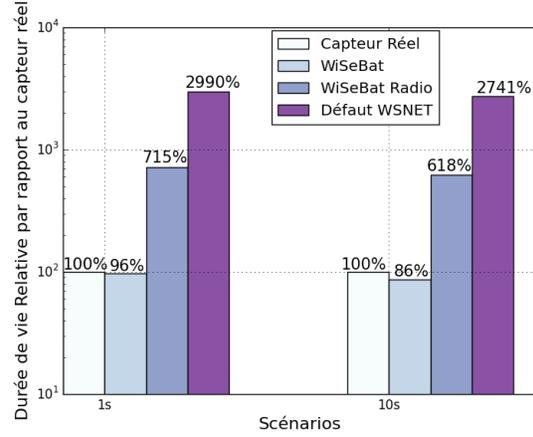
$$V(t) = E_{mf}(C(t)) - i(t)R(C(t))$$

Ce voltage affecte les composants d’un capteur de deux manières. Premièrement, la consommation des composants dépend du voltage à ses bornes (*i.e.*, pour $x \in \mathcal{X}$, i_x dépend en fait du mode de x et de la tension). Deuxièmement chaque composant nécessite un voltage minimum pour fonctionner, appelé *cut-off*. C’est-à-dire qu’un capteur va cesser de fonctionner *avant* que la batterie soit totalement déchargée.

Grâce aux équations précédentes, le module WiSeBat met à jour les valeurs de $i(t)$, $C(t)$ et $V(t)$ de manière discrète à des instants t_1, t_2, \dots en s’appuyant sur les relations suivantes (où pour un composant x , $m_x(t)$ est son mode à l’instant t et i_x est la fonction de consommation en fonction du voltage et du mode) :

Composant	Mode	Courant
Micro-controller voltage <i>cut-off</i> : 1.65V	Run	10 mA
	Sleep	2.4 uA
Radio transeiver voltage <i>cut-off</i> : 1.8V	Tx	7.7 mA
	Rx	5.7 mA
	Idle	1.7 mA
	Sleep	1.1 uA
Capteur de pression voltage <i>cut-off</i> : 1.71V	Init	42 mA
	Read	5 mA
	Sleep	0.5 uA
Led	On	7.6 mA
Gestionnaire d'alimentation voltage <i>cut-off</i> : 2.4V	Éfficacité : 95 - 99%	

1.1: Caractéristiques des composants



1.2: Précision des modèles dans deux scénarios.

$$C(t_i) = C(t_{i-1}) - i_{eq}(i(t_{i-1}))(t_i - t_{i-1}) \quad (2)$$

$$V(t_i) = E_{mf}(C(t_i)) - i(t_{i-1})R(C(t_i)) \quad (3)$$

$$i(t_i) = \sum_{x \in X} i_x(V(t_i), m_x(t_i)) \quad (4)$$

Les relations sont liées de manière circulaire car le voltage dépend de l'intensité et inversement. C'est pourquoi, il faut que plusieurs mises à jours soient effectuées après chaque changement de mode, jusqu'à ce que les valeurs se stabilisent. De cette manière, l'erreur entre le modèle et les valeurs calculées de $C(t)$, $V(t)$, et $i(t)$ peut être maintenue en dessous d'une valeur arbitraire $\epsilon > 0$.

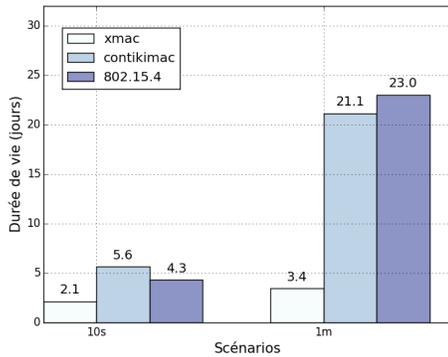
3 Validation Expérimentale

Pour évaluer WiSeBat nous avons mesuré la durée de vie d'un capteur réel avec une application et une architecture réaliste, et nous l'avons comparé à la durée de vie simulée par WiSeBat dans le simulateur WSNET. Pour cela nous avons implémenté dans WSNET l'application et l'architecture présente dans le capteur. Les composants du capteur ainsi que leur consommation sont présentés dans le tableau 1.1. Les résultats donnés dans la figure 1.2 représentent la durée de vie simulée des capteurs (fournie par WiSeBat et par le modèle par défaut de WSNET, respectivement) par rapport au capteur réel, dans deux scénarios (une transmission toutes les secondes et une transmission toutes les dix secondes). On remarque que le modèle par défaut de WSNET, utilisant une batterie au comportement linéaire et ne prenant en compte que la radio, surestime la durée de vie du capteur de plus de 2600%. Pourtant, ce modèle est très présent dans la littérature pour évaluer les performances énergétiques des protocoles réseaux. L'utilisation d'un modèle de batterie au comportement non linéaire améliore significativement les performances, puisque la durée de vie simulée n'est plus que 7 fois supérieure à la durée de vie mesurée sur le capteur réel. Pour finir, la prise en compte de tous les composants (ce qui correspond au module complet WiSeBat) permet alors de sous-estimer la durée de vie avec une erreur comprise entre 4% et 14%. C'est un point important pour la production effective des capteurs : une sur-estimation aboutit à l'utilisation de batteries moins efficaces dans les capteurs menant à une défaillance largement prématurée du réseau (par rapport au temps estimé par la simulation). Au contraire, la sous-estimation de la durée de vie garantit que le capteur réel dure au moins le temps attendu. Lors de ces simulations, WiSeBat a représenté entre 0.1% et 30% du temps de simulation, ce qui encourage son utilisation systématique.

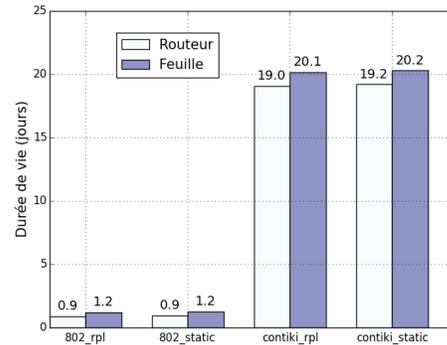
4 Simulations

Nous utilisons maintenant notre modèle validé pour comparer différentes piles réseau. Les nœuds simulés ont les mêmes caractéristiques que lors de l'évaluation présentée Section 3.

Scénarios avec un nœud sur batterie. On considère un réseau comportant un nœud feuille fonctionnant sur batterie et un nœud coordinateur (alimenté par une source extérieure). Toutes les dix secondes (resp. toutes les minutes), le nœud feuille effectue une mesure de pression puis envoie la valeur au coordinateur. La



2.1: Durée de vie du nœud en fonction de la couche MAC dans deux scénarios : un nœud transmet au coordinateur un message toutes les dix secondes ou toutes les minutes.



2.2: Durée de vie des nœuds feuilles et routeur en fonction de la couche MAC et de la couche routage (un nœud transmet au coordinateur un message toutes les minutes).

figure 2.1 donne la durée de vie du nœud en fonction de la couche MAC utilisée (X-MAC, ContikiMAC ou 802.15.4) pour les deux scénarios. Tout d'abord, comme cela a déjà été montré par des études précédentes [MB14], on remarque les meilleures performances de ContikiMAC par rapport à X-MAC. De plus, de manière prévisible, on montre que la couche MAC 802.15.4 est plus efficace lorsque le nœud passe plus de temps endormi. En effet, pendant les phases de sommeil, ContikiMAC continue d'effectuer régulièrement des évaluations de canal libre (CCA) pour vérifier la présence de messages, ce que ne fait pas 802.15.4 pour les nœuds feuilles.

Scénarios avec deux nœuds sur batterie. Dans ces scénarios, un nœud feuille, un nœud routeur et le coordinateur forment une ligne. Les données transmises par le nœud feuille au coordinateur doivent être transférées par le nœud routeur intermédiaire. Nous étudions le scénario où les nœuds transmettent des données toutes les minutes. La pile réseau est composée d'une couche MAC (ContikiMAC ou 802.15.4), et d'une couche routage (statique ou RPL). Les résultats, présentés dans la figure 2.2, montrent que la durée de vie est assez homogène entre le routeur et la feuille, quel que soit la couche routage, pour une couche MAC donnée. En effet, lorsque le routeur est à cours d'énergie, le nœud feuille tente de se reconnecter en envoyant des messages de contrôle, et cela jusqu'à épuisement de sa batterie. Par contre, ContikiMAC offre de bien meilleures performances par rapport à 802.15.4. En effet, la durée de vie du nœud routeur (19 jours) atteint presque la durée de vie de l'unique nœud du scénario précédent (21 jours). Contrairement au nœud 802.15.4 qui nécessite beaucoup plus d'énergie pour permettre de fonctionner en mode routeur.

5 Conclusion

Nous proposons un module pour le simulateur WSNET permettant d'estimer efficacement la durée de vie des nœuds dans un réseaux de capteur sans fil avec un faible coût computationnel. Nous montrons que les résultats obtenus ont une précision pouvant aller jusqu'à 94%, contrairement au module par défaut qui surestime de plus de 2600% la durée de vie. Pour finir nous avons retrouvé des résultats connus concernant l'efficacité énergétique de X-MAC, ContikiMAC et 802.15.4 ainsi que de nouveaux résultats précis concernant l'impact de la topologie sur la durée de vie des nœuds.

Références

- [DD09] I. Dietrich and F. Dressler. On the lifetime of wireless sensor networks. *ACM Transaction on Sensor Network*, pages 1–38, 2009.
- [DDV⁺14] W. Dron, S. Duquennoy, T. Voigt, K. Hachicha, and P. Garda. An emulation-based method for lifetime estimation of wireless sensor networks. In *Distributed Computing in Sensor Systems (DCOSS), 2014 IEEE International Conference on*, pages 241–248, 2014.
- [MB14] Mathieu Michel and Quoitin Bruno. Analyse des raisons de l'efficacité de ContikiMAC. In *ALGOTEL 2014 – 16èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications*, pages 1–4, Le Bois-Plage-en-Ré, France, June 2014.