Validation formelle d'un mécanisme de synchronisation pour réseaux sans fil

Jackson Francomme, Karen Godary-Dejean, Thierry Val

Validation formelle d'un mécanisme de synchronisation pour réseaux sans fil

Jackson Francomme* — **Karen Godary**** — **Thierry Val***

* *Université de Toulouse ; UTM ; LATTIS EA4155 (LAboratoire Toulousain de Technologie et d'Ingénierie des Systèmes) ; IUT Blagnac, 1 Place Georges Brassens ; F-31 703 Blagnac, France*
{francomme,val}@iut-blagnac.fr

** *Université de Montpellier 2 ; LIRMM, 161 rue Ada, 34392 Montpellier Cedex 5*
karen.godary@lirmm.fr

RESUME. *Le développement des réseaux et notamment les réseaux de capteurs incite les industries à considérer pour leurs systèmes de communication des alternatives amenant une réduction des coûts et de la complexité tout en garantissant la fiabilité. Ce papier décrit la validation formelle par réseaux de Petri temporels et model checking d'un nouveau protocole de synchronisation avec qualité de service pour réseau maillé de capteurs sans fil utilisant le standard de communication IEEE 802.15.4/ZigBee. L'utilisation des méthodes formelles dans le cadre des réseaux sans fil est assez récente et les résultats obtenus dans cet article prouvent que ces méthodes sont intéressantes dans ce contexte.*

ABSTRACT. *The development of networks and more particularly of sensor networks is an incentive for industries to consider alternatives leading to cost and complexity reduction while ensuring reliability. This paper deals with the formal validation with Time Petri Net and model checking of synchronization protocol with quality of service for wireless mesh networks using the IEEE 802.15.4/ZigBee standard. Formal methods are not currently used in the wireless domain, but this paper's results show that they are an interesting solution for validation.*

MOTS-CLES : *Validation formelle, Réseaux de Petri temporels, Model checking, Analyse exhaustive, IEEE 802.15.4, ZigBee, Réseau de capteurs sans fil, Protocole de communication.*

KEY WORDS: *Formal validation, Time Petri Nets, Model checking, IEEE 802.15.4, ZigBee, Wireless sensor network, Communication protocol.*

1. Introduction

Nous assistons ces dernières années à l'essor des systèmes de communication sans fil dans les applications industrielles. Cela nécessite notamment de développer de nouveaux protocoles et de nouvelles techniques de validation, notamment pour les problèmes de passage à l'échelle et de qualité de service. Aujourd'hui, les concepts liés aux réseaux sans fil sont difficiles à représenter en langage formel et les méthodes de validation formelle ne sont pas toujours adaptées. Cependant, nous montrons dans ce papier que certains concepts protocolaire des réseaux sans fil peuvent être représentés dans le formalisme des *réseaux de Petri temporels* et que leurs propriétés principales sont validables. Nous illustrons notre démarche de validation formelle par une méthode de *model checking* appliquée sur un nouveau protocole de synchronisation pour réseaux sans fil.

2. La validation formelle pour les protocoles de communication sans fil

La validation de protocoles de communication utilise depuis longtemps plusieurs méthodes complémentaires : simulation, test et expérimentation sur prototypes, et méthodes formelles. Ces dernières offrent un complément précieux en assurant une plus grande confiance dans les résultats et en offrant des possibilités de validation impossible en simulation ou sur prototype. En effet, les expérimentations sur prototypes sont limitées par le matériel nécessaire et les conditions pouvant être reproduites. La simulation permet l'exécution de gros systèmes et l'introduction de conditions simulées (surcharge d'un lien réseau, etc.), mais elle ne permet pas l'étude exhaustive de toutes les exécutions du système : il est alors impossible d'être certain qu'une propriété sera toujours vérifiée. Or dans certains contextes, comme les systèmes critiques, cette notion de confiance dans la validation est primordiale. Les méthodes formelles sont donc indispensables pour compléter la validation de nombreux protocoles. En particulier, le *model checking* qui permet la vérification de propriétés exprimées en logique temporelle sur l'ensemble des états accessibles d'un modèle du système est très utilisé dans le domaine de la validation des systèmes temps réel communicants.

Cependant, il n'existe à ce jour que peu de travaux traitant de la validation formelle dans le domaine des réseaux sans fil. Les techniques utilisées sont plus souvent les simulateurs (NS2, OPNET, OMNET++, Truetime), le test sur prototype ou des approches probabilistes qui ne permettent pas d'obtenir des informations sur le pire cas d'exécution. Cependant, les solutions protocolaires proposées dans les réseaux sans fil sont de complexité croissante, et les domaines d'applications émergeants de ces réseaux impliquent des contraintes nouvelles en termes de temps réel et de fiabilité. Les processus de validation doivent ainsi être plus rigoureux, plus complet, voir exhaustifs. Les techniques de validation formelle peuvent donc ici aussi se révéler nécessaires. La modélisation dans un langage formel est en elle-même une étape utile pour la détection des erreurs dans le processus de conception. Par exemple, les réseaux de Petri colorés (CPN, *Coloured Petri Nets*) sont de plus en plus utilisés dans le domaine des réseaux sans fil. Ainsi dans [XIO 02] et [KRI 04], les auteurs ont utilisé les possibilités de ce formalisme afin de modéliser, simuler et/ou analyser des protocoles spécifiques aux réseaux sans fil. Mais si on considère des systèmes temps réel avec contraintes temporelles, les travaux dans le domaine de la validation dans les réseaux sans fil sont rares. Récemment dans [WIB 04] ou [GOD 06], UPPAAL a été utilisé pour modéliser et valider des protocoles de routage et de sécurité pour réseaux ad-hoc mobiles. Ces travaux ont montré que les concepts de base du sans fil (*voisinage*, *broadcast*,..) peuvent être modélisés formellement en automates temporisés. Les

auteurs ont cependant été confrontés à la limitation classique du *model checking* : l'explosion combinatoire de l'espace d'états. Il serait ainsi nécessaire d'approfondir l'étude de l'utilisation de cette méthode d'analyse pour les réseaux sans fil, afin de maîtriser ou contourner ce problème et d'obtenir des résultats exploitables pour des systèmes de tailles et de complexité réalistes. Dans ce contexte, nous proposons dans cet article la validation formelle d'un nouveau mécanisme de synchronisation pour un réseau maillé sans fil basé sur *IEEE 802.15.4*.

3. Protocoles de synchronisation dans les réseaux 802.15.4

L'objectif de ce papier portant sur notre démarche de validation formelle, nous présenterons ici les aspects nécessaires à la compréhension du mécanisme de synchronisation servant de support à la validation. Nous renvoyons le lecteur à la bibliographie et notamment à la thèse de J. FRANCOMME [FRA 08] pour les aspects détaillés du protocole global associé à ce réseau de capteurs sans fil.

Nous considérons un exemple simple de topologie d'un réseau de capteurs en arbre de cellules (figure 1). Chaque nœud capteur ou nœud enfant (N_i) est associé à un nœud parent coordinateur (R_j), ce qui forme une cellule ; le réseau possède un superviseur de réseau : le CPAN (coordinateur de réseau PAN). Dans cette topologie en arbre de cellules, un coordinateur de cellule peut également appartenir à une autre cellule en tant que noeud-fils. Le réseau est ainsi constitué de plusieurs *associations parent-fils* de coordinateurs sur une certaine profondeur de l'arbre. Par exemple, R_2 est le coordinateur parent de R_5, mais aussi le fils du coordinateur du réseau PAN.

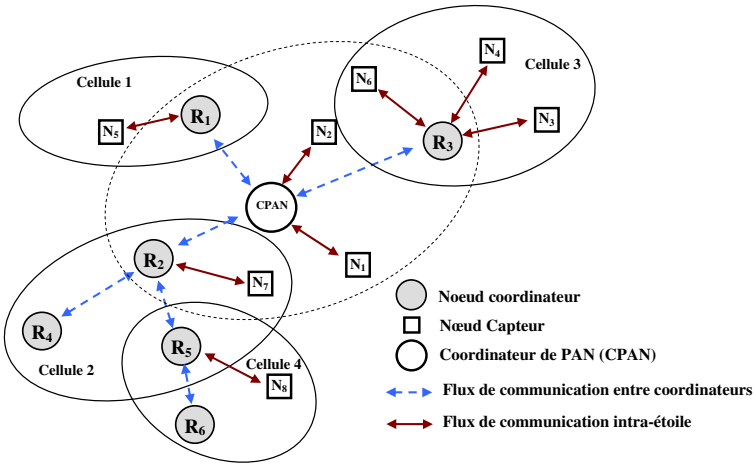

Figure 1. *Modèle de réseau en arbre de cellules*

La synchronisation dans un réseau maillé doit permettre l'utilisation d'une réference temporelle, mais également l'évitement de collisions de trames. En effet, les communications du réseau s'effectuent sur le même canal, et la proximité des cellules permet d'envisager que des trames émises sans précaution perturbent les communications de certaines cellules voisines à portée radio en créant des collisions. Cela peut amener certains nœuds à perdre la synchronisation avec leur tête de cel-

lule, à perdre des données voir à perturber complètement les communications. La synchronisation est donc une fonctionnalité essentielle des architectures de réseaux maillés. Nous résumons dans le paragraphe suivant le standard de communication *IEEE 802.15.4/Zigbee* sur lequel est basée notre nouveau protocole de synchronisation.

3.1. *Généralités sur le standard IEEE 802.15.4/ZigBee*

ZigBee [ALL 06] est une norme de transmission de données sans fil permettant la communication de *machine à machine*. *ZigBee* utilise les couches MAC et PHY du standard *IEEE 802.15.4* [COM 06]. Les débits autorisés sont relativement faibles, entre *20 et 250 Kbits/s*. *ZigBee* fonctionne principalement sur la bande de fréquences du *2,4 GHz* et sur *16 canaux*. Sa portée est de plusieurs dizaines de mètres. Un réseau *ZigBee* peut contenir jusqu'à *254 nœuds* par cellule en plus d'un nœud qui en assure la gestion, nommé *coordinateur* [VAL 08]. Le standard *IEEE 802.15.4* supporte deux modes de fonctionnement gérés par le coordinateur de réseau : *le mode non balisé* et *le mode balisé*. Nous ne considérons que ce dernier qui permettra d'organiser et de synchroniser toutes les communications dans un réseau maillé au sein d'une supertrame. Celle-ci n'est utilisable que dans un réseau utilisant une topologie étoile. La supertrame, émise régulièrement par le coordinateur de réseau, est composée d'une section active, pendant laquelle le coordinateur interagit avec ses nœuds enfants, et d'une section inactive où tous les nœuds entrent dans un mode de basse consommation. La supertrame est découpée en *16 slots*, que nous considèrerons de taille fixe pour rester dans un cas général du standard. La trame balise qui débute la supertrame est émise dans le $slot_0$.

3.2. *Techniques d'évitement de collision dans un réseau maillé*

Il n'existe à ce jour aucun mécanisme d'évitement de collision de balises dans le standard *IEEE 802.15.4* pour des réseaux utilisant le même canal de transmission. Le groupe de travail *IEEE TG4b* [COM 07] a identifié deux types de collisions : les *collisions directes* entre trames balise, et les *collisions indirectes*. Les *collisions directes* de trames balise se produisent lorsqu'au moins deux coordinateurs se trouvent à portée radio l'un de l'autre (R_2 et R_4 fig. 1) , et émettent leurs trames balise au même moment. Les *collisions indirectes* de trames balise se produisent lorsqu'au moins deux coordinateurs qui ne s'entendent pas directement mais ont un voisin commun, émettent leur balise approximativement en même temps (R_1 et R_3 fig. 1). Ce problème est plus complexe que celui des collisions directes de trames balise ; en effet, cela ne concerne plus seulement les coordinateurs voisins à *1 saut*, mais également les coordinateurs qui sont à *2 sauts*.

Parmi les propositions du groupe de travail *TG4b* pour la gestion des collisions, nous avons retenu la proposition [LEE 04, SHA 04] avec *section exclusive de balises*. Celle-ci est basée sur une extension de la structure initiale de la supertrame du mode balisé du standard *IEEE 802.15.4*. Une fenêtre temporelle à accès sans contention est réservée en début de chaque supertrame pour l'émission de l'ensemble des balises des coordinateurs. Cette fenêtre temporelle, appelée *section exclusive de balise*, se trouve dans le $slot_0$ de la supertrame. Chaque coordinateur se voit attribuer un slot de cette section de temps (un CFTS, *Contention Free Time Slot*) pour transmettre sa balise de synchronisation, évitant ainsi toutes collisions de trames balise entre coordinateurs voisins. Cette méthode nécessite une démarche d'assignation des CFTS à chaque coordinateur pour éviter la collision des

trames balise d'une part, mais également pour permettre la synchronisation hiérarchique de tout le réseau. Cependant, nous pouvons aisément voir que cette technique ne fonctionne que lorsqu'aucun évènement ne vient rompre la chaîne de propagation de la synchronisation depuis le superviseur de synchronisation jusqu'aux nœuds les plus profonds de l'arbre. Dans le cas contraire, on se retrouverait avec un réseau inopérant avec une partie des communications qui ne seraient plus synchronisées. Or, les causes d'interruptions temporaires ou permanentes de la synchronisation peuvent être nombreuses dans notre contexte : erreur de transmission, panne d'un coordinateur, mobilité d'un nœud, etc. Il serait donc nécessaire, dans un contexte de garantie minimum de qualité de service, d'implémenter un protocole de synchronisation permettant une tolérance à ces interruptions et garantissant une synchronisation de bout-en-bout du réseau même en présence d'une faute.

3.3. *Notre protocole de synchronisation avec QdS*

Le protocole de synchronisation que nous proposons et que nous avons validé est illustré sur la figure 2. Ce protocole améliore la QoS par rapport à la synchronisation de base du standard par la synchronisation par *section exclusive de balises* avec un mécanisme que nous appelons *contrôle de propagation*, basé sur la notion de demi-périodes proposée dans [GOD 07]. Nous avons adapté ce principe au contexte de communication sans fil en considérant les notions de puissance d'émission du standard et de voisinage. Le *contrôle de propagation* permettra de gérer le remplacement d'un coordinateur défaillant dans la chaîne de synchronisation. Le mécanisme de *contrôle de propagation* est basé sur le découpage du CFTS en deux demi-périodes comme cela est illustré sur la figure 2, par exemple entre t_1 et t_2 :

• la première demi-période $DP1$ est utilisée par le nœud coordinateur associé à ce CFTS pour y transmettre sa balise (*b*) destinée à ses descendants à *1 saut*. Elle contient l'ensemble des informations de synchronisation, en particulier l'ordonnancement des différents CFTS.

• la seconde demi-période $DP2$ est utilisée par le nœud coordinateur parent afin de remplacer l'un de ses enfants (donc à *1 saut*) si celui-ci ne peut assurer la transmission de sa balise à ses propres enfants (et donc à *2 sauts* de son parent), ou si cette trame balise a été perdue.

Lorsqu'un coordinateur remplace son fils défaillant, il augmente sa puissance d'émission afin de pouvoir atteindre les nœuds appartenant à son voisinage à *2 sauts*. Cela ne pose pas de problème dans notre cas étant donné que nous considérons un seul coordinateur associé à chaque CFTS : il n'y a donc pas de collision possible. Le scénario illustré sur la figure 2 représente le fonctionnement du protocole de synchronisation lorsque le coordinateur R_5 n'assure plus sa tâche de relais pour la synchronisation. Nous décrivons les différentes étapes de ce scénario :

• à t_0 débute une nouvelle supertrame avec l'émission de la balise du superviseur de synchronisation du réseau dans la première demi-période. La deuxième demi-période reste vide.

• à t_1 se termine le premier CFTS et commence le deuxième. R_2 transmet sa balise dans la première demi-période, et ses voisins dont CPAN, reçoivent cette trame. La deuxième demi-période de ce CFTS n'est donc pas utilisée car tout va bien.

• à t_2 se termine le deuxième CFTS et commence le troisième. R_2 est en attente de la balise de son fils R_5 durant $DP1$, mais la première demi-période n'est pas utilisée par R_5 qui est défaillant. Cela est détecté par R_2 qui ne reçoit pas la trame attendue, il adapte alors sa puissance d'émission

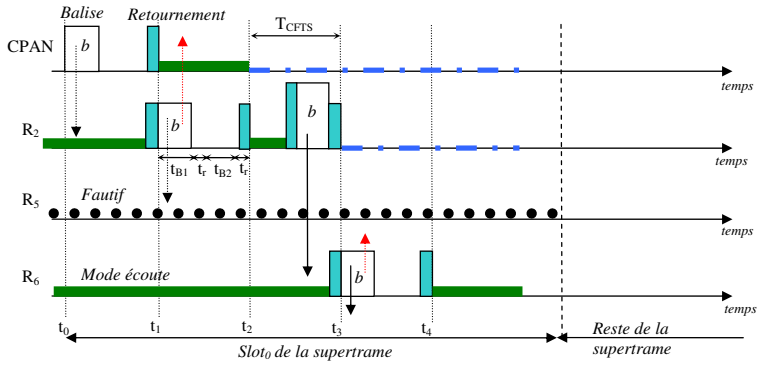

Figure 2. *Synchronisation avec erreur de R_5*

pour pouvoir atteindre les nœuds attendant la trame de synchronisation manquante, en l'occurrence ici le coordinateur R_6. R_2 émet une trame balise à forte puissance dans le deuxième demi-intervalle de ce CFTS. R_6 reçoit la trame balise de R_2 avec les informations de synchronisation.

• à t_3 se termine le troisième CFTS, et le mécanisme de synchronisation continue normalement.

Outre le bon fonctionnement du protocole, nous devrons montrer que la durée de propagation des informations de synchronisation dans tout le réseau respecte la relation $T_{Tot_Sync} \leq N * T_{CFTS}$ avec N le nombre de coordinateurs dans le réseau (voir [FRA 08] pour le détail de la durée théorique), et ceci malgré un défaut de transmission d'un coordinateur.

Nous avons présenté un protocole de synchronisation des balises permettant la synchronisation hiérarchique des coordinateurs du réseau en évitant les collisions, et introduisant de la qualité de service grâce à la compensation d'une faute d'un coordinateur. Cela permet ainsi d'envisager l'utilisation du mode balisé du standard *IEEE 802.15.4* dans un réseau maillée dans un contexte industriel. Nous décrivons dans la section suivante, la méthodologie de validation formelle puis la démarche de modélisation des différentes fonctionnalités de ce protocole ainsi que les résultats de sa validation.

4. Méthodologie de validation

4.1. *Validation : Méthodes et Outils*

Comme nous l'avons abordé au §2, le *model checking* est une méthode formelle permettant la validation de systèmes de façon exhaustive en vérifiant des propriétés sur l'ensemble des états accessibles d'un modèle du système. Ainsi, il est possible de vérifier par exemple qu'une situation indésirable ne sera jamais atteinte, ou qu'une propriété essentielle sera toujours respectée. Dans un contexte de système contraint temporellement, comme les systèmes temps réel de type industriel, le *model checking* peut de plus s'appliquer à des formalismes de modélisation temporisés, qui permettent l'expression et la vérification de propriétés temporelles quantitatives. Nous avons ainsi choisi d'utiliser le formalisme des *réseaux de Petri temporels* (RdPT) [MER 74]. Les *réseaux de Petri* sont en effet un formalisme bien adapté à l'expression des systèmes distribués, et sont associés

à des techniques mathématiques permettant la validation formelle. Les *réseaux de Petri temporels* permettent par extension la représentation du temps quantitatif sous la forme d'intervalles de tir sur les transitions. L'ensemble des états du modèle RdPT est calculé par la méthode du graphe des classes d'états [BER 91] implémentée dans l'outil TINA [1] [BER 03]. Cet outil permet la génération du graphe de l'ensemble des états accessibles. Les propriétés ont ensuite été vérifiées sur ce graphe par l'intermédiaire de l'outil LPT [2] [GOD 08], qui permet la vérification de propriété d'accessibilité, de marquage et de bornes temporelles sur un graphe généré par TINA.

4.2. *Propriétés à vérifier*

Notre proposition de synchronisation garantit la propagation de la synchronisation sans collision des trames balise, et ceci même lors d'une faute d'un des coordinateurs. Ainsi, le processus de validation devra vérifier tout d'abord le bon fonctionnement de l'accès au médium et l'absence de collision. L'accès au médium doit être vérifié d'un point de vue logique, c'est-à-dire l'ordre dans lequel les coordinateurs émettent, mais également d'un point de vue temporel afin de vérifier le pire temps de synchronisation. Nous avons vérifié trois propriétés principales, d'abord en comportement normal puis en présence de fautes :

Prop1 : vérification de l'absence de collision de balise ;

Prop2 : vérification de l'ordre d'émission des coordinateurs (chacun dans son propre CFTS) ;

Prop3 : vérification de la borne temporelle de la synchronisation, pour chacun des coordinateurs.

4.3. *Hypothèses de fautes*

Un processus de validation d'un protocole assurant de la tolérance aux fautes se doit de préciser les hypothèses de fautes : sous quelles conditions la validation, et donc le bon fonctionnement du protocole, sont garantis. Le protocole de synchronisation considéré peut être soumis à de nombreuses fautes, comme par exemple une panne permanente ou transitoire d'un coordinateur ou une perte ou altération d'une trame balise. Ces fautes peuvent avoir des répercutions différentes ; une panne permanente peut par exemple entraîner la nécessité d'une modification de la topologie de l'arbre de cellule. Cependant, la réaction primaire du mécanisme de synchronisation à l'une de ces fautes est la même dans tous les cas : le coordinateur parent du coordinateur fautif émet la balise à sa place. Ce comportement est assuré par le mécanisme de *contrôle de propagation*, qui est garanti avec l'hypothèse que le CPAN n'est jamais fautif (il n'a pas de parent pour compenser la perte de sa balise), et que deux fautes ne peuvent se produire simultanément. Cela se traduit dans notre cas par l'assurance que la trame balise de "*récupération*", émise par le parent du coordinateur fautif immédiatement après la détection de la faute, ne sera pas perdue. Ces hypothèses étant précisées, nous pouvons passer à la phase de validation proprement dite, en commençant ci-dessous par la modélisation du système.

1. www.laas.fr/tina/
2. www.lirmm.fr/ godary/SOFT/Logiciels.html

5. Modélisation

Cette section présente des extraits du modèle du système. Le modèle global est composé de plusieurs sous-modèles interagissant ensemble : le superviseur (CPAN) (non présenté ici car très proche du modèle des coordinateurs), les coordinateurs, le modèle du médium et bien entendu le modèle des propriétés. Tous ces modèles n'ont pu être présentés faute de place : voir [FRA 08].

5.1. *Modéle d'un coordinateur*

Le modèle complet d'un coordinateur de synchronisation est présenté sur la figure 3, pour le coordinateur $R2$. Cette figure étant complexe, nous allons la décrire en plusieurs parties : la partie cyclique, à gauche, qui représente l'état global du coordinateur ; la partie cyclique inférieure à droite qui représente l'attente de l'accès au médium, la partie centrale (à droite) qui représente la réception d'une trame et enfin la partie supérieure (à droite) qui représente le contrôle de propagation.

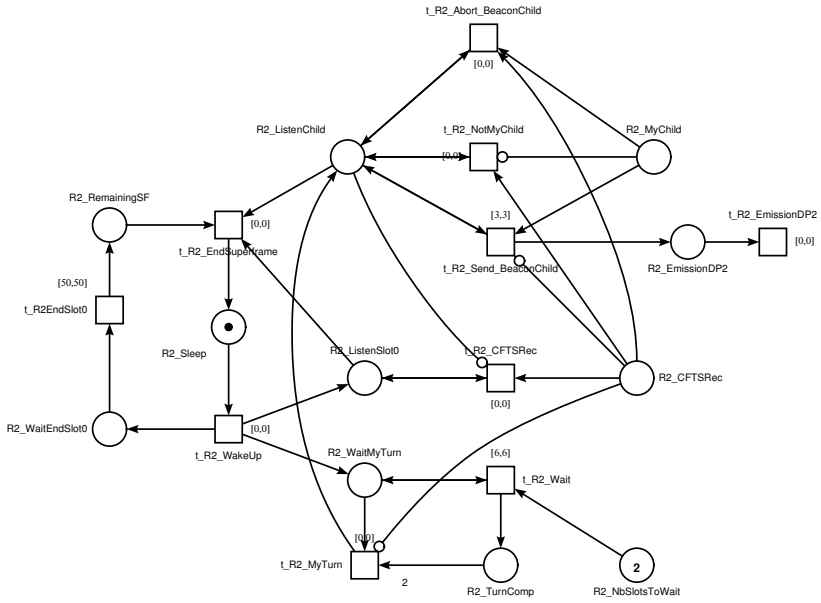

Figure 3. *Modèle d'un coordinateur*

Partie "état global du coordinateur" : Le coordinateur est soit en mode basse consommation (place $R2\_Sleep$), soit en phase de synchronisation (attente fin $slot_0$ en $R2\_WaitEndSlot0$), soit dans la section active de la supertrame (en $R2\_RemainingSF$). Le tir de t_R2_WakeUp représente le début du $slot0$ et donc du mécanisme de synchronisation : lancement des tâches de réception (marquage de $R2\_ListenSlot0$) et d'accès au médium (marquage de $R2\_WaitMyTurn$).

Partie "accès au medium" : L'ordonnancement des CFTS et donc le rang du CFTS du coordinateur est représenté par le marquage de la place $R2\_NbSlotsToWait$. Pour chaque coordina-

teur, cette place contient autant de jetons que le nombre de CFTS à attendre avant l'émission de sa propre balise. La place $R2\_TurnComp$ contiendra autant de jetons que le coordinateur $R2$ a déjà attendu de CFTS. La durée du CFTS est représentée par le délai associé à la transition t_R2_Wait. Lors de l'attente de l'accès au médium (place $R2\_WaitMyTurn$ marquée), le temps associé à la transition t_R2_Wait s'écoule. A la fin de ce délai, deux comportements sont possibles : soit $R2$ doit encore attendre, la transition t_R2_Wait est tirée et le nombre de jetons dans les places $R2\_NbSlotsToWait$ et $R2\_TurnComp$ change, mais le coordinateur reste dans la place $Rx\_WaitMyTurn$; soit il n'y a plus de CFTS à attendre, la transition t_Rx_MyTurn est tirée et le coordinateur accède au médium et transmet sa balise.

Partie "réception" : La réception d'une trame par le nœud $R2$ est représentée par le marquage de la place $R2\_CFTSRec$ (ce jeton provient du modèle du médium) et par le tir de la transition $t\_R2\_CFTSRec$. L'arc inhibiteur entre $R2\_CFTSRec$ et la transition t_R2_MyTurn modélise l'impossibilité de notre technologie sans fil à émettre et recevoir au même moment.

Partie "contrôle de propagation" : Cette partie permet la détection de la non-émission de la balise de l'un de ses fils. La détection commence après l'émission de sa propre balise (marquage de $R2\_ListenChild$ lors du tir de t_R2_MyTurn). Elle est basée sur la connaissance de sa descendance, représentée par $R2\_MyChild$: si cette place est marquée, le CFTS en cours est celui de l'un des fils du coordinateur concerné. Dans ce cas, le coordinateur écoute le médium pendant la première demi-période. Si une trame est reçue (marquage de $R2\_CFTSRec$), le coordinateur arrête d'observer le médium (consommation du jeton de $R2\_MyChild$ par le tir de $t\_R2\_Abort\_BeaconChild$). Si au contraire la trame est perdue, la date de tir maximale de $t\_R2\_Send\_BeaconChild$ est atteinte et le coordinateur $R2$ doit émettre pour son fils : tir de cette transition, marquage de $R2\_EmissionDP2$ et émission sur le médium pendant la deuxième demi-période par la transition $t\_R2\_EmissionDP2$. Dans le cas où le CFTS observé ne correspond pas à celui d'un fils de $R2$, la réception d'une trame d'un autre coordinateur entrainera le tir de $t\_R2\_NotMyChild$.

5.2. *Modèle du médium et modèle de fautes*

Dans le contexte des réseaux sans fil, la problématique de la représentation du médium doit être considérée. Il est nécessaire de trouver une modélisation simple et exploitable du point de vue validation, qui représente de façon réaliste les contraintes réelles et qui assure une vérification fiable des propriétés. Dans notre contexte, l'accès au médium durant la synchronisation est basé sur un concept TDMA, ce qui n'implique pas de représentation de type probabiliste. De plus, pour le mécanisme de contrôle de propagation, la seule information nécessaire est la réception ou non d'une trame correcte. Enfin, la représentation des zones d'émission n'est pas indispensable ici : l'ordonnancement des CFTS et l'accès TDMA garantissent qu'une seule trame est émise à la fois. Il est donc possible de représenter le médium de façon globale comme une seule zone radio. Cette simplification facilite de plus la vérification de la propriété de détection des collisions. Le modèle du médium devra également représenter le contrôle de propagation et l'occurrence de fautes. Les fautes sont modélisées sur le principe de la non-réception de la trame balise. Nous avons supposé (§4.3) que deux fautes ne se produisent jamais dans deux demi-périodes consécutives. Ainsi, une distinction entre les deux demi-périodes d'un CFTS est nécessaire dans ce modèle, afin de représenter la possibilité de l'occurrence d'une faute durant la première demi-période, mais pas durant la seconde.

5.3. *Modélisation des propriétés à valider*

Certaines propriétés, comme les propriétés de marquage ou d'accessibilité, peuvent être vérifiées directement sur le graphe d'analyse du modèle en parcourant l'ensemble des états possibles et en vérifiant leur marquage. C'est le cas de la propriété *Prop1* (§4.2) que l'on désire vérifier : vérification du marquage des places représentant le médium de communication. Cependant, certaines propriétés, telles que les propriétés temporelles quantitatives, sont plus complexes et ne peuvent être exprimées et vérifiées directement par des techniques d'analyse simples et efficaces. Il est ainsi nécessaire de les modéliser explicitement afin de transformer la propriété initiale en une propriété plus simple. Les propriétés Prop2 et Prop3 ont ainsi été modélisées et ajoutées au modèle global. Prop2 est une simple modélisation de l'ordre d'émission désiré des coordinateurs. Intégrée au modèle globale, cette propriété provoque un blocage si l'ordre d'accès au médium n'est pas celui désiré.

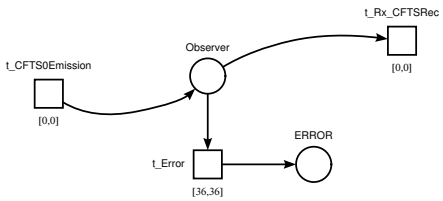

Figure 4. *Modèle de la propriété de la borne temporelle de synchronisation*

Par contre, la figure 4 modélise *Prop3* par un observateur qui transforme la vérification de la borne temporelle de synchronisation en un problème d'accessibilité d'un état erreur. La vérification de cette borne pour le coordinateur Rx est la vérification du pire temps d'exécution entre l'émission de la première trame balise par le CPAN dans le *CFTS0* (tir de la transition $t\_CFTS0Emission$ et marquage de la place $Observer$) et la réception par Rx de la trame balise le concernant (tir de la transition $t\_Rx\_CFTSRec$ qui consomme le jeton de la place $Observer$), provenant de son coordinateur parent ou du parent de celui-ci si problème. Si la durée entre ces deux transitions est supérieure à la durée associée au tir de la transition t_Error, alors cette dernière est tirée et la place $ERROR$ est atteinte. Le modèle présenté ici est une version simplifiée de celui utilisé pour la validation de notre protocole. Le modèle réel, plus complexe, n'est pas présenté ici.

6. Validation du protocole de synchronisation : résultats et analyse

Cette section présente les résultats de l'analyse de la borne temporelle du mécanisme de synchronisation, fournis par les logiciels LPT et Tina. Cette analyse utilise l'observateur de la figure 4, qui permet de vérifier le pire temps écoulé entre l'émission d'une balise par le CPAN et la réception des informations de synchronisation par le coordinateur pour lequel la borne est vérifiée. Ce modèle permet de vérifier que le temps écoulé est inférieur à une borne temporelle fixée : ici 36 unités de temps. Afin d'obtenir la valeur maximale de cette borne, sans en fixer la valeur a priori, le logiciel LPT effectue une recherche de cette borne par dichotomie sur la valeur associée à la transition t_ERROR. LPT modifie le modèle RdP pour fixer la valeur de cette transition, le logiciel Tina construit le graphe d'analyse de ce modèle, puis LPT effectue un parcours de ce graphe afin

de vérifier l'accessibilité de l'état $ERROR$. Ensuite, la valeur associée à la transition t_ERROR est modifiée et l'analyse est de nouveau effectuée sur un nouveau modèle avec une valeur de borne différente. L'analyse LPT se termine lorsque la valeur maximale de la borne temporelle recherchée à été trouvée (ou lorsque la borne n'a pas été trouvée). Cf. [GOD 08] pour plus de détails sur LPT.

Les résultats de la figure 5 représentent le délai maximal de propagation de la synchronisation donné en fonction du rang hiérarchique du coordinateur considéré (résultats obtenus avec un CFTS = 6 unités de temps). Ils montrent que le délai maximal de synchronisation jusqu'à un nœud terminal est proportionnel au rang du CFTS. Nous pouvons ainsi valider l'équation de l'analyse théorique $T_{Tot\_Sync} = N \times T_{CFTS}$. L'infléchissement du début de la courbe s'explique par le fait que le CPAN émet par hypothèse sa trame dans la première demi-période sans erreur possible. Le délai de synchronisation est donc simplement égal au temps de propagation d'une trame balise (2 unités de temps). Pour les autres coordinateurs, par contre, le pire cas de délai d'émission d'une balise dans leur CFTS correspond au moment où le parent émet cette trame, dans la deuxième demi-période, lorsqu'il y a eu une faute. Un exemple d'un tel scénario a été donné figure 2, pour le coordinateur R_6. Dans ce cas, le délai entre le début de la synchronisation et l'émission de la balise dans le CFTS associé à R_6 est de 17 unités de temps. L'équation exacte de nos résultats est $T_{Tot\_Sync} = N \times T_{CFTS} - t_r$ (avec t_r : temps de commutation entre le mode récepteur et le mode émetteur).

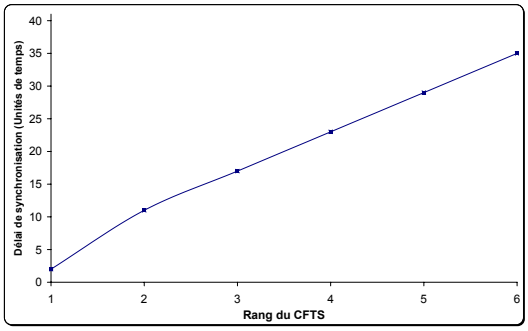

Figure 5. *Résultats de validation : bornes du délai de synchronisation*

7. Conclusion

Dans cet article, nous avons montré que le *model checking*, méthode de validation formelle, est utile pour la validation d'un protocole de synchronisation pour réseaux sans fil. Malgré le problème connu de cette méthode, l'explosion combinatoire, qui limite la taille des systèmes validés, nous avons pu vérifier les principes de base de notre protocole et cela pour n'importe quel rang de CFTS. Les spécificités des réseaux sans fil ont été intégrées dans notre modèle du système, et donc prises en considération lors de sa validation. De plus, nous avons dans cet article effectué la vérification d'un type spéciale de propriété : une propriété temporelle paramétrée. En généralisant ce type d'analyse à des propriétés paramétrées plus générales, la validation formelle peut devenir un outil précieux dans le cadre des réseaux sans fil comportant souvent des paramètres de configuration complexes.

8. Bibliographie

[ALL 06] ALLIANCE Z., " ZigBee Specification : ZigBee Standards Organization ", *Available from http : //www.zigbee.org/en/spec_download/download_request.asp*, , 2006.

[BER 91] BERTHOMIEU B., DIAZ M., " Modeling and verification of time dependent systems using time Petri nets ", *IEEE Transactions on Software Engineering*, vol. 17, n° 3, 1991.

[BER 03] BERTHOMIEU B., RIBET P.-O., VERNADAT F., " The tool TINA - Construction of Abstract State Spaces for Petri Nets and Time Petri Nets ", *International Journal of Production Research*, vol. 46, n° 10, October 2003, page 22.

[COM 06] COMPUTER-SOCIETY I., " Std $802.15.4^{TM}$-2003, Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs) ", *IEEE Std* $802.15.4^{TM}$*-2006 (Revision of IEEE Std 802.15.4-2003)*, , 2006.

[COM 07] COMPUTER-SOCIETY I., " $WPAN^{TM}TG4b$ ", *http ://www.ieee802.org/15*, , 2007.

[FRA 08] FRANCOMME J., " Propositions pour un protocole déterministe de contrôle d'accès et de routage avec économie d'énergie dans les réseaux ZigBee ", Thèse de doctorat, Université de Toulouse, LATTIS EA4155, France, Juin 2008.

[GOD 06] GODSKESEN J. C., GRYN O., " Modelling and verification of security protocols for ad hoc networks using UPPAAL ", *18th Nordic Workshop on Programming Theory (NWPT'06)*, Reykjavík, Iceland, 18-20 October, 2006.

[GOD 07] GODARY K., ANDREU D., SOUQUET G., " Sliding Time Interval based MAC Protocol and its Temporal Validation ", 7^{th} *IFAC International Conference On Fieldbuses & Networks in Industrial Systems (FET'07)*, vol. Toulouse, France, 2007.

[GOD 08] GODARY K., " LPT : Little Parametric Tool, outil pour la validation d'une borne temporelle paramétrée ", *Proc. of the Confrence Internationale Francophone d'Automatique (CIFA'08)*, Bucarest, Roumanie, Sept. 2008.

[KRI 04] KRISTENSEN L. M., JENSEN K., " Specification and Validation of an Edge Router Discovery Protocol for Mobile Ad-hoc Networks ", *Integration of software specification techniques for applications in engineering, LNCS*, vol. 3147, 2004, p. 248-269.

[LEE 04] LEE M., ZHENG J., LIU Y., SHAO H.-R., DAI H., ZHANG J., JEON H., " Combined Beacon Scheduling ", *Proposal to IEEE 802.15.4b*, , Sept. 2004.

[MER 74] MERLIN P., " A Study of the Recoverability of Computing System ", PhD Thesis, Univ. of California, 1974.

[SHA 04] SHAO H.-R., ZHANG J., DAI H., " Enhancements to IEEE 802.15.4 ", *Proposal to IEEE 802.15.4b Task Group*, , July 2004.

[VAL 08] VAL T., BOSSCHE A. V. D., " Développement et analyse multi outils d'un protocole MAC déterministe pour un réseau de capteurs sans fil ", *Colloque francophone sur l'ingénierie des protocoles (CFIP 2008)*, Les Arcs, France, Mars 2008, HERMES.

[WIB 04] WIBLING O., PARROW J., PEARS A., " Automatized Verification of Ad Hoc Routing Protocols ", *24th IFIP WG 6.1 International Conference on Formal Techniques for Networked and Distributed Systems (FORTE 2004)*, Madrid, Spain, 27-30 September 2004, Springer Berlin / Heidelberg, p. 343-358.

[XIO 02] XIONG C., MURATA T., TSAI J., " Modeling and simulation of routing protocol for mobile ad hoc networks using colored petri nets ", *ACM International Conference on Application and theory of petri nets : formal methods in software engineering and defence systems - Volume 12*, Adelaide, Australia, 2002, Australian Computer Society, Inc., p. 145 - 153.