

Systèmes Distribués et Réseaux

Projet

Réseaux de capteurs

Florian Gourmelen
Valentin Guerlesquin

26 janvier 2007

Résumé

Ce document constitue le compte-rendu du travail, des études, et des réflexions menées lors du projet de l'Unité d'Enseignement Systèmes Distribués.

Dans le cas d'un réseau de capteurs autonomes, disséminés aléatoirement sur un champ d'observation, on s'intéresse aux problèmes de routage de l'information et de connexité du réseau. L'application prise en exemple lors de cette étude est la protection d'une large zone inhabitée contre les incendies (typiquement, un grand domaine forestier).

Table des matières

1 Etude préliminaire	2
1.1 Hypothèses de départ	2
1.2 Etude de la connexité	2
1.3 Interprétations des résultats	3
1.4 Conclusions	3
2 Routage et acheminement des messages	3
2.1 Protocoles standards	4
2.2 Contraintes du cas	4
2.3 Protocole naïf	4
2.3.1 Scénario	4
2.3.2 Structure de paquet associée	5
2.3.3 Ce que permet ce protocole	5
2.3.4 Ce que ne permet pas ce protocole	5
3 Simulation du fonctionnement d'un réseau	5
3.1 Code d'un noeud	5
4 Comment localiser le noeud initial ?	6
4.1 Cas d'un protocole standard	6
4.2 Cas du protocole présenté	6

1 Etude préliminaire

On dispose d'un générateur aléatoire de modèles de réseau de type "sans fil", écrit en SmallTalk. Celui ci génère un réseau en "parachutant", au hasard, des modules autonomes ayant une portée de communication fixée. Ce générateur est capable de produire un programme Occam représentant le réseau, constitué de noeuds et de canaux de communication bidirectionnels. Lorsque un noeud est situé dans la zone de couverture d'un autre, alors il existe un canal entre ces deux noeuds.

L'un des premiers problèmes auquel il convient de s'intéresser réside dans la connexité d'un tel réseau. Les réseaux générés par cet outils sont-ils connexes¹ ?

1.1 Hypothèses de départ

On suppose disposées sur le champ d'étude un ensemble de bornes satellites permettant la communication du réseau de capteurs avec un poste d'observation distant, par exemple. Dans un premier temps, on considère que ces bornes ont été installées à intervalles réguliers.

1.2 Etude de la connexité

Pour générer des statistiques sur la connexité des réseaux générés par l'outil donné, on réalise une méthode permettant de générer un nombre conséquent de réseaux de même type (portée, nombre de noeuds) et de vérifier leur connexité. Cette méthode retourne le pourcentage de réseaux connexes générés.

L'usage de cette méthode, avec une variété de paramètres, permet d'obtenir les statistiques de Figure 1. Ces courbe a été obtenue en générant, pour chaque valeur du nombre de capteurs entre zéro et cent, et pour les quatres configurations de passerelles différentes, dix milles réseaux aléatoires.

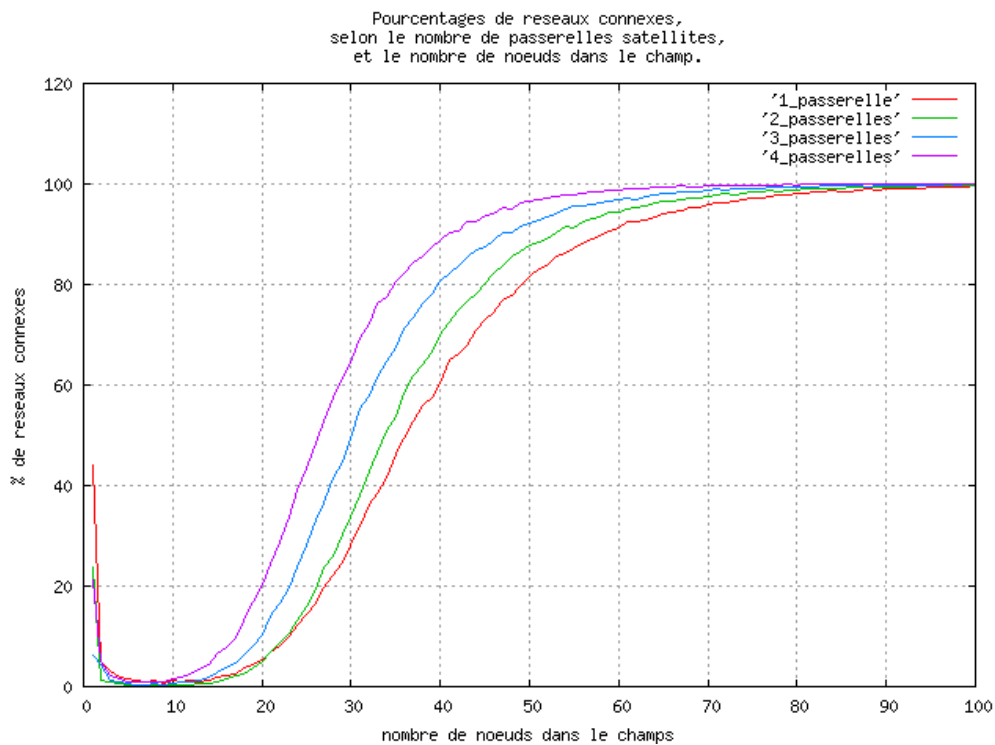


FIG. 1 – Statistiques de connexité d'un réseau

¹On définira comme connexe un réseau dont tout les noeuds sont joignables depuis l'une des passerelles satellites.

Pour la génération de ces statistiques, nous avons utilisés les dispositions suivantes :

4 passerelles Elles sont disposées aux quatre coins du champs d'étude

3 passerelles Elles sont disposées dans les coins du champs d'étude. L'un des coins reste inoccupé.

2 passerelles Disposées dans les coins, elles sont diamétralement opposées

1 passerelle Disposée au centre. Cependant, d'autres essais on montrés qu'une passerelle située dans un coin donnait des statistiques rigoureusement identiques

1.3 Interprétations des résultats

Les premières valeurs (connexité moyenne pour un faible nombre de noeuds), sont justifiée par la disposition des noeuds à proximité de la (des) borne(s). Il est en effet plus difficile d'obtenir un réseau vérifiant la proposition

“Chaque noeuds est suffisamment proche de son voisin, et au moins un voisin (des voisins) d'un noeud est assez proche d'une passerelle.”

que

“Chaque noeuds est suffisamment proche d'une passerelle (et les noeuds sont au nombre de 1, 2, 3 ou 4).”

Quoi qu'il en soit, les premières parties des courbes ne nous intéressent pas.

Les courbes sont des sigmoïdes, dont la pente est proportionnelle au nombre de passerelles.

L'observation faite avec une seule passerelle fait ressortir la conjecture suivante :

“A nombre de noeuds égal, on a autant de chances d'obtenir un réseau hyper-connexe² avec un nombre quelconque de passerelles, que d'obtenir un réseau connexe avec une seule passerelle.”

Cette conjecture a d'ailleurs été vérifiée expérimentalement.

1.4 Conclusions

A partir de ces statistiques (à adapter à la portée réelle des modules), accompagnées

- des tolérances d'erreur (réseau non-connexes),
- des coûts d'implantation (d'un capteur, d'une borne satellite),
- des caractéristiques du protocole mis en place (diagnostics de réseau possibles, ou pas),

il doit être possible de déterminer avec précision le nombre de capteurs à disposer sur une surface donnée.

En particulier, si l'erreur est quasi-inacceptable, et que le protocole ne permet pas de déterminer la topologie (connexité) du réseau, alors il faudra utiliser un grand nombre de capteurs, et le plus possible de passerelles.

Si par contre, le coût d'une passerelle est prohibitif, mais que le protocole utilisé permet un diagnostic aisé du réseau, alors il doit être possible d'utiliser un nombre réduit de passerelles, et de compléter ponctuellement le réseau de capteurs pour le rendre connexes (un agent se rends sur place pour disposer des capteurs là ou l'on sait qu'il n'y en a pas assez).

2 Routage et acheminement des messages

L'un des problèmes majeur pour le choix du protocole de routage réside dans le caractère autonome des capteurs. Il est indispensable que la durée de vie d'un tel équipement soit maximisée, et cela passe par une économie des ressources énergétiques, et donc matérielles.

²Un réseau est hyper-connexe si toutes les passerelles peuvent joindre l'intégralité des noeuds du réseau.

2.1 Protocoles standards

Les trois protocoles appliqués aux réseaux mobiles présentés en travaux dirigés disposent de caractéristiques que nous allons résumer ici.

DSDV Chaque noeud tiens à jour une table de routage contenant le prochain saut, et le nombre de sauts nécessaires pour rejoindre chacune des destinations (ici, les quatres passerelles satellites).

AODV Les routes sont constituées à la demande. Les noeuds n'ont pas besoin de maintenir à jour une table de routage. Les noeuds maintiennent uniquement la liste des noeuds voisins. La création d'une route est réalisée par l'envoi d'un message Route Request, qui est diffusé au travers du réseau, jusqu'à atteindre la destination. Lors du voyage de la réponse au message Route Request, la route vers la destination est mémorisée au sein de chaque noeud intermédiaire. Cette route a une durée de vie temporaire.

DSR Le noeud source doit connaitre avant d'envoyer son paquet la route que celui ci doit emprunter. Les routes connues par un noeud sont mémorisées dans une table de routage interne. Les noeuds peuvent s'échanger des semi-routes, afin de rendre l'établissement d'une route plus rapide.

2.2 Contraintes du cas

Dans notre cas précis, les contraintes liées au matériel et à l'environnement sont les suivantes

- Nécessité absolue de minimiser l'activité d'un noeuds
- Nombre limité de destinataires (les bornes satellites uniquement)
- Nombre limité de messages limités (principalement, les messages d'alerte)
- Sous-réseau(x) potentiellement isolé(s)³
- Localisation du capteur en alerte la plus fine possible
- Possibilité de mort d'un noeud (fin de vie de batterie, ingestion par un sanglier, chasseur émêché...)

2.3 Protocole naïf

2.3.1 Scénario

Le scénario du protocole de communication et de routage implanté au sein des noeuds pourrait être le suivant :

- Ecoute permanente,
- Lors de la détection d'un incendie,
 - vérifier au préalable que personne dans son entourage proche n'émet (principe de Collision Avoidance),
 - puis émission d'un message d'alerte, comportant un ID unique, comportant les diverses informations relevées,
 - écoute simultanée (principe de Collision Detection),
 - si une collision est détectée, ré-émettre le message après un certain délais,
 - régulièrement, émettre le paquet d'alerte (de moins en moins souvent cependant, pour éviter une saturation du réseau), et ce jusqu'à la mort du module (due à l'incendie), ou jusqu'à la fin de l'alerte,
- Lors de la réception d'un message d'alerte, ou de type "clear"
 - Si ce message d'alerte est déjà passé par ici, ne rien faire (opération réalisée en verifiant la liste interne),
 - Sinon, le relayer, en incrémentant au passage le compteur de sauts du message. Ajouter l'ID du message associé au noeud source à une liste interne,
- Lors de la fin d'une alerte (température revenue à la normale), émission d'un message "clear", et ce, un certain nombre de fois, puis retour à la seule écoute.

³L'isolement d'un sous-réseau est une situation critique que l'on ne peut éviter qu'en disposant initialement de suffisamment de capteurs. Cependant, une autre solution réside dans le protocole, permettant ou non de détecter l'isolement d'un réseau. Il est alors possible de déplacer un agent qui ira compléter le réseau de capteurs, afin de le rendre connexe.

2.3.2 Structure de paquet associée

ID source
ID message
NB sauts
DATAS

2.3.3 Ce que permet ce protocole

Tel qu'il est décrit, ce protocole permet de minimiser l'usage des ressources matérielles, et donc d'augmenter la durée de vie des modules. Il permet également d'éviter le bouclage de paquets, et, grâce au nombre de sauts, permet une localisation plus ou moins fine du noeud source.

2.3.4 Ce que ne permet pas ce protocole

Ce protocole ne permet pas de diagnostics réseau. Il n'est pas possible, de diagnostiquer un partitionnement du réseau. Il n'est pas non plus possible d'établir une topographie du réseau.

3 Simulation du fonctionnement d'un réseau

Le réseau est modélisé en Occam, en utilisant le générateur proposé.

On considère qu'il y a brouillage si un même noeud reçoit plusieurs messages non-nuls, lors du même cycle, sur deux canaux de communication différents.

On réalise le codage du protocole naïf en Occam.

3.1 Code d'un noeud

Voici la procédure d'un noeud standard. Il n'y a pas de génération aléatoire d'incidents. Pour cela, on utilise une procédure particulière.

```
PROC Node( []CHAN OF someNetwork.proto in , []CHAN OF someNetwork.proto out, VAL INT id)
-- Cette procédure décrit le fonctionnement d'un noeud standard.
-- messages declaration Node
[17]INT inMessagesNode :
[17]INT inMessagesID :
[17]INT inMessagesHopCount :
[10]INT recusNode, recusID :
INT outMessageID, outMessageHopCount, outMessageNode :
INT nbRecep, nbAlertes, timerRepete, enAttente :
-- Code of procedure Node
SEQ
  -- d'abord, on initialise les différentes valeurs
  SEQ i=0 FOR 17
    SEQ
      inMessagesNode[i], inMessagesID[i], inMessagesHopCount[i] := 0, 0, 0
      outMessageID, outMessageHopCount, outMessageNode := 0, 0, 0
  SEQ i=0 FOR 10
    recusNode[i], recusID[i] := 0, 0
  nbRecep, nbAlertes, timerRepete, enAttente := 0, 0, 0, 0
  -- Boucle de fonctionnement normal
  WHILE TRUE
    SEQ
      PAR -- Communications
        PAR i=0 FOR SIZE in
          in[i] ? inMessagesNode[i] ; inMessagesID[i] ; inMessagesHopCount[i]
        PAR j=0 FOR SIZE out
          SEQ
            IF
              enAttente = 0
                out[j] ! outMessageNode ; outMessageID ; outMessageHopCount
            TRUE
              out[j] ! 0 ; 0 ; 0
      SEQ -- Traitements
        IF
          timerRepete > 0 -- On est en attente d'envoi de messages. On oublie tout.
```

```

timerRepete := timerRepete - 1
enAttente = 1 -- Il est temps d'envoyer notre message!
enAttente := 0
TRUE -- Si on est arrivé là, c'est que l'on n'a rien à envoyer, alors on peut analyser le reste
SEQ
  SEQ -- Analyse des messages reçus
  nbRecep := 0
  SEQ i=0 FOR SIZE in
    IF
      inMessagesNode[i] > 0
      SEQ
        nbRecep := nbRecep + 1
        outMessageNode := inMessagesNode[i]
        outMessageID := inMessagesID[i]
        outMessageHopCount := inMessagesHopCount[i] + 1
      TRUE
      SKIP
  SEQ -- Choix du comportement à adopter
  IF
    nbRecep > 1 -- Il y a eut brouillage!!!
    outMessageNode, outMessageID, outMessageHopCount := 0, 0, 0
    nbRecep = 1 -- On a reçu qu'une seule alerte
    SEQ i=0 FOR nbAlertes -- On teste le tableau d'alertes reçues
      IF
        (recusNode[i] = outMessageNode) AND (recusID[i] = outMessageID) -- On l'a déjà reçu...
        outMessageNode, outMessageHopCount, outMessageID := 0, 0, 0
        TRUE -- On ne l'a pas reçue, il faut la relayer
        SEQ
          --outMessageHopCount := outMessageHopCount + 1
          enAttente, timerRepete := 1, id*10
          recusNode[nbAlertes], recusID[nbAlertes] := outMessageNode, outMessageID
          nbAlertes := nbAlertes + 1
      TRUE -- Dans les autres cas, on a rien à transmettre
      outMessageNode, outMessageID, outMessageHopCount := 0, 0, 0

```

:

4 Comment localiser le noeud initial ?

4.1 Cas d'un protocole standard

Si l'on utilise, sur notre réseau, un protocole permettant, localement, l'établissement d'une topographie (pour chaque noeuds, établissement de la liste des noeuds voisins), alors cette topographie peut sans doute être récupérée par le centre de surveillance.

A partir de ces topographies, sans doutes est-il possible d'établir une topographie générale du réseau. Cependant, la résolution de ce problème étant probablement NP-complet, ou très longue, il serait sans doutes bon de l'effectuer AVANT le départ d'un feu.

La localisation du noeud peut ainsi être assez fine, en disposant dans le paquet d'alerte du seul ID du noeud initial.

4.2 Cas du protocole présenté

Le protocole présenté n'effectue aucune inspection du réseau avant une alerte. La topographie du réseau n'est donc pas connue à l'avance.

Cependant, il est possible de localiser le noeud initiale grace aux seules informations du paquet. Plus le nombre de passerelles recevant l'alerte est important, plus la localisation peut être fine.

La Figure 2 montre le cas ou trois passerelles reçoivent l'alerte. Avec le nombre de saut, et connaissant l'écart moyen entre chaques noeuds, il est possible de localiser, pour chaques passerelles, un périmètre dans lequel le noeud initial est susceptible de se trouver.

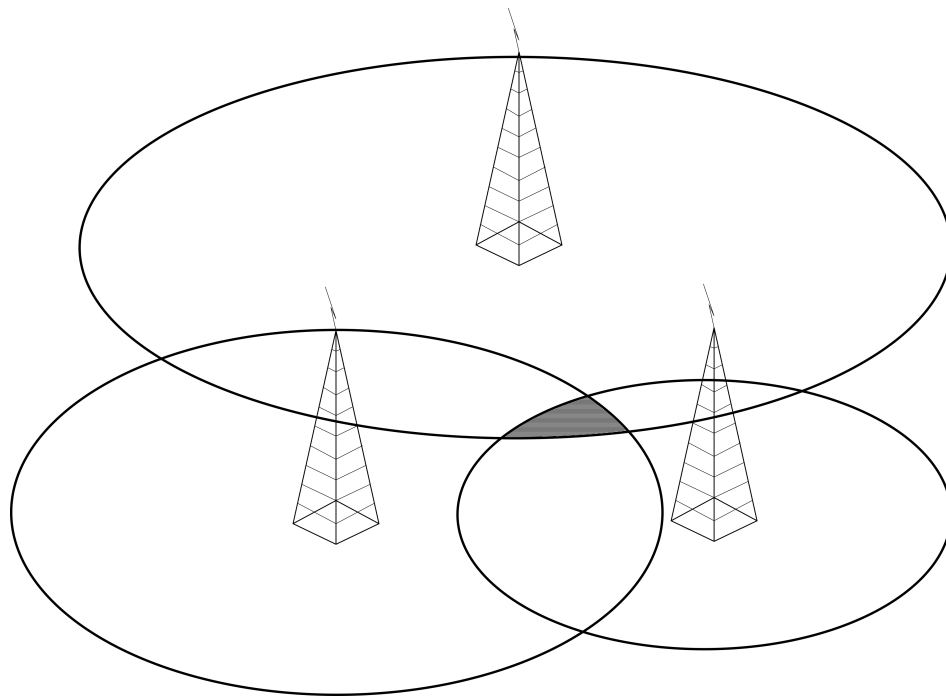


FIG. 2 – Schéma de localisation d'un noeud en alerte, en disposant uniquement du nombre de sauts, et de l'écart moyen.