
Surveillance vidéo sur réseaux de capteurs sans-fils : ordonnancement adaptatif avec prise en compte de la criticité.

Abdallah Makhoul — Rachid Saadi — Congduc Pham

LIUPPA, Université de Pau et des Pays de l'Adour (UPPA)

{abdallah.makhoul, rachid.saadi, congduc.pham}@univ-pau.fr

RÉSUMÉ. Faire de la vidéo surveillance avec des capteurs vidéo sans-fil possédant des ressources très limitées nécessite d'économiser l'énergie de manière efficace mais il est aussi très important de s'assurer que les objectifs de l'application soient atteints. Dans cet article, nous proposons un ordonnancement adaptatif de l'activité des nœuds vidéo pour augmenter la durée de vie du réseau en prenant en compte la criticité des applications de surveillance. Nous présentons une approche distribuée qui maintient la couverture et la connectivité de la zone à surveiller en fournissant plusieurs ensembles couvrants pour gérer la redondance des champs de vision des caméras. Nous proposons ensuite un modèle d'activité multi-niveaux qui introduit des fonctions de comportement modélisées par des courbes de Bézier, définissant ainsi différentes classes d'applications. Ces propositions sont simulées numériquement et nous présentons dans l'article leurs performances.

ABSTRACT. In video surveillance with resource-constrained devices, such as wireless video sensor nodes, power conservation is an important issue but one must also ensure that the application's objectives are met. In this paper, we address the problem of adaptive scheduling of sensor node's activity to extend the network lifetime while fulfilling the surveillance application needs. We first present a distributed algorithm that ensures both coverage of the deployment area and network connectivity by providing multiple cover sets to manage Field of View redundancies. Then, we propose a multiple-level activity model that uses behavior functions modeled by modified Bezier curves to define application classes and allow for adaptive scheduling. Simulation results are presented to validate the performance of the proposed approach.

MOTS-CLÉS : Réseaux de capteurs, Surveillance Vidéo, Ordonnancement

KEYWORDS: Sensor networks, Video surveillance, Scheduling

1. Introduction

Un réseau de capteur (RdC) vidéo sans-fils consiste en un ensemble de nœuds autonomes dotés d'une petite caméra embarquée. Ce type de réseau permet de déployer un grand nombre d'applications et dans cet article nous nous intéressons plus particulièrement aux applications de surveillance. Les applications sur RdC qui sont orientées surveillance ont des besoins spécifiques du fait de leur criticité. Par exemple, la qualité des images capturées et transmises doit être en adéquation avec les besoins et objectifs de l'application. De telles infrastructures de surveillance perdent très rapidement leurs intérêts si les scènes importantes ne sont pas correctement capturées.

Un aspect important en surveillance est la couverture et son maintien. Ce problème a été largement étudié dans les systèmes multi-robots et les RdCs de type scalaire [WAN 07a, BAH 08a, GAL 08, BAH 08b] en considérant une capacité de capture omnidirectionnelle. Ainsi 2 nœuds sont considérés comme redondants s'ils sont proches l'un de l'autre. Dans les RdCs vidéo, les caméras possèdent un champ de vision (CdV) et éventuellement des capacités de zoom. 2 nœuds peuvent ainsi être redondants même s'ils sont relativement éloignés l'un de l'autre. Parfois plusieurs vues sont souhaitables pour résoudre les ambiguïtés, parfois les nœuds éloignés peuvent fournir des informations mieux exploitables en fonction des conditions météo par exemple.

Un autre aspect très important est celui de la gestion de l'énergie. La rareté de cette ressource va avoir un impact très fort sur la couverture puisqu'il n'est pas réaliste que tous les nœuds puissent être actifs en même temps. Lorsque le déploiement est aléatoire, il peut y avoir une grande redondance entre les nœuds et une approche couramment utilisée est de définir un sous-ensemble de nœuds qui seront actifs pendant que d'autres seront inactifs. Le résultat est un ordonnancement de l'activité des nœuds qui maintient la couverture et la connectivité de la zone à surveiller. De plus, il est souhaitable de pouvoir définir plusieurs niveaux d'activité qui peuvent correspondre au nombre d'images qui seront capturées par unité de temps. Par exemple, certaines applications se focalisent sur la surveillance de la zone frontière plutôt que de la zone intérieure. Dans ce cas, les nœuds se situant à la frontière doivent capturer les images à un rythme plus soutenu que ceux se situant à l'intérieur, qui, à la rigueur, peuvent même se mettre en veille. Disposer de plusieurs niveaux d'activité est aussi nécessaire car les applications de surveillance comme la détection d'intrusion doivent pouvoir être opérationnelles sur le long terme puisque personne ne sait quand une intrusion se produira. En ce qui concerne les nœuds vidéo qui sont l'objet de cet article, la capacité à avoir plusieurs niveaux d'activité est encore plus indispensable que dans les réseaux de capteurs scalaires traditionnels (température, pression, ...) car capturer et transmettre des images consomme beaucoup plus d'énergie.

Cet article présente une approche pour ordonnancer de manière adaptative l'activité des nœuds vidéo en fonction de la couverture, des considérations énergétiques et des objectifs de l'application. Le but final est de fournir les algorithmes nécessaires à une application de surveillance pour exprimer ses objectifs. En ce qui concerne la couverture, notre approche utilise plusieurs ensembles couvrant pour gérer la redondance des CdV des caméras. Elle est basée sur un algorithme distribué qui aide chaque nœud à organiser ses voisins en des ensembles non-disjoints, chacun d'eux étant un ensemble couvrant qui recouvre son CdV. Ensuite, en se basant sur l'activité de ses voisins, un nœud décide d'être actif ou non, sans pour cela compromettre la couverture de sa zone. Puis, pour prendre en compte l'énergie et les objectifs de l'application, notre approche apporte un modèle d'activité multi-

niveaux qui comporte un mode par défaut et 2 modes forçant les nœuds sur un mode "high" et "low" en fonction de la criticité de l'application. Ces modes opératoires utilisent des fonctions de comportement modélisées par des courbes de Bézier, définissant ainsi des classes d'applications.

Le reste de l'article est organisé de la manière suivante. La section 2 présente un modèle simple de couverture permettant de définir plusieurs ensembles couvrants pour chaque nœud. Un modèle d'activité multi-niveaux est présenté dans la section 3. Les résultats de simulation sont présentés dans la section 4 avant de conclure.

2. Couverture et ordonnancement des nœuds vidéo

Le problème de la couverture dans les réseaux de capteurs vidéo n'est pas récent. Plusieurs travaux ont déjà traité ce problème de plusieurs manières. Cependant, nous pouvons distinguer deux catégories de couverture :

- *Couverture des cibles pré-déterminées*, qui consiste à trouver un sous-ensemble de nœuds connexes et qui assure la surveillance d'un ensemble de cibles dont la position est connue *a priori*.
- *Couverture d'une zone*, qui consiste à trouver un sous-ensemble de nœuds connexes et qui assure la surveillance de toute la zone de déploiement.

La majorité des travaux existants considèrent le problème de la couverture des cibles pré-déterminées [CAI 07, AI 06, LIU 06, WAN 07b, CHE 05]. Par exemple, les auteurs dans [AI 06] proposent un modèle directionnel en considérant qu'un nœud vidéo peut ajuster la direction de son CdV. L'idée est de trouver un sous-ensemble minimal de directions qui couvre le plus grand nombre de cibles. Par contre, l'idée énoncée dans [CAI 07] consiste à ordonner les nœuds dans des sous-ensembles non-disjoints sachant que chacun d'eux couvre la totalité des cibles. Les travaux concernant le problème de la couverture d'une zone 2D se basent sur l'étude du déploiement des nœuds afin de réduire le taux de chevauchement (zones couvertes plusieurs fois par plusieurs capteurs). Cette réduction peut être basée sur un modèle probabiliste qui précise le pourcentage de la couverture demandée [MA 07]. D'autres travaux considèrent des caméras pouvant tourner afin d'augmenter la surface de surveillance [TAO 06].

Dans ce papier, nous considérons le problème de la couverture d'une zone avec un déploiement aléatoire. Nous proposons un algorithme permettant à chaque nœud vidéo de définir plusieurs sous-ensembles non disjoints parmi ses voisins, sachant que chaque sous-ensemble couvre la totalité de la surface de son CdV. Cependant, un nœud peut décider d'être actif ou inactif selon l'activité de ses voisins.

2.1. Modèle de couverture et construction des ensembles couvrants

La zone de surveillance d'un nœud vidéo v est généralement représentée par son CdV. Nous supposons que les nœuds vidéo sont déployés aléatoirement, qu'ils sont statiques (positions et direction du CdV) et qu'ils connaissent ces informations. Nous considérons donc un modèle 2D qui définit le CdV par le 4-uplet $v(P, R_s, \vec{V}, \alpha)$, où P représente la position, R_s le rayon de couverture, \vec{V} le vecteur de direction de la camera, et α le demi-angle de vision du CdV. La figure 1 (a) illustre le

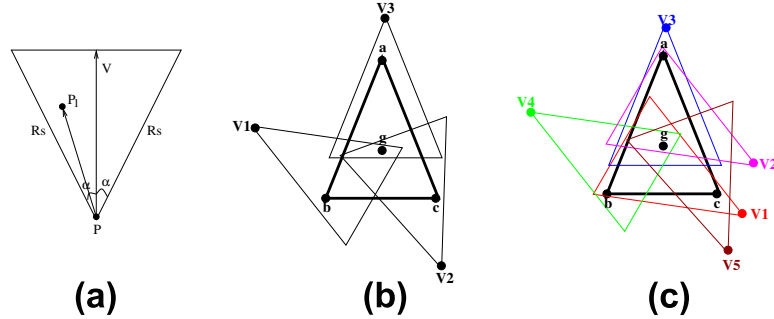


Figure 1. *Modèle de couverture.*

CdV d'un nœud vidéo v . Dans la suite de notre approche nous supposons que tous les nœuds sont identiques : même rayon de couverture R_s et même valeur α .

Notre approche est complètement distribuée : chaque nœud doit assurer sa propre couverture indépendamment des autres. Un nœud v couvre généralement une surface triangulaire grâce à son CdV. Cependant son objectif est d'assurer tant qu'il est vivant la couverture de cette surface, soit par lui-même (état actif), soit par des nœuds redondants (état inactif).

Définition 1 Nous définissons l'ensemble couvrant i du nœud v , noté $Co_i(v)$, comme étant un sous-ensemble de nœuds vidéo tel que : $\bigcup_{v' \in Co_i(v)} (CdV \text{ de } v')$ couvre la surface du CdV de v .

Définition 2 $Co(v) = \{Co_i(v)\}$ désigne tous les ensembles couvrant du nœud v .

Un exemple d'un ensemble couvrant est illustré dans la figure 1 (b), où les nœuds v_1 , v_2 et v_3 couvrent la surface du CdV du nœud v représentée par abc . La méthode que nous adoptons dans notre approche pour trouver l'ensemble $Co(v)$ est de représenter la surface du CdV par les quatre points suivants : a , b , c et g (le centre de gravité de (abc)) comme le montre la figure 1 (b). Dans une seconde étape, nous considérons que la surface du CdV du nœud v est couverte par un ensemble $Co_i(v) \in Co(v)$ si :

- 1) $\forall v' \in Co_i(v)$, v' couvre le point g et au moins un des points $\{a, b, c\}$,
- 2) a, b, c et g sont tous couverts par les nœuds de $Co_i(v)$.

Pour qu'un nœud v puisse calculer $Co(v)$, il lui suffit de trouver les ensembles A, B, C et G qui sont l'ensemble des nœuds $v' \in N(v)$ tels que v' couvre le point correspondant. $N(v)$ représente l'ensemble des nœuds voisins du nœud v (les nœuds avec qui il peut communiquer). Nous avons ensuite $AG = \{A \cap G\}$, $BG = \{B \cap G\}$, $CG = \{C \cap G\}$. $Co(v)$ est alors formé par le produit cartésien $\{AG \times BG \times CG\}$. Notons que la complexité du calcul de l'intersection entre deux ensembles s'élève à $O(k + m)$ où k et m sont les cardinaux respectifs des 2 ensembles. Comme la taille des ensembles A, B, C et G est réduite, un nœud peut facilement trouver leurs intersections deux à deux. Dans la figure 1 (c), la surface du CdV du nœud v est représentée par $(abcg)$ et nous avons $AG = \{v_2, v_3\}$, $BG = \{v_1, v_4\}$, $CG = \{v_1, v_5\}$, et l'ensemble $Co(v) = \{\{v\}, \{v_2, v_1\}, \{v_3, v_1\}, \{v_2, v_4, v_5\}, \{v_3, v_4, v_5\}\}$.

Cette méthode simple permet aux nœuds de trouver parmi leurs voisins ceux qui couvrent leur CdV. Néanmoins il peut arriver qu'un ensemble couvrant ne couvre pas la totalité de la surface du CdV (occlusion par exemple). Il est possible de considérer plus de points pour représenter la surface du CdV, comme par exemple les médianes des segments [ab], [ac] et [bc] mais cela réduit le nombre des ensembles couvrant et peut augmenter sensiblement la complexité des calculs. Notons que la méthode que nous proposons reste une méthode simple pour définir des ensembles couvrants qui vont nous servir pour l'ordonnancement.

2.2. Ordonnancement de l'activité des nœuds vidéo

Notre approche pour la couverture des nœuds vidéo fonctionne en 2 phases : (i) initialisation où chaque nœud construit ses ensembles couvrants, (ii) ordonnancement de l'activité et un nœud décidera d'être actif ou non.

2.2.1. Construction de l'ensemble $Co(v)$

Dans cette phase, chaque nœud cherche à trouver tous ses ensembles couvrant. Chaque nœud diffuse sa position P et sa direction \vec{V} à ses voisins et reçoit leurs informations. Dans une seconde étape, il calcule les ensembles A, B, C et G afin de trouver les ensembles couvrants $Co_i(v)$ (voir section 2.1).

La majeure partie des solutions au problème de couverture de surfaces consiste à trouver des ensembles couvrants disjoints [WAN 07a, BAH 08a, GAL 08, BAH 08b]. Dans notre approche distribuée, un nœud peut construire des ensembles ayant un ou plusieurs capteurs en commun. Cette dépendance doit être prise en considération dans la phase d'ordonnancement pour ne pas épuiser les nœuds en commun et essayer d'équilibrer l'activité des nœuds. Nous utilisons le niveau d'énergie d'un ensemble couvrant comme critère de priorité. Ce niveau peut être égal au niveau de la batterie du nœud dans $Co_i(v)$ qui possède le moins d'énergie (sachant que le nœud v lui même appartient à l'ensemble $Co(v)$).

2.2.2. Algorithme d'ordonnancement

L'activité des nœuds se répartit en itérations. Pour chaque itération, chaque nœud décide d'être actif ou non selon l'activité de ses voisins et son niveau d'énergie. Il commence à ordonner ses ensembles couvrants selon leur cardinalité, en donnant la priorité à ceux qui contiennent le moins d'éléments. Dans le cas où deux ou plusieurs ensembles ont le même cardinal, ils seront classés selon leur niveau d'énergie. Un nœud vidéo v reçoit les messages d'activité de ses voisins et teste si un de ses ensembles couvrants est satisfait ou non. Si un $Co_i(v)$ est actif, v s'endort et diffuse sa décision à ses voisins. D'un autre côté si aucun $Co_i(v)$ n'est satisfait, il décide de rester actif et diffuse également sa décision.

Le nœud v ordonne son ensemble $Co(v)$ selon la priorité (qui change durant la durée de vie du réseau selon la consommation d'énergie) en classant le singleton $\{v\}$ en dernier. Si l'un des nœuds appartenant à un ensemble couvrant de v décide d'être inactif alors l'ensemble ne peut plus être satisfait et v passe au suivant jusqu'à ce qu'il arrive au singleton, auquel cas $\{v\}$ décide d'être actif. Ce processus est illustré dans l'algorithme 1.

Algorithme 1 Ordonnancement d'un nœud v

```

1:  $v$  est actif
2:  $v$  ordonne ses ensembles couvrants  $Co_i(v) \in Co(v) \ i = 1, 2, \dots |Co(v)|$ 
3:  $i \leftarrow 1$ 
4: tantque  $i \leq |Co(v)|$  faire
5:    $v$  commence par l'ensemble couvrant ayant la plus grande priorité  $Co_i(v)$ 
6:   si voisin  $v'$  décide d'être inactif alors
7:     si  $v' \notin Co_i(v)$  alors
8:       continue avec  $Co_i(v)$ 
9:     finsi
10:  sinon
11:     $v$  choisit l'ensemble couvrant suivant  $Co_{i+1}(v)$ 
12:     $i \leftarrow i + 1$ 
13:  finsi
14:  si  $v'$  décide d'être actif alors
15:    si  $v' \in Co_i(v)$  alors
16:      continue avec  $Co_i(v)$ 
17:    finsi
18:  finsi
19:  si  $\forall v', v' \in Co_i(v), v'$  est actif alors
20:     $v$  devient inactif et diffuse sa décision à ses voisins
21:  finsi
22: fin tantque
23: si aucun  $Co_i(v)$  n'est satisfait alors
24:    $v$  reste en mode actif et diffuse sa décision à ses voisins
25: finsi

```

3. Régulation en fonction de la criticité

Dans les réseaux de capteurs vidéo, l'autonomie d'un capteur est directement liée à sa vitesse de capture d'images. Autrement dit, un nœud qui capture lentement (e.g. 1 image/s) restera actif plus longtemps qu'un autre nœud capturant à une vitesse plus grande (e.g. 3 images/s). D'un autre côté, la vitesse de capture peut être synonyme de qualité de surveillance. Cet indice de qualité n'est pas universel mais dépend de manière intrinsèque de la criticité de l'application. Une application ayant pour but la détection d'intrusion doit avoir une vitesse de capture assez élevée afin d'éviter de manquer une intrusion. Dans d'autres applications la vitesse de capture peut être extrêmement réduite jusqu'à quelques images par jour (surveillance environnementale). Ainsi, il est important de pouvoir ajuster la vitesse de capture afin de permettre un juste compromis entre autonomie, criticité et couverture.

Dans ce qui suit, nous allons définir la variable r^0 qui indique le niveau de criticité de l'application. r^0 est compris dans l'intervalle $[0,1]$ où 0 représente l'application la moins critique et 1 l'application la plus critique. Ainsi, tous les capteurs vidéo sont initialisés, avant leur déploiement, par la valeur r^0 qui correspond à la criticité de l'application. Afin de définir la vitesse de capture de chaque capteur nous allons présenter deux approches nommées respectivement : "Approche par régulation statique" et "Approche par régulation dynamique".

3.1. Approche par régulation statique

L'approche par régulation statique est une approche naïve qui consiste à fixer, en fonction de r^0 , une vitesse de capture constante pendant toute la durée de vie du capteur. Comme l'illustre la figure 2 (a), la vitesse de capture est fixée proportionnellement à r^0 . Ainsi, on peut remarquer que pour des applications très critiques ($r^0 = 1$) les nœuds du réseau vont capturer au maximum de leur capacité (max). De même, plus la criticité diminue plus la vitesse de capture diminue également.

Cette approche a l'avantage d'être simple à mettre en place mais présente cependant un certain nombre d'inconvénients : (i) configurer les nœuds pour fonctionner au maximum de leur capacité fournit un niveau de surveillance maximal mais présente le désavantage d'épuiser rapidement le réseau ; (ii) inversement, faire en sorte que les nœuds capture très lentement prolonge la durée de vie du réseau au détriment de la qualité de surveillance ; (iii) choisir le juste milieu semble être un compromis entre la qualité de capture et la longévité du réseau mais en même temps, pendant toute la durée de vie de ce dernier les capteurs sont tout le temps sous exploités.

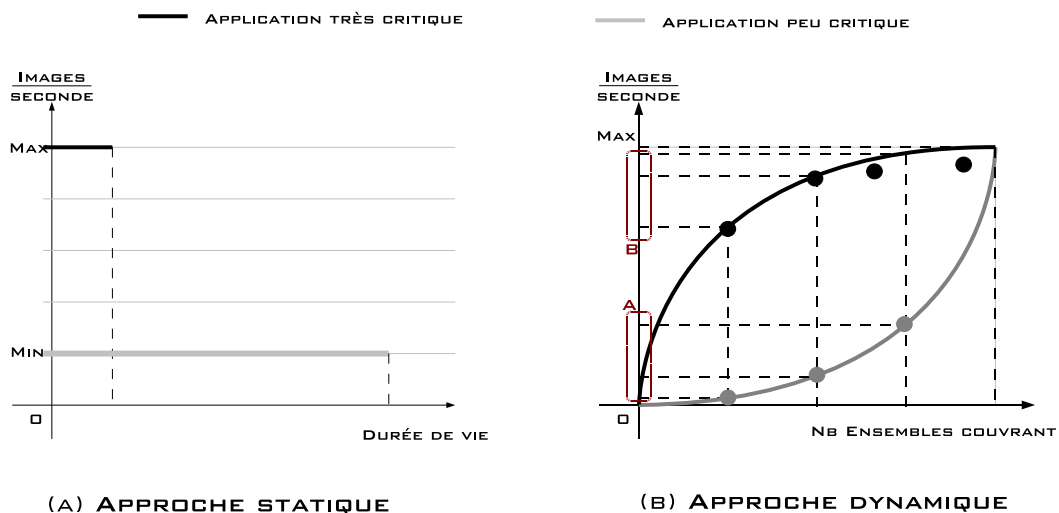


Figure 2. Approches pour la régulation de l'autonomie.

3.2. Approche par régulation dynamique

Dans le but d'exploiter de manière optimale la capacité du réseau de capteur, nous allons définir dans cette section une méthode qui permet de faire varier la vitesse de capture d'un nœud v en fonction de la taille de $Co(v)$. Autrement dit, plus la zone d'un capteur est couverte, plus ce dernier peut se permettre de capturer rapidement, et inversement, si la zone n'est pas suffisamment couverte le nœud responsable de celle-ci doit se préserver pour prolonger, le plus longtemps possible, la couverture globale. Afin de faire varier la vitesse de capture en fonction de la couverture, mais également

en fonction du degré de criticité, nous allons, dans ce qui suit, définir un modèle mathématique permettant de modéliser ce type de comportement.

3.2.1. Modélisation du comportement

La criticité d'une application définie de manière directe le comportement de celle-ci en terme de vitesse de capture. Ainsi afin de modéliser le comportement de n'importe quelle type d'application, nous avons utilisé une fonction cartésienne noté $y = BV_{r^0}(x)$ (BV pour **B**eha**V**ior). La valeur de x représente la cardinalité de l'ensemble des ensembles couvrants ($|Co|$). x prend des valeurs dans l'intervalle $[0, n]$ où n représente un haut niveau de couverture. Ce dernier est fixé en fonction de la topologie du réseau. y représente la vitesse de capture du capteur et est le résultat de la projection de x sur la courbe qui représente la niveau de criticité de l'application.

Tel que l'illustre la figure 2 (a), la fonction BV permet, en fonction du r^0 , de dessiner des courbes concaves et convexes à différents degrés. Ainsi, en fonction de la courbure dessinée par la fonction BV, deux classes sont définies comme suit :

– Classe à haut niveau de criticité : $0 \leq r^0 < 0.5$: applications non critiques ne nécessitant pas une grande vitesse de capture. Nous définissons cette caractéristique par une fonction de forme convexe. Comme l'illustre la figure 2 (b) zone A, les vitesses de capture qui seront attribuées sont proches du 0.

– Classe à haut niveau de criticité : $0.5 \leq r^0 \leq 1$: applications critiques nécessitant une grande vitesse de capture. Nous définissons cette caractéristique par une fonction de forme concave. Comme illustré dans la figure 2 (b) zone B, les vitesses de capture qui seront attribuées sont proches de *max* (vitesse de capture maximale).

Ainsi, le comportement d'une application peut être modélisé à l'aide de courbes qui vont d'une forme convexe (peu critique) à une forme concave (très critique).

3.2.2. Fonction de comportement BV

Pour définir un schéma de fonctions qui modélise nos courbes concaves et convexes, nous proposons d'utiliser les courbes paramétrées de Bézier qui ont été dans des travaux portant sur la modélisation de la subjectivité dans le domaine de la gestion de la confiance en environnement pervasif [SAA 07]. De cette manière la fonction BV peut dessiner n'importe quelle courbe approchant le point $P_1(b_x, b_y)$ qui se trouve dans le rectangle délimité par $P_0(0, 0)$ et $P_2(h_x, h_y)$. comme suit :

$$\begin{aligned}
 BV_{r^0} : [0, h_x] &\longrightarrow [0, h_y] \\
 X &\longrightarrow Y
 \end{aligned}$$

$$BV_{r^0, h_x, h_y}(X) = \begin{cases} \frac{(h_y - 2b_y)}{4b_x^2} X^2 + \frac{b_y}{b_x} X & \text{si } (h_x - 2b_x = 0) \\ (h_y - 2b_y)(\alpha(X))^2 + 2b_y \alpha(X), & \text{si } (h_x - 2b_x \neq 0) \end{cases} \quad (1)$$

$$\text{Telque } \begin{cases} \alpha(X) = \frac{-b_x + \sqrt{b_x^2 - 2b_x * X + h_x * X}}{h_x - 2b_x} \\ 0 \leq b_x \leq h_x \\ h_x > 0 \\ b_x = (1 - r^0) \times h_x \\ b_y = h_y \times r^0 \end{cases}$$

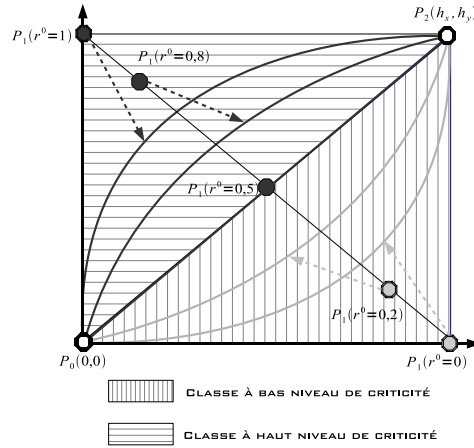


Figure 3. *Fonctions de comportement.*

Afin de modéliser les courbes décrites précédemment, il suffit de définir la fonction BV à travers les points suivants :

- le point d'origine $P_0(0, 0)$,
- le point repère $P_2(h_x, h_y)$: h_x représente la plus grande cardinalité de l'ensemble des ensembles couvrants considéré par le réseau et h_y représente la vitesse maximale de capture pouvant être atteinte par le capteur.
- le point de comportement $P_1(b_x, b_y)$ qui se déplace sur l'anti-diagonale afin de faire passer la courbe d'une forme concave à une forme convexe. Ce point est conditionné par $r^0 \in [0, 1]$. Par exemple, si $r^0 = 0$ alors le point P_1 aura les coordonnées $(h_x, 0)$, et inversement, si $r^0 = 1$, alors le point P_1 aura les coordonnées $(0, h_y)$.

4. Simulations et résultats

L'évaluation de nos propositions est faite par simulation avec le simulateur OMNet++ [OMN 09]. Les nœuds capteurs ont tous un rayon de communication de $30m$ et une distance de capture vidéo de $25m$ avec $\alpha = \pi/6$. La batterie possède 100 unités, la position est aléatoire, ainsi que la direction de visé \vec{V} de la caméra. La simulation débute par une découverte de voisinage : chaque nœud récupère les positions et la ligne de visée de leurs voisins et détermine les ensembles AG , BG et CG définis précédemment. A chaque itération chaque nœud décide d'être actif ou non. Pour ces simulations chaque itération dure 1s. Une simulation se termine lorsque les nœuds possédant encore de l'énergie ne forment plus un ensemble connecté. Les résultats présentés sont la moyenne de 15 exécutions différentes.

4.1. Résultats sur la couverture

La densité varie de 50 à 200 nœuds dans la surface de $100m * 100m$ (soit entre $0,005$ et $0,02$ capteur/ m^2) et nous regardons à chaque itération le pourcentage de nœuds actifs et le pourcentage de surface couverte. A la fin de chaque itération chaque nœud décrémente sa batterie d'une unité. La figure 4 (a) montre l'évolution du pourcentage de nœuds actifs. A partir de l'itération 100, certains nœuds sans ensemble couvrant n'ont plus d'énergie et meurent. Cependant, on voit que le nombre de nœuds actifs pour garantir la couverture maximale est inférieur à 68%, pour 50 nœuds, à 36% pour 200 nœuds.

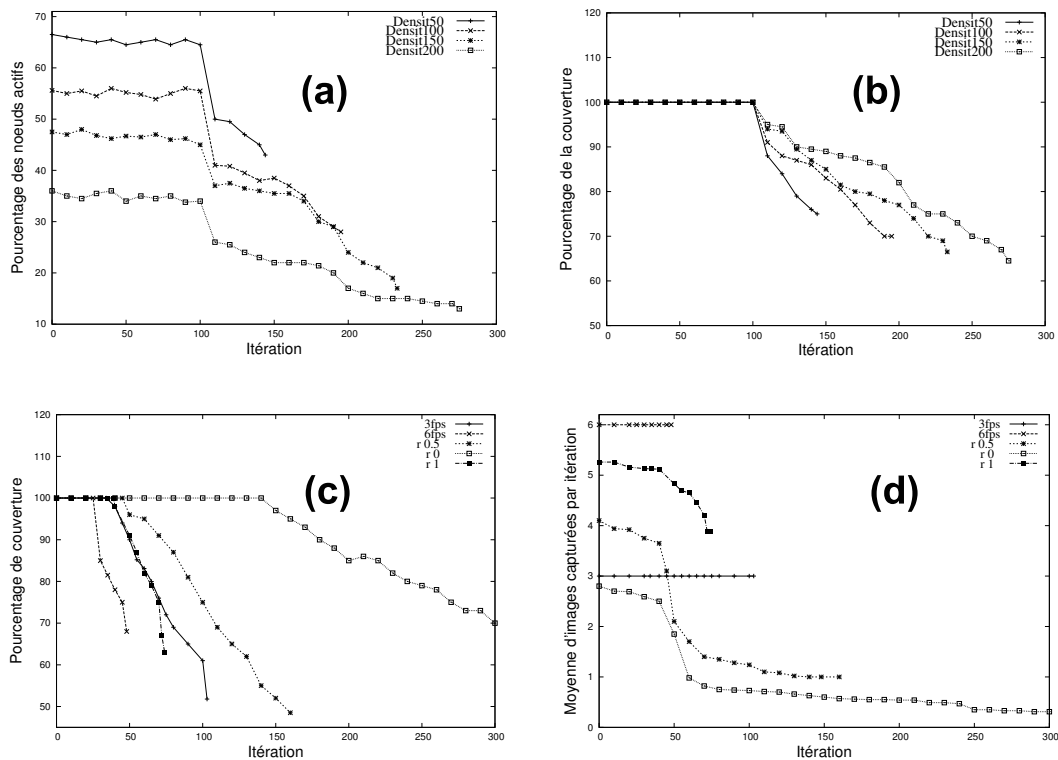


Figure 4. Résultats.

L'objectif principal de notre approche est de maintenir une couverture maximale (équivalente au cas où tous les capteurs sont actifs) à chaque itération. La figure 4 (b) montre le pourcentage de couverture à chaque itération. Nous pouvons voir que notre approche permet de conserver au moins 66% de la couverture dans tous les cas.

4.2. Impact de la criticité de l'application

150 capteurs sont déployés sur une surface de $75m * 75m$ pour augmenter la densité tout en gardant des temps de simulation raisonnables. Chaque capteur peut capturer un certain nombre d'images par seconde (fps, entre 0 et 6) auquel cas la batterie est décrémentée d'une unité pour chaque image. Nous évaluons 2 scénarii de vitesse de capture fixe (3fps et 6fps) et un scénario dynamique où 3 niveaux de criticité sont définis en fonction du modèle proposé dans la section 3 : $r^0 = 0$ (faible criticité), $r^0 = 0.5$ et $r^0 = 1$ (forte criticité). Dans ce cas, la vitesse de capture d'un nœud dépend de la taille de son ensemble couvrant.

La figure 4 (c) montre le pourcentage de couverture lorsque la vitesse de capture varie (plus la vitesse de capture est grande, moins les nœuds durent longtemps). La figure 4 (d) montre la vitesse moyenne de capture de tous les nœuds actifs à chaque itération. Il est souhaitable pour une application de surveillance de maximiser à la fois la vitesse de capture, la couverture et la durée de vie du réseau.

Lorsque le niveau de criticité r^0 varie, la vitesse de capture d'un nœud v dépend de la taille de $Co(v)$ mais dépend aussi de sa fonction BV. Au delà du fait que notre modèle permet à une application de spécifier un niveau de criticité adapté à ses objectifs, ce modèle optimise la couverture ainsi que la durée de vie du réseau. Par exemple, on peut voir dans la figure 4 (d) que le scénario "forte criticité" donne une vitesse moyenne de capture de 4.63fps qui devrait donner dans la figure 4 (c) une courbe entre le cas 6fps et le cas 3fps. On voit dans la figure 4 (c) que le cas "forte criticité" donne une courbe très proche du cas 3fps indiquant ainsi que notre modèle dynamique permet d'avoir le même pourcentage de couverture (et donc en conservant la durée de vie du réseau) avec une meilleure vitesse de capture.

5. Conclusion

Dans cet article nous avons proposé un ordonnancement prenant en compte la criticité des applications de vidéo surveillance sur réseaux de capteurs sans-fils. Notre approche permet de pouvoir fixer un degré de criticité qui va se traduire sur le réseau en une vitesse de capture en adéquation avec les besoins de l'application et en fonction de la redondance des champs de vision de chaque capteur. Les résultats de simulation montrent que l'approche proposée permet d'optimiser à la fois la vitesse de capture, la couverture et la durée de vie du réseau. Nos recherches futures en ce domaine s'orientent vers une gestion du risque au niveau du réseau en permettant aux nœuds de se configurer automatiquement en fonction des stimuli extérieurs.

6. Bibliographie

- [WAN 07a] C. Wang, M. T. Thai, Y. Li, F. Wang, and W. Wu, "Coverage based information retrieval for lifetime maximization in sensor networks," *In IEEE GlobeCom*, Nov. 2007.
- [BAH 08a] J. Bahi, A. Makhoul, and A. Mostefaoui, "Hilbert mobile beacon for localization and coverage in sensor networks," *International Journal of Systems Science, Taylor & Francis*, vol. 39, no. 11, pp. 1081–1094, Novembre 2008.

- [GAL 08] A. Gallais, J. Carle, D. Simplot-Ryl, and I. Stojmenovic, "Localized sensor area coverage with low communication overhead," *IEEE Transactions on Mobile Computing (TMC)*, vol. 7, no. 5, pp. 661–672, 2008.
- [BAH 08b] J. Bahi, A. Makhoul, and A. Mostefaoui, "Improving lifetime and coverage through a mobile beacon for high density sensor networks," *The Second IEEE International Conference on Sensor Technologies and Applications, SENSORCOMM08*, august 2008.
- [CAI 07] Y. Cai, W. Lou, M. Li, and X.-Y. Li, "Target-oriented scheduling in directional sensor networks," *26th IEEE INFOCOM*, 2007.
- [AI 06] J. Ai and A. A. Abouzeid, "Coverage by directional sensors in randomly deployed wireless sensor networks," *Journal of Combinatorial Optimization*, vol. 11, no. 1, 2006.
- [LIU 06] H. Liu, P. Wan, and X. Jia, "Maximal lifetime scheduling for sensor surveillance systems with k sensors to one target," *IEEE Transactions on Parallel and Distributed Systems*, vol. 17, no. 12, pp. 1526–1536, 2006.
- [WAN 07b] J. Wang, C. Niu, , and R. Shen, "Randomized approach for target coverage scheduling in directional sensor network," *LNCS - Springer*, 2007.
- [CHE 05] M. X. Cheng, L. Ruan, and W. Wu, "Achieving minimum coverage breach under bandwidth constraints in wireless sensor networks," in *IEEE INFOCOM*, 2005.
- [MA 07] H. Ma and Y. Liu, "Some problems of directional sensor networks," *International Journal of Sensor Networks*, vol. 2, no. 1-2, pp. 44–52, 2007.
- [TAO 06] D. Tao, H. Ma, and L. Liu, "Coverage-enhancing algorithm for directional sensor networks," *Lecture Notes in Computer Science - Springer*, pp. 256–267, November 2006.
- [SAA 07] R. Saadi, O. Hasan, J. Pierson, and L. Brunie, "Establishing trust beliefs based on a uniform disposition to trust," *IEEE international Conference on Signal Image and Technologie Based Systems*, 2007.
- [OMN 09] OMNeT++, "<http://www.omnetpp.org/>."