# Partionnement et Geocasting dans les Réseaux Mobiles Ad hoc et Collecte des Données dans les Réseaux de Capteurs

Idrissa Sow

MIS / Université de Picardie Jules Verne

Amiens, 4 juin 2009



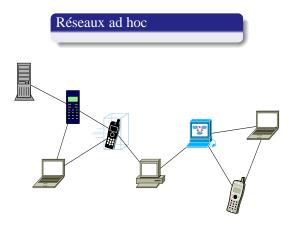
- Introduction et Problématique
- 2 Partitionnement
- 3 Geocasting
- 4 Architecture Virtuelle
- 5 Conclusion

- 1 Introduction et Problématique
- 2 Partitionnement
- 3 Geocasting
- 4 Architecture Virtuelle
- 5 Conclusion



Contexte

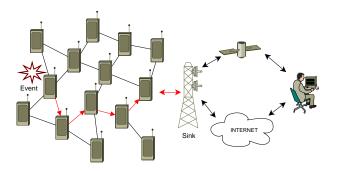
## Introduction



Contexte

## Introduction

## Réseaux de capteurs



#### Difficultés communs

- ▶ L'accès au ressources partagées (Couche MAC)
- Le routage
- L'énergie ⇒ durée de vie
- ► L'espace de stockage

#### **Motivations**

- ▶ Pourquoi ?
- ► Identifier les problèmes
- Proposer des approches

- 1 Introduction et Problématique
- 2 Partitionnement
- 3 Geocasting
- 4 Architecture Virtuelle
- 5 Conclusion

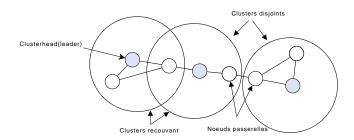
## Présentation du problème

Le partionnement consiste à regrouper les nœuds proches géographiquement et d'en élire un chef selon un processus distribué d'élection de leader.



# Présentation du problème

Le partionnement consiste à regrouper les nœuds proches géographiquement et d'en élire un chef selon un processus distribué d'élection de leader.



# Présentation du problème

#### Plusieurs stratégies

- o Choix du leader et politique de maintiens
- Clusters recouvrant ou disjoints ⇒ notion de diamètre
- o Clusters denses ou moins denses ⇒ notion de cardinalité



Ephremides et Baker [1981]

Lowest-ID heuristic

Gerla et Perekh [1987]

Hightest-Degree heuristic

#### Ephremides et Baker [1981]

Lowest-ID heuristic

## Gerla et Perekh [1987]

Hightest-Degree heuristic

#### Basagni [1999]

Node Weight heuritic (DCA & DMAC)



#### Idée de Basagni

- o Deux nœuds ne peuvent pas avoir le même poids
- Exemple de poids = inverse de la vitesse

#### Idée de Basagni

- Deux nœuds ne peuvent pas avoir le même poids
- Exemple de poids = inverse de la vitesse

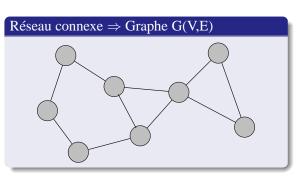
# Poids plus réaliste

#### Contribution

#### Former des clusters

Selon une heuristique de poids plus réaliste ne nécessitant de poids unique par nœud





- Voisinage  $\Gamma(u)$
- ∘ Degré d<sub>u</sub>

#### Chaque nœud u

 $\mathbf{I}$   $v_i$ : vitesse entre deux instants  $t_i$  et  $t_j$ 



#### Chaque nœud u

- $\mathbf{I}$   $v_i$ : vitesse entre deux instants  $t_i$  et  $t_j$
- 2  $M_u = \frac{1}{n} \sum_{i=1}^n v_i$ : vitesse moyenne



15 / 65

#### Chaque nœud u

- $\mathbf{1}$   $v_i$ : vitesse entre deux instants  $t_i$  et  $t_j$
- $2 M_u = \frac{1}{n} \sum_{i=1}^n v_i : \text{vitesse moyenne}$
- $\omega_u = 1/M_u$ : mobilité

#### Chaque nœud u

- $\mathbf{1}$   $v_i$ : vitesse entre deux instants  $t_i$  et  $t_j$
- $2 M_u = \frac{1}{n} \sum_{i=1}^n v_i : \text{vitesse moyenne}$
- $\omega_u = 1/M_u$ : mobilité
- 4  $E_r$  et  $E_s$ : Enérgies résiduelle et seuil

# Expression du poids

#### Poids de u

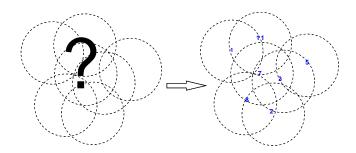
$$P_u = \{\omega_u, E_r(u), \phi_u, d_u, ID_u\}$$

- o  $\phi_u = \delta d_u$ : différence de degré ( $\delta$  une constante positive)
- $\circ$  *ID*<sub>u</sub>: identification

<sup>&</sup>lt;sup>1</sup>Stoimenovic et al avaient déjà émis cette idée mais ne l'ont pas mis en œuvre : 🔻 : 🔊 🤉 🧇

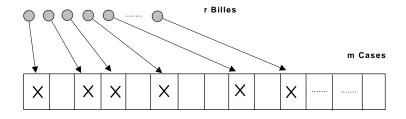
## Identification : Cas de Réseau anonyme

- ► Réseau anonyme de *n* nœuds
- ► Identification temporaire ⇒ identifiant unique



## Identification : Cas de Réseau anonyme

▶ Placer r billes dans m cases [Occupancy Problem]



Quelle sera la valeur minimale de m?



## Identification : Cas de Réseau anonyme

#### Théorème

Preuve Si n est le nombre total des nœuds connus de tous et N une borne supérieure de n alors la probabilité d'octroyer un identifiant unique à chaque nœud selon une distribution uniforme d'intervalle

$$[1 \cdots N^3]$$
 est supérieur à  $\exp(-\Theta(\frac{n^2}{N^3}))$ .

#### u est candidat au status de clusterhead

$$\forall z \in \Gamma(u) / P_u > P_z$$



 $ightharpoonup P_u > P_z$ 

1. 
$$(\omega_u > \omega_z) \wedge (E_r(u) > E_r(z)) \wedge (\phi(u) > \phi(z))$$



 $ightharpoonup P_u > P_z$ 

**1.** 
$$(\omega_u > \omega_z) \wedge (E_r(u) > E_r(z)) \wedge (\phi(u) > \phi(z))$$

**2.** 
$$(\omega_u = \omega_z) \wedge (E_r(u) = E_r(z)) \wedge (\phi(u) = \phi(z)) \wedge (ID_u > ID_z)$$

- $ightharpoonup P_u > P_z$
- **1.**  $(\omega_u > \omega_z) \wedge (E_r(u) > E_r(z)) \wedge (\phi(u) > \phi(z))$
- **2.**  $(\omega_u = \omega_z) \wedge (E_r(u) = E_r(z)) \wedge (\phi(u) = \phi(z)) \wedge (ID_u > ID_z)$

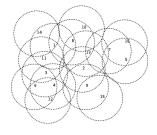
3. 
$$(\omega_u \ge \omega_z)$$

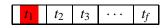
$$(E_r(u) \ge E_r(z)) \lor (E_r(u) \ge E_s)$$

$$(\phi(u) \ge \phi(z)) \lor \{(E_r(u) \ge E_s) \lor (E_r(u) \ge E_r(z))\}$$



## Formation des clusters

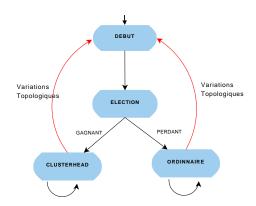




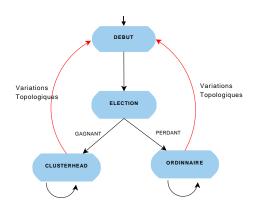
#### On suppose

- Suite finie d'instants  $\{t_1, t_2, \dots, t_i, \dots t_f\}$
- ▶ Deux primitives
  - o CH(u)
  - o JOIN(v,u)
- ► Tout message est réçu

## Formation des clusters

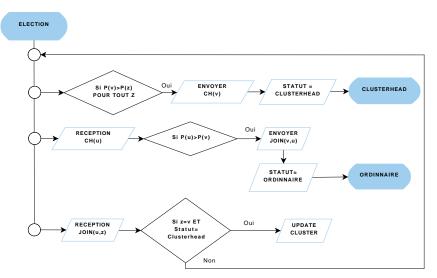


#### Formation des clusters

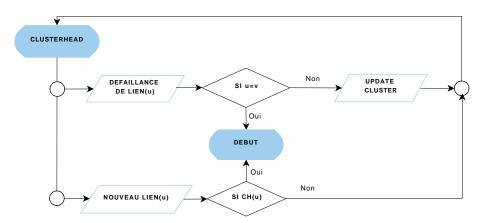




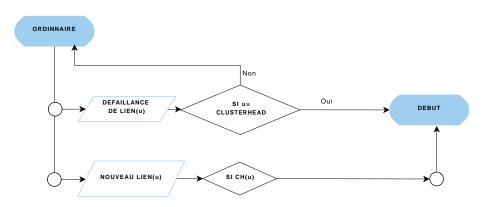
## Formation des clusters [nœud v]



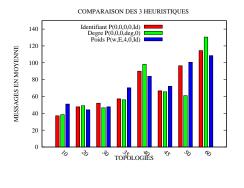
## Maintenance des clusters [nœud v]

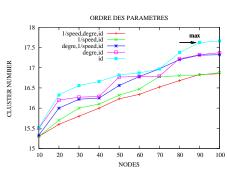


## Maintenance des clusters [nœud v]



# Quelques résulats de simulations

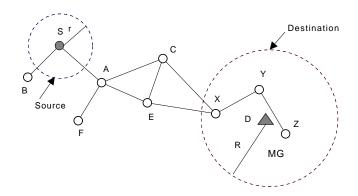




- Voisinage à 1-saut
- Diff de degré  $\delta = 4$

- 1 Introduction et Problématique
- 2 Partitionnement
- 3 Geocasting
- 4 Architecture Virtuelle
- 5 Conclusion

### Présentation du problème



- $\triangleright$  Source :  $S(x_s, y_s)$
- ► Destination =  $\zeta(D(x_d, y_d), R) \Rightarrow$  Multicast group (MG)
- ▶  $Z \in Destination ssi d(Z, D) \leq R$



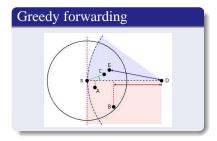
# Formulation du problème

► Transmission unicast de  $S \Rightarrow D$ 



### Formulation du problème

- ► Transmission unicast de  $S \Rightarrow D$
- Basée sur deux approches de routage géographique

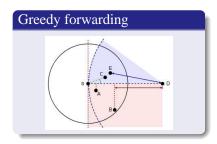


- DIR [Kranakis et al 1999]
- MF [Kleinrock et al 1984]
- NFP [Li et al 1986]

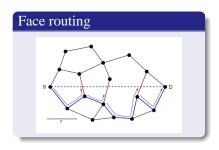


### Formulation du problème

- ► Transmission unicast de  $S \Rightarrow D$
- Basée sur deux approches de routage géographique



- DIR [Kranakis et al 1999]
- MF [Kleinrock et al 1984]
- NFP [Li et al 1986]



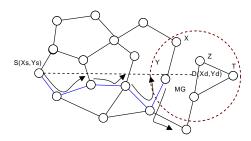
- [Bose et al 1999]

### Approches avec garantie de livraison

- ▶ Basée sur face routing
  - Greedy-Face Greedy [Bose,Stojmenovic 1999]
  - GPSR [Karp, Kung 2000]

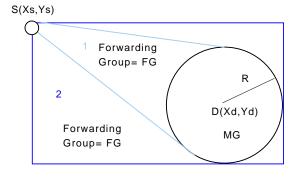
\_ ...

► Necéssite un graphe planaire



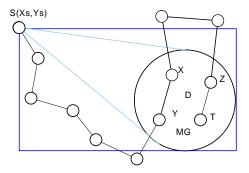
### Approches sans garantie de livraison

- Basagni [1999]
- ▶ Vaidya [1999]
- ► Papavassiliou [2003]



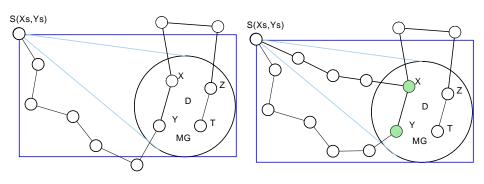
Modèle

### Approches sans garantie de livraison



Échec de route

# Approches sans garantie de livraison





Livraison partielle

Échec de route

### Contribution

### Proposer une approche

- o Nécessitant pas de graphe planaire
- o Améliorant progressivement le taux de réussite dans le MG



### Contribution

### Utiliser DSR

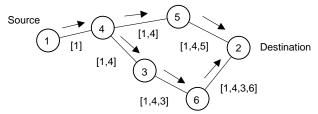
- o De nouveaux messages de contrôle
- o Modifier les procédures de découvertes de route



# Philosophie de DSR [Dynamic Source Rounting]

Initiator Id Tai	rget Id List of r	odes
------------------	-------------------	------

### Route Request

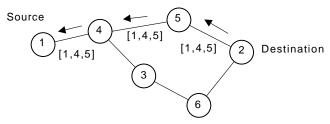


Enregistrement du chemin

# Philosophie de DSR [Dynamic Source Rounting]

Target Id	Initiator Id	List of nodes
-----------	--------------	---------------

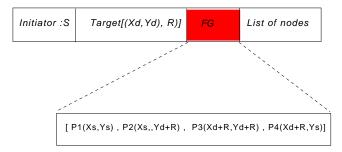
### Route Reply



Renvoi du chemin

### Procédures de découverte de routes

► Format de paquets Route Request :



► Format de paquets Route Reply : Reste identique [DSR]

# Messages de contrôle

► Position\_Echange\_msg:



- ∘ Collecter des voisins potentiels  $\Rightarrow \exists ?u/u \in FG$
- Envoyer Route Request aux voisins potentiels

# Messages de contrôle

► Position\_Echange\_msg :



- Collecter des voisins potentiels  $\Rightarrow \exists ?u/u \in FG$
- Envoyer Route Request aux voisins potentiels
- ► Impass\_msg:

Emetteur	Destinataire	Status
----------	--------------	--------

Notifier un échec de route (FG?,MG?)

Espace de déploiement de taille : L x l

### La source S définie

∘ Multicat Group  $\Rightarrow$  MG=  $\zeta(D(X_d, Y_d), R)$ 



► Espace de déploiement de taille : L x l

#### La source S définie

- Multicat Group  $\Rightarrow$  MG=  $\zeta(D(X_d, Y_d), R)$
- Forwarging Group  $\Rightarrow FG = P1\binom{x}{y}, P2\binom{x}{y}, P3\binom{x}{y}, P4\binom{x}{y}$ L' = dist(P1, P4) et l' = dist(P1, P2)

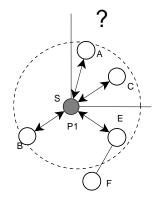


► Espace de déploiement de taille : L x l

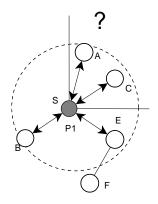
#### La source S définie

- Multicat Group  $\Rightarrow$  MG=  $\zeta(D(X_d, Y_d), R)$
- Forwarging Group  $\Rightarrow FG = P1\binom{x}{y}, P2\binom{x}{y}, P3\binom{x}{y}, P4\binom{x}{y}$ L' = dist(P1, P4) et l' = dist(P1, P2)
- Deux paramètres de dilatation  $d_1$  et  $d_2 \Rightarrow d_1 = \frac{L-L'}{n}$ ,  $d_2 = \frac{l-l'}{n}$

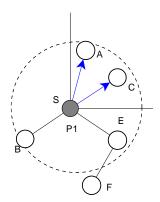
#### Diffusion Position\_Echange\_msg

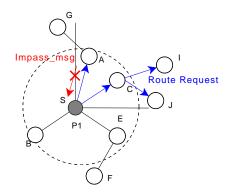


Diffusion Position\_Echange\_msg

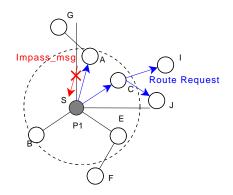


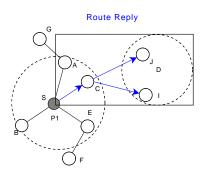
#### **Route Request**

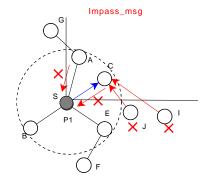






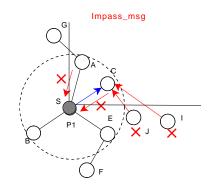




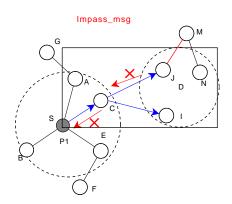


Échec de route en C





Échec de route en C



Échec de route en J

- ► Échec de route à la source :
  - o Ré-adapter le Forwarging Group

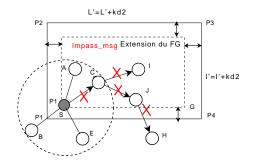
$$FG = P1\left(\frac{x - \frac{d_1}{2}}{y - \frac{d_2}{2}}\right), P2\left(\frac{x - \frac{d_1}{2}}{y + \frac{d_2}{2}}\right), P3\left(\frac{x + \frac{d_1}{2}}{y + \frac{d_2}{2}}\right), P4\left(\frac{x + \frac{d_1}{2}}{y - \frac{d_2}{2}}\right)$$



- ► Échec de route à la source :
  - o Ré-adapter le Forwarging Group

$$FG = P1\binom{x - \frac{d_1}{2}}{y - \frac{d_2}{2}}, P2\binom{x - \frac{d_1}{2}}{y + \frac{d_2}{2}}, P3\binom{x + \frac{d_1}{2}}{y + \frac{d_2}{2}}, P4\binom{x + \frac{d_1}{2}}{y - \frac{d_2}{2}}$$

o Relancer la procédure de découverte

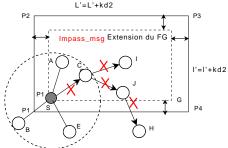




- ► Échec de route à la source :
  - o Ré-adapter le Forwarging Group

$$FG = P1\binom{x - \frac{d_1}{2}}{y - \frac{d_2}{2}}, P2\binom{x - \frac{d_1}{2}}{y + \frac{d_2}{2}}, P3\binom{x + \frac{d_1}{2}}{y + \frac{d_2}{2}}, P4\binom{x + \frac{d_1}{2}}{y - \frac{d_2}{2}}$$

o Relancer la procédure de découverte

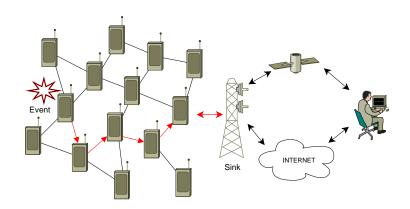


la  $n^{ime}$  procédure  $(k = n) \Rightarrow$  Flooding



- I Introduction et Problématique
- 2 Partitionnement
- 3 Geocasting
- 4 Architecture Virtuelle
- 5 Conclusion

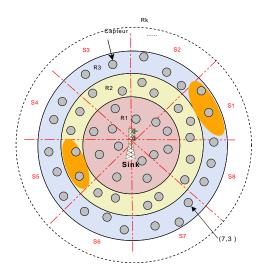
# Réseaux de Capteurs



### Qu'est que c'est?

Consiste à construire par dessus du réseau une structure virtuelle avec un ensemble de fonctionnalité associées.

- ➤ Structure : Organiser les entités et faciliter leurs interactions
- ▶ Fonctionnalités : Maintenir la structure et faciliter son utilisation



### Sink

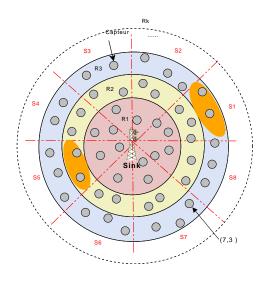
► T. omnidirectionnelles

$$\circ R_1 < R_2 \ldots < R_k$$

► T. directionnelles

$$\circ S_1(R_k,\alpha)..S_m(R_k,\alpha)$$

- $ightharpoonup R_x \cap S_y \Rightarrow \text{cluster}(x,y)$
- $k = 2^i$  et  $m = 2^j$



### Sink

► T. omnidirectionnelles

$$\circ$$
  $R_1 < R_2 \ldots < R_k$ 

► T. directionnelles

$$\circ$$
  $S_1(R_k,\alpha)..S_m(R_k,\alpha)$ 

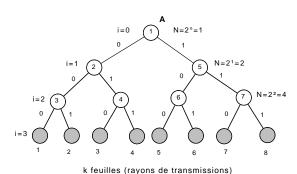
- $ightharpoonup R_x \cap S_y \Rightarrow \text{cluster}(x,y)$
- $k = 2^i$  et  $m = 2^j$

### Capteur doit lire

► l'identité (*x*, *y*) de son cluster

# Apprentissage des coordonnées[k=8]

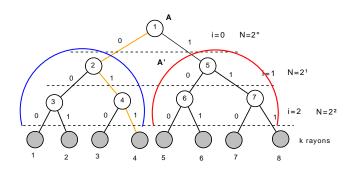
$R_{x}$	$b_1 b_2 b_3$
$R_1$	000
$R_2$	001
$R_3$	010
$R_4$	011
$R_5$	100
$R_6$	101
$R_7$	110
$R_8$	111



Les  $R_x$ 

# Apprentissage des coordonnées[k=8]

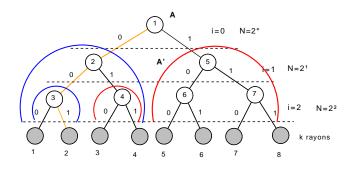
$R_{x}$	$b_1 b_2 b_3$
$R_1$	000
$R_2$	001
$R_3$	010
$R_4$	011
$R_5$	100
$R_6$	<b>1</b> 01
$R_7$	<b>1</b> 10
$R_8$	<b>1</b> 11

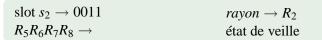


slot  $s_1 \rightarrow 00001111$   $rayon \rightarrow R_4$ 

# Apprentissage des coordonnées[k=8]

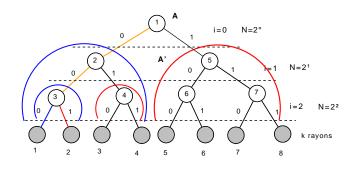
$R_{x}$	$b_1b_2b_3$
$R_1$	000
$R_2$	001
$R_3$	010
$R_4$	011
$R_5$	100
$R_6$	101
$R_7$	110
$R_8$	111





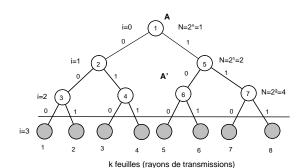
# Apprentissage des coordonnées[k=8]

$R_{x}$	binaire
$R_1$	000
$R_2$	001
$R_3$	010
$R_4$	011
$R_5$	100
$R_6$	101
$R_7$	110
$R_8$	111



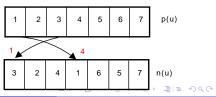
$$\begin{array}{ccc} \operatorname{slot} s_3 \to 01 & \operatorname{rayon} \to R_1 \\ \operatorname{slot} s_4 \to 01 & \operatorname{rayon} \to R_3 \end{array}$$

# Choix des slots coté capteurs



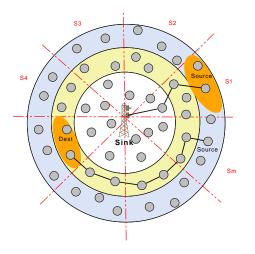
### Méthode analytique

- Soient A et A'
- slot= $1 + \sum_{j=1}^{log(k)} b_j 2^{log(k)-j}$
- $\circ$   $n(slot) \Rightarrow rayon$



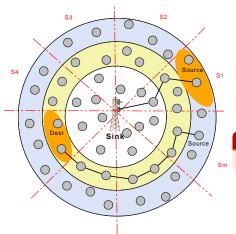
numéro

# Pour quel but?



- Routage centralisé (dans le secteur)
- Routage distribué (externe)

# Pour quel but?



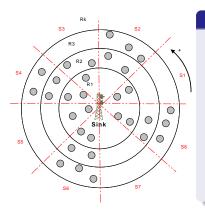
- Routage centralisé (dans le secteur)
- o Routage distribué (externe)

#### Problème

Clusters vides

Proposer un algorithme permettant au Sink de repérer l'ensemble des clusters vides

### Découvertes des clusters vides



#### Secteurs Rayons $\overline{R}_2$ $R_i$ $\overline{R_k}$ $R_1$ 0 0 0 0 0 0 0 0 0 0

Table des Clusters

0

0 0

 $S_1$ 

 $S_2$ 

 $S_i$ 

0 0

0

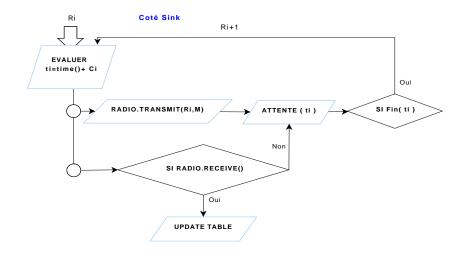
### Découvertes des clusters vides

#### On suppose

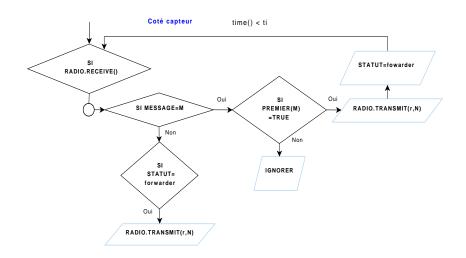
- $\triangleright$  Durée  $t_i$  suffisamment choisi :
  - pour chaque  $R_i$ 
    - $\circ$  Sink  $\Rightarrow R_i$
    - $\circ$   $R_i \Rightarrow Sink$
    - Forwarder (pour  $R_i > R_j$ )
- ▶ Primitives :
  - o radio.transmit(r/R, N/M)
  - o radio.receive()



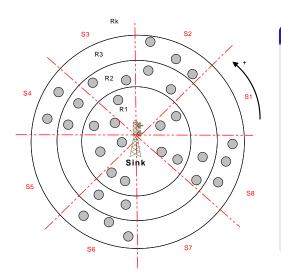
### Découverte des clusters vides



### Découverte des clusters vides



### Découvertes des clusters vides



#### Table des Clusters

Secteurs	Rayons		
	$R_1$	$R_2$	$R_3$
$S_1$	1	0	0
$S_2$	0	1	1
$egin{array}{c} S_2 \ S_3 \ S_4 \ \end{array}$	1	1	0
	1	1	1
$S_5$ $S_6$ $S_7$ $S_8$	1	0	0
$S_6$	1	1	1
$S_7$	0	0	0
$S_8$	1	1	1

### Choix des routes

#### Table des Clusters

Secteurs	Rayons			
	$R_1$	$R_2$	$R_3$	
$S_1$	1	0	0	
$S_2$	0	1	1	
$S_3$	1	1	0	
$S_4$	1	1	1	
$S_5$	1	0	0	
$S_6$	1	1	1	
S <sub>3</sub> S <sub>4</sub> S <sub>5</sub> S <sub>6</sub> S <sub>7</sub> S <sub>8</sub>	0	0	0	
$S_8$	1	1	1	

- Choisir un cluster pour une rêquete donnée
- Choisir un modèle de communication
  - o centralisé (routage interne)
  - o distribué (routage externe)

Rêquete  $r : (Clu(R_3, S_2), \{Clu(R_2, S_2), Clu(R_2, S_3), Clu(R_1, S_3)\})$ 

- I Introduction et Problématique
- 2 Partitionnement
- 3 Geocasting
- 4 Architecture Virtuelle
- 5 Conclusion



### Conclusions

► Clusterisation dans les réseaux mobiles ad hoc basée sur des paramètres qui reflètent l'état du réseau



#### **Conclusions**

- ► Clusterisation dans les réseaux mobiles ad hoc basée sur des paramètres qui reflètent l'état du réseau
- ► Geocasting avec garantie de livraison basé sur un algorithme classique de découverte de route sans inonder le réseau de hello paquet

### **Conclusions**

- Clusterisation dans les réseaux mobiles ad hoc basée sur des paramètres qui reflètent l'état du réseau
- ► Geocasting avec garantie de livraison basé sur un algorithme classique de découverte de route sans inonder le réseau de hello paquet
- ▶ Une architure virtuelle qui favorisant l'aide au routage et la fusion des données.

▶ Mesurer les performances de notre algorithme de geocast au sein de la région Geocast (MG)



- ▶ Mesurer les performances de notre algorithme de geocast au sein de la région Geocast (MG)
- Clusterisation appliquée au problème d'initialisation



- ► Mesurer les performances de notre algorithme de geocast au sein de la région Geocast (MG)
- Clusterisation appliquée au problème d'initialisation
- ► Introduction de l'économie d'énergie dans nos protocoles

- ► Mesurer les performances de notre algorithme de geocast au sein de la région Geocast (MG)
- ► Clusterisation appliquée au problème d'initialisation
- ▶ Introduction de l'économie d'énergie dans nos protocoles
- ► Tolérances aux pannes de nos protocoles

Annexe



### P.Théorème1

#### Preuve du Théorème1

$$\begin{aligned} & p = \frac{Card(\psi)}{Card(\varphi)} = \frac{m(m-1)(m-2)\dots m-r+1)}{m^r} \\ & = (1-\frac{1}{m)}(1-\frac{2}{m})\dots(1-\frac{(r-1)}{m}) \\ & = (1-\frac{2}{m}-\frac{1}{m}+\frac{1}{m^2})\dots(1-\frac{(r-1)}{m}) \approx 1-\frac{(1+2+\dots r-1)}{m} \\ & \text{Si r est petit alors } p = 1-\frac{r(r-1)}{2m} \\ & \text{Si r est grand alors } p = \log(p) = \log(1-\frac{r(r-1)}{2m}) \approx \log(-\frac{r(r-1)}{2m}) \\ & \approx \exp(-\frac{r(r-1)}{2m}) > \exp(-\Theta(n^2/N^3)) \text{ en posant } r = n \text{ et } m = N^3 \end{aligned}$$