



**HAL**  
open science

# Une solution tolérante aux délais pour des applications de localisation et de traçabilité a posteriori en milieux confinés

Mohamed Chakib Baouche

► **To cite this version:**

Mohamed Chakib Baouche. Une solution tolérante aux délais pour des applications de localisation et de traçabilité a posteriori en milieux confinés. Autre. Université Blaise Pascal - Clermont-Ferrand II, 2012. Français. NNT : 2012CLF22261 . tel-00857206

**HAL Id: tel-00857206**

**<https://theses.hal.science/tel-00857206>**

Submitted on 12 Sep 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° D'ORDRE : D.U : 2261  
E D S P I C : 567

Année 2012

UNIVERSITÉ BLAISE PASCAL

ECOLE DOCTORALE SCIENCES  
POUR L'INGÉNIEUR DE CLERMONT-FERRAND

Laboratoire d'Informatique,  
de Modélisation et d'Optimisation des Systèmes

THÈSE

Pour obtenir le grade de  
DOCTEUR DE L'UNIVERSITÉ

Discipline : Informatique

Présentée et soutenue publiquement par

Chakib BAOUCHE

**Une solution tolérante aux délais pour des applications de  
localisation et de traçabilité a posteriori en milieux  
confinés**

**Soutenu publiquement le 12 Juillet 2012 devant le jury :**

|                      |                                    |   |
|----------------------|------------------------------------|---|
| <b>Rapporteurs :</b> | <b>M<sup>r</sup>. RONDEAU E.</b>   | <b>Professeur à l'Université de Lorraine</b>      |
|                      | <b>M<sup>r</sup>. SPIES F.</b>     | <b>Professeur à l'Université de Franche-Comté</b> |
| <b>Directeurs :</b>  | <b>M<sup>r</sup>. MISSON M.</b>    | <b>Professeur à l'Université de Clermont I</b>    |
|                      | <b>M<sup>r</sup>. FREITAS A.</b>   | <b>MCF à l'Université de Clermont I</b>           |
| <b>Examineurs :</b>  | <b>M<sup>r</sup>. BARKAOUI K.</b>  | <b>Professeur au CNAM de Paris</b>                |
|                      | <b>M<sup>me</sup> SERVAJEAN M.</b> | <b>MCF à l'Université de Clermont I</b>           |
| <b>Invité :</b>      | <b>M<sup>r</sup>. BLANC G.</b>     | <b>Manager ExoTIC Systems</b>                     |



*À ma famille*

## REMERCIEMENTS

Cette thèse a été réalisée au sein de l'équipe Réseaux et Protocoles du Laboratoire d'Informatique LIMOS de Modélisation et d'Optimisation des Systèmes de l'université Blaise Pascal - Clermont-Ferrand II. Cet aboutissement n'aurait pas été possible sans l'aide de plusieurs personnes. Qu'elles puissent trouver sur cette page l'expression de ma profonde et sincère gratitude.

Mes premiers remerciements vont tout d'abord à mon directeur de thèse, Monsieur Michel MISSON, Professeur à l'Université Clermont I. Je le remercie vivement pour sa confiance, sa disponibilité, sa patience et son encadrement précieux qui ont permis de mener à bien ce travail. Je salue aussi sa rigueur scientifique ainsi que ses grandes valeurs humaines.

Mes profonds remerciements s'adressent aussi à mes encadrants Monsieur Antonio FREITAS, Maître de conférences à l'Université Clermont I, et à Madame Marie-Françoise SERVAJEAN, Maître de conférences à l'Université Clermont I pour avoir guidé et orienté ce travail, ainsi que pour leur soutien, leur écoute et tous leurs bons conseils au cours de ces années.

Très honoré, je remercie sincèrement M. Eric RONDEAU, Professeur à l'Université de Lorraine et M. François SPIES, Professeur à l'Université de Franche-Comté pour l'intérêt qu'ils ont bien voulu porter à ce travail en acceptant d'en être les rapporteurs.

Sans oublier tous mes collègues du bureau, je les remercie pour leurs conseils, leur sympathie et leur bonne humeur, merci à Alexandre, François, Fred, Hervé, Ismail (pour son email), Lionel, Malik, Nicolas sans oublier bien sûr l'inimitable Sabri.

Thank you so much Gérard Chalhoub, for your help during my thesis and your contribution for improving my English ☺.

Enfin, je dédie ce travail à toutes les personnes chères à mes yeux, mes amis et ma famille, en particulier à ma mère Nadjia, à mon père Messaoud, à mes frères Amine, Nassim, Raouf, Toufik ainsi qu'à ma tendre épouse Narimane.

Cette liste de remerciements est non exhaustive, que tous ceux qui ont contribué à faciliter cette étude de près ou de loin trouvent ici le témoignage de ma sincère gratitude.

*Chakib*

## Table des matières

|   |    |
|---|----|
| Introduction .....  | 12 |
| Chapitre 1 : Etat de l'art .....                                      | 15 |
| 1.1. Introduction .....   | 15 |
| 1.2. Les réseaux de capteurs sans fil .....                           | 15 |
| 1.2.1. Définition .....   | 15 |
| 1.2.2. Applications et scenarii .....                                 | 16 |
| 1.2.2.1. Applications militaires .....                                | 17 |
| 1.2.2.2. Applications de sécurité .....                               | 17 |
| 1.2.2.3. Applications environnementales .....                         | 17 |
| 1.2.2.4. Applications médicales .....                                 | 17 |
| 1.2.2.5. Applications industrielles et commerciales .....             | 18 |
| 1.2.2.6. Applications de localisation et de traçabilité .....         | 18 |
| 1.2.3. Problématiques et défis .....                                  | 18 |
| 1.2.3.1. Economie d'énergie .....                                     | 18 |
| 1.2.3.2. Routage .....  | 19 |
| 1.2.3.3. Qualité de service .....                                     | 19 |
| 1.2.3.4. Auto-configuration .....                                     | 19 |
| 1.2.3.5. Sécurité .....   | 19 |
| 1.2.3.6. Mobilité .....   | 20 |
| 1.2.3.7. Localisation .....   | 20 |
| 1.2.4. La synchronisation dans les réseaux de capteurs sans fil ..... | 20 |
| 1.3. La localisation dans les réseaux de capteurs sans fil .....      | 23 |
| 1.3.1. Méthodes de localisation physique .....                        | 24 |
| 1.3.1.1. Trilatération par calcul de temps de transit du signal ..... | 24 |
| 1.3.1.2. Trilatération par mesure de puissance du signal .....        | 26 |
| 1.3.2. Méthodes de localisation logique .....                         | 26 |
| 1.7. Les couches basses des réseaux de capteurs sans fil .....        | 29 |
| 1.7.1. Caractéristiques .....   | 29 |
| 1.7.2. Couche physique IEEE 802.15.4 .....                            | 31 |
| 1.7.3. Couche MAC IEEE 802.15.4 .....                                 | 32 |
| 1.7.3.1. L'accès au médium .....                                      | 33 |
| 1.4. Les réseaux tolérants aux délais .....                           | 38 |
| 1.4.1. Définition .....   | 38 |

|   |    |
|---|----|
| 1.4.2. Caractéristiques des DTN.....  | 39 |
| 1.4.3. Applications des réseaux tolérants aux délais.....                           | 39 |
| 1.5 Le transport de l'information dans un réseau à connectivité intermittente.....  | 41 |
| 1.5.1. Les méthodes basées sur la connaissance du réseau.....                       | 41 |
| 1.5.1.1. Epidemic Routing.....  | 42 |
| 1.5.1.2. Controlled flooding Spray and Wait.....                                    | 42 |
| 1.5.1.3. Deleting obsolete messages.....  | 43 |
| 1.5.1.4. Packet dropping policy.....  | 44 |
| 1.5.2. Les méthodes basées sur des nœuds additionnels.....                          | 45 |
| 1.5.2.1. Data MULE.....   | 45 |
| 1.5.2.2. Message Ferry.....   | 47 |
| 1.5.2.3. Throwboxes.....  | 48 |
| 1.6. Les modèles de déplacement.....  | 49 |
| 1.6.1. Les modèles analytiques.....   | 49 |
| 1.6.1.1. Les modèles individuels.....   | 49 |
| 1.6.1.2. Les modèles de groupe.....   | 52 |
| 1.6.2. Les modèles basés sur des traces.....  | 55 |
| 1.6.2.1. ZebraNet.....  | 55 |
| 1.6.2.2. DakNet.....  | 55 |
| 1.6.2.3. Cambridge/Haggle.....  | 55 |
| 1.6.2.4. The One.....   | 55 |
| 1.6.2.5. The SUVNet.....  | 56 |
| 1.6.2.6. VehILux.....   | 56 |
| 1.8. Conclusion.....  | 56 |
| Chapitre 2 : Colportage de connaissance dans une population mobile.....             | 58 |
| 2.1. Introduction.....  | 58 |
| 2.2. Le concept du colportage dans une population mobile.....                       | 58 |
| 2.2.1. Terminologie.....  | 61 |
| 2.2.2. Création de l'information élémentaire.....                                   | 62 |
| 2.2.3. Processus de colportage.....   | 63 |
| 2.2.3.1. Principe de colportage appliqué à une étude biologique.....                | 66 |
| 2.2.3.2. Principe de colportage appliqué à un domaine médical.....                  | 67 |
| 2.2.3.3. Principe de colportage appliqué à un domaine social.....                   | 68 |
| 2.2.3.4. Principe de colportage appliqué à la localisation en milieux confinés..... | 69 |
| 2.3. Modèle générique.....  | 70 |
| 2.3.1. Orientations fonctionnelles d'un système de colportage.....                  | 71 |
| 2.3.1.1. Le choix du domaine applicatif.....  | 72 |

|  |     |
|--|-----|
| 2.3.1.2. La construction du système de colportage .....                            | 74  |
| 2.3.1.3. L'analyse des données collectées .....                                    | 75  |
| 2.3.1.4. Cas d'utilisation pour une application de localisation a posteriori ..... | 76  |
| 2.3.1.5. Représentation en couche du modèle de colportage .....                    | 78  |
| 2.3.2. Modèle du domaine .....   | 80  |
| 2.3.3. Diagramme de séquence.....  | 82  |
| 2.4. Conclusion.....   | 84  |
| Chapitre 3 : LoWCA (Exemple d'application cible) .....                             | 85  |
| 3.1. Introduction .....  | 85  |
| 3.2. Entités du réseau.....  | 85  |
| 3.2.1. Les nœuds mobiles .....   | 85  |
| 3.2.2. Les nœuds fixes .....   | 86  |
| 3.2.3. Les collecteurs.....  | 86  |
| 3.3. Connaissance et information atomique .....                                    | 87  |
| 3.3.1. Événement de contact complet.....   | 89  |
| 3.3.2. Événement de contact incomplet.....   | 90  |
| 3.4. Stratégies de diffusion.....  | 90  |
| 3.4.1. Mécanisme d'écho.....   | 90  |
| 3.4.1.1. Le problème du « Rendez-vous manqué ».....                                | 91  |
| 3.4.1.2. Piste d'implémentation.....   | 93  |
| 3.4.2. Notion d'état .....   | 99  |
| 3.4.3. Format d'une trame de diffusion .....                                       | 103 |
| 3.4.4. Algorithme de transmission de trames (diagramme de transmission) .....      | 104 |
| 3.4.5. Algorithme de réception de trames (diagramme de réception).....             | 106 |
| 3.5. Mécanisme de routage.....   | 108 |
| 3.5.1. Stratégies de filtrage.....   | 108 |
| 3.5.2. Liste des filtres .....   | 111 |
| 3.5.2.1 Filtres permanents (anti-redondance).....                                  | 111 |
| 3.5.2.2 Filtres de gestion de mémoire .....  | 114 |
| 3.5.2.3 Filtres applicatifs .....  | 117 |
| 3.6. Synchronisation.....  | 125 |
| Chapitre 4 : Evaluation de LoWCA .....   | 127 |
| 4.1. Introduction .....  | 127 |
| 4.2. Simulation .....  | 127 |
| 4.2.1. Les modèles de propagation.....   | 128 |
| 4.2.1.1. Le modèle Free Space .....  | 128 |
| 4.2.1.2. Le modèle Two-Ray ground .....  | 129 |



|  |     |
|--|-----|
| 4.2.1.3. Le modèle Shadowing.....                    | 130 |
| 4.2.1.4. Le modèle ITU-R P1238-4.....                | 130 |
| 4.3. Evaluation de LoWCA .....                       | 131 |
| 4.3.1. Evaluation du comportement d'une cellule..... | 131 |
| 4.3.1.1. Topologie et conditions de simulations..... | 131 |
| 4.3.1.2. Détection de proximité.....                 | 132 |
| 4.3.1.3. Capacité d'une cellule.....                 | 139 |
| 4.3.2. Evaluation du mécanisme de routage.....       | 148 |
| 4.3.2.1. Topologie et conditions de simulation ..... | 148 |
| 4.3.2.2. Effets des filtres.....                     | 154 |
| 4.3.2.3. Effets de la taille mémoire .....           | 159 |
| 4.4. Validation par maquettage .....                 | 165 |
| 4.4.1. Implémentation sur des cartes TelosB .....    | 166 |
| 4.4.1.1. Scenarii des tests .....                    | 168 |
| 4.4.1.2. Validation du mécanisme de colportage ..... | 168 |
| 4.4.2. Application graphique .....                   | 170 |
| 4.5. Conclusions .....                               | 172 |
| Conclusions et perspectives .....                    | 173 |
| 5.1. Contribution de la thèse.....                   | 173 |
| 5.2. Perspectives.....                               | 174 |
| Liste des publications .....                         | 176 |
| Bibliographie.....                                   | 177 |

## Table des figures

|   |    |
|---|----|
| Figure 1 : Exemple d'un réseau de capteurs sans fil.....  | 16 |
| Figure 2 : Synchronisation de type maître-esclave et point-à-point.....                           | 22 |
| Figure 3 : Localisation par trilatération.....  | 25 |
| Figure 4 : Localisation par intersection des signaux.....   | 27 |
| Figure 5 : Localisation par coordonnées virtuelles.....   | 28 |
| Figure 6 : Pile protocolaire IEEE 802.15.4/ZigBee.....  | 30 |
| Figure 7 : Exemple de topologie en étoile et de topologie point à point.....                      | 31 |
| Figure 8 : Structure d'une supertrame 802.15.4.....   | 33 |
| Figure 9 : Algorithme CSMA/CA slotté.....   | 35 |
| Figure 10 : Algorithme CSMA/CA non slotté.....  | 37 |
| Figure 11 : Exemple d'une communication DTN.....  | 38 |
| Figure 12 : Des étudiants équipant des zèbres avec des dispositifs GPS.....                       | 40 |
| Figure 13 : Concept DakNet.....   | 41 |
| Figure 14 : Exemple de routage épidémique.....  | 42 |
| Figure 15 : Routage épidémique en deux phases.....  | 43 |
| Figure 16 : Architecture trois étages dite data MULE.....   | 46 |
| Figure 17 : Déplacement aléatoire d'une mule dans un champ de capteurs.....                       | 47 |
| Figure 18 : Exemple d'un DTN utilisant des throwboxes.....  | 48 |
| Figure 19 : Exemple d'un déplacement aléatoire <i>Random Direction</i> .....                      | 51 |
| Figure 20 : Exemple d'un déplacement aléatoire <i>Restricted Random Waypoint</i> .....            | 51 |
| Figure 21 : Exemple d'un déplacement aléatoire <i>Boundless Simulation Area</i> .....             | 52 |
| Figure 22 : Mouvement d'un groupe de mobiles avec la méthode Column.....                          | 53 |
| Figure 23 : Mouvement de sept mobiles utilisant le modèle nomade.....                             | 54 |
| Figure 24 : Transmission de proche en proche.....   | 59 |
| Figure 25 : Transmission d'un virus entre deux entités.....                                       | 60 |
| Figure 26 : Caractéristique d'un événement de contact au niveau signal.....                       | 63 |
| Figure 27 : Structure fonctionnelle d'un événement de contact.....                                | 63 |
| Figure 28 : Principe du colportage dans une population mobile.....                                | 65 |
| Figure 29 : Principe du colportage appliqué à l'étude des déplacements d'espèce animale....       | 66 |
| Figure 30 : Principe du colportage appliqué au suivi d'agents contaminants.....                   | 67 |
| Figure 31 : Principe du colportage appliqué à l'étude de l'attractivité d'une exposition.....     | 68 |
| Figure 32 : Principe du colportage appliqué à la localisation dans des milieux confinés.....      | 69 |
| Figure 33 : Diagramme de cas d'utilisation pour le choix du domaine applicatif.....               | 72 |
| Figure 34 : Diagramme de cas d'utilisation pour la construction du système de colportage...       | 74 |
| Figure 35 : Diagramme de cas d'utilisation pour l'analyse des données collectées.....             | 76 |
| Figure 36 : Exemple d'un cas d'utilisation pour une application de localisation a posteriori..... | 77 |
| Figure 37 : Représentation en couche du processus de colportage.....                              | 79 |
| Figure 38 : Diagramme de classes d'un système de colportage.....                                  | 80 |
| Figure 39 : Scénario d'échange.....   | 82 |
| Figure 40 : Diagramme de séquence pour l'échange de connaissance.....                             | 83 |
| Figure 41 : Représentation générique d'une galerie minière.....                                   | 87 |
| Figure 42 : Format générique d'un événement de contact.....                                       | 88 |
| Figure 43 : Exemple de création d'événements de contact complets.....                             | 89 |
| Figure 44 : Echange de contact incomplet.....   | 90 |
| Figure 45 : Problème du rendez-vous manqué.....   | 92 |

|  |     |
|--|-----|
| Figure 46 : Fonctionnement du système d'écho.....  | 92  |
| Figure 47 : Structure d'une table des nouveaux entrants. ....  | 93  |
| Figure 48 : Scénario d'usage du système d'écho. ....   | 95  |
| Figure 49 : Structure d'une table des voisins. ....  | 95  |
| Figure 50 : Scénario d'usage de la table des voisins.....  | 97  |
| Figure 51 : Structure d'une table de connaissance.....   | 98  |
| Figure 52 : Automate d'états pour la gestion des nœuds.....  | 99  |
| Figure 53 : Exemple d'évolution de l'état des mobiles et de leur connaissance.....                   | 102 |
| Figure 54 : format d'une trame de diffusion. ....  | 103 |
| Figure 55 : Algorithme de transmission de trame.....   | 104 |
| Figure 56 : Algorithme de réception de trame. ....   | 106 |
| Figure 57 : Fonctionnement du filtrage des événements de contact [75].....                           | 110 |
| Figure 58 : Succession de contacts lors d'un aller-retour. ....                                      | 112 |
| Figure 59 : Fonctionnement du filtre concaténation.....  | 113 |
| Figure 60 : Exemple d'une collecte indirecte d'informations concernant des nœuds non rencontrés..... | 118 |
| Figure 61 : Exemple de fonctionnement du filtre de localisation.....                                 | 121 |
| Figure 62 : Exemple de fonctionnement du filtre traçabilité aléatoire.....                           | 123 |
| Figure 63 : Datation des événements de contact lors d'un échange entre nœuds.....                    | 126 |
| Figure 64 : Modèle Free Space. ....  | 128 |
| Figure 65 : Modèle Two-Ray ground.....   | 129 |
| Figure 66 : Configuration pour la détection de proximité.....  | 132 |
| Figure 67 : Durée d'un événement de contact.....   | 133 |
| Figure 68 : Fausse détection de rupture de contact.....  | 134 |
| Figure 69 : Effets de la densité des mobiles avec 5 mobiles.....                                     | 134 |
| Figure 70 : Effets de la densité des mobiles avec 10 mobiles.....                                    | 135 |
| Figure 71 : Effets de la densité des mobiles avec 100 mobiles.....                                   | 135 |
| Figure 72 : Effets de la densité des mobiles avec 200 mobiles.....                                   | 136 |
| Figure 73 : Effets de 100 mobiles. ....  | 137 |
| Figure 74 : Effets de 200 mobiles. ....  | 137 |
| Figure 75 : Pourcentage des fausses détections de rupture de contact.....                            | 138 |
| Figure 76 : Trafic moyen en fonction de la charge soumise : cas des trames longues. ....             | 141 |
| Figure 77 : Trafic moyen en fonction de la charge soumise : cas des trames courtes. ....             | 142 |
| Figure 78 : Trafic moyen en fonction de la charge soumise : cas des trames moyennes.....             | 142 |
| Figure 79 : Trafic moyen en fonction de la charge soumise : cas d'une transmission hybride.....      | 143 |
| Figure 80 : Trafic moyen pour des trames longues.....  | 144 |
| Figure 81 : Influence de la longueur des trames sur le nombre de drops.....                          | 144 |
| Figure 82 : Nombre de collision en fonction de la charge offerte soumise à la couche physique.....   | 145 |
| Figure 83 : Nombre de trames perdues en fonction de la charge offerte.....                           | 146 |
| Figure 84 : Evaluation de la charge d'une cellule.....   | 147 |
| Figure 85 : Représentation réaliste d'une partie d'une mine. ....                                    | 148 |
| Figure 86 : Plan des trois étages du bâtiment INFO-R&T de l'IUT de Clermont-Ferrand. ..              | 149 |
| Figure 87 : Déploiement des nœuds dans le couloir d'un bâtiment.....                                 | 150 |
| Figure 88 : Trajectoire d'un mobile entre deux jonctions.....  | 151 |
| Figure 89 : Représentation d'une galerie minière avec trajectoires de nœuds mobiles.....             | 152 |
| Figure 90 : Nombre de contacts pour le filtre permanent.....   | 155 |
| Figure 91 : Distribution des contacts pour les filtres permanents.....                               | 156 |
| Figure 92 : Nombre de contact pour le filtre aléatoire.....  | 157 |

|  |     |
|--|-----|
| Figure 93 : Distribution des contacts pour le filtre aléatoire.....  | 157 |
| Figure 94 : Nombre de contact pour le filtre avantage balise. ....   | 158 |
| Figure 95 : Distribution des contacts pour le filtre avantage balise. ....   | 158 |
| Figure 96 : Effet de la taille mémoire pour le filtre permanent.....   | 160 |
| Figure 97 : Distribution des contacts pour le filtre permanent et pour une capacité de 15 places. ....               | 160 |
| Figure 98 : Distribution des contacts pour le filtre permanent et pour une capacité de 100 places. ....              | 161 |
| Figure 99: Effet de la taille mémoire pour le filtre aléatoire.....  | 162 |
| Figure 100 : Distribution des contacts pour le filtre aléatoire et pour une capacité de 15 places .....              | 162 |
| Figure 101: Distribution des contacts pour le filtre aléatoire et pour une capacité de 100 places. ....              | 163 |
| Figure 102: Effet de la taille mémoire pour le filtre avantage balise.....   | 164 |
| Figure 103: Distribution des contacts pour le filtre avantage balise et pour une capacité mémoire de 15 places.....  | 164 |
| Figure 104: Distribution des contacts pour le filtre avantage balise et pour une capacité mémoire de 100 places..... | 165 |
| Figure 105: Architecture d'une plateforme <i>TelosB</i> .....  | 166 |
| Figure 106: Présentation d'une plateforme <i>TelosB</i> .....  | 167 |
| Figure 107: Simulation et expérimentation pour le filtre permanent.....  | 169 |
| Figure 108 : Simulation et expérimentation pour le filtre aléatoire.....   | 169 |
| Figure 109: Simulation et expérimentation pour le filtre avantage balise.....  | 170 |
| Figure 110 : Exemple d'une animation graphique.....  | 171 |

## Liste des tableaux

|   |     |
|---|-----|
| Tableau 1 : Paramètres de simulation pour la détection de proximité.....                                  | 131 |
| Tableau 2 : Paramètres de simulation pour l'évaluation de la capacité d'une cellule.....                  | 139 |
| Tableau 3 : Nombre maximum de mobiles dans une cellule.....   | 147 |
| Tableau 4 : Liste des filtres avec $CEa : Idx, Idy, NSa, T1, T2$ et $CEb : Idy, Idx, NSb, T'1, T'2$ ..... | 153 |
| Tableau 5 : Paramètres de simulation pour l'évaluation du mécanisme de routage.....                       | 155 |
| Tableau 6 : Diversité des contacts selon la capacité mémoire et le type de filtre.....                    | 165 |
| Tableau 7 : Scenarii des tests.....   | 168 |

# Introduction

## Problématique

Les réseaux de capteurs sans fil (RCSF) viennent au secours de l'environnement et de l'industrie grâce aux récents développements réalisés ces dernières années dans les domaines de la microélectronique et des technologies sans fil (Wireless). Réalisés à partir de petites entités électroniques, ils représentent aujourd'hui une solution crédible quand il s'agit d'observer, de contrôler et éventuellement de traiter des phénomènes physiques tels que la température, la pression, les mouvements ou encore la luminosité. Les applications dédiées à de tels réseaux sont nombreuses et de plus en plus variées, allant de la défense à la surveillance, de la santé à l'environnement et le commerce.

Les constituants (ou nœuds) de ces réseaux sont généralement déployés en grand nombre en vue de collecter et de transmettre des données environnementales, d'une manière autonome, vers un ou plusieurs points de collecte. Les caractéristiques de ces nœuds impliquent diverses contraintes en matière de puissance de calcul, d'espace de stockage mémoire, d'énergie et de communication. Les nœuds dans un RCSF sont souvent amenés à opérer dans des milieux difficiles d'accès, ceci rendant leur recharge en énergie délicate, voire impossible.

Bien que dans cette catégorie de réseaux, les nœuds sont le plus souvent fixes, leur mobilité présente une multitude d'intérêts pour de nombreuses applications. La surveillance de mouvement (ou des déplacements) en est un exemple. Dès que certains nœuds du réseau de capteurs se déplacent, ils peuvent alors contribuer à étendre la couverture d'un réseau, d'en améliorer les performances de routage mais aussi avoir un impact sur sa connexité globale. Ceci étant, la mobilité dans les RCSF permet d'envisager la satisfaction d'autres besoins comme la localisation et la traçabilité d'entités mobiles qui deviennent un challenge intéressant pour les chercheurs dans plusieurs domaines d'application. Ainsi dans un contexte médical, suivre des personnes ou des matériaux susceptibles d'être des agents contaminants, faire face à des infections nosocomiales ou à l'étude d'une propagation d'un virus dans une population mobile, sont des défis à relever. Enfin, dans un contexte social, l'évaluation de l'attractivité d'une exposition, d'un salon, d'un parcours dans un lieu public peut servir à des applications d'analyse du comportement des visiteurs.

Parmi les solutions existantes pour répondre aux besoins de géolocalisation, les dispositifs de type GPS (Global Positioning System) sont une alternative pour connaître avec précision sa position terrestre. Cependant, cette solution s'avère inappropriée dans des milieux confinés et/ou d'intérieurs

tels que des galeries minières ou des établissements hospitaliers. Cela est dû aux propriétés de réflexion, de diffraction et d'absorption liées au milieu et à la topologie des parois dans le cas d'une mine par exemple.

Ainsi, des différentes alternatives et contraintes des réseaux de capteurs sans fil citées ci-dessus, l'objectif de ce travail est de proposer un modèle générique impliquant les RCSF afin de modéliser et trouver des solutions à des problématiques allant de la localisation dans des milieux d'intérieurs, au suivi et à la reconstitution de trajectoires d'entités mobiles que ce soit dans un contexte médical ou social. Le modèle de réseau proposé est de type MANET (Mobile Ad-hoc Network). Les réseaux MANET sont généralement un ensemble autonome et mobile de nœuds mobiles communiquant par une transmission sans fil qui ne suppose pas d'infrastructure préexistante. Les applications visées par cette approche sont dites tolérantes aux délais (DTN, Delay Tolerant Network) quand les délais sont souvent longs et variables et ne sont pas un paramètre critique.

La démarche proposée dans cette thématique est basée sur l'échange de proche en proche d'informations dites atomiques. Ces informations atomiques appelées par la suite 'événement de contact' représentent entre autres l'enregistrement d'une rencontre entre entités fixes et/ou mobiles. Pour des applications de localisation ou de traçabilité, l'information qu'un nœud est passé à proximité d'un autre nœud peut être une information cruciale lorsqu'il s'agit de reconstituer un parcours. Cette approche s'inspire du principe store-carry-and forward et donne la possibilité à chaque entité dans le réseau de stocker, porter et livrer cette collecte d'information vers un puits de données pouvant alimenter a posteriori des applications de localisation par exemple.

L'objectif de ce travail est de mettre en avant une démarche générique permettant d'aborder la satisfaction des applications citées précédemment. Cependant, pour l'évaluation de l'approche, nous avons décidé de nous pencher sur un seul cas d'utilisation qui est la localisation et la traçabilité dans un espace confiné. Nous montrerons à travers des résultats de simulations et d'expérimentations les impacts du mécanisme de routage proposé sur la qualité et la diversité des informations recensées concernant les déplacements et rencontres des entités mobiles.

## **Organisation de la thèse**

Le travail entrepris dans cette recherche comporte six chapitres. Dans le prochain chapitre, un état de l'art sera présenté en exposant de prime abord les spécificités des réseaux de capteurs sans fil et leurs domaines d'application. Ensuite, nous nous intéresserons notamment aux méthodes de localisation dans des milieux confinés qui est notre principal domaine d'application. Les applications tolérantes aux délais seront décrites par la suite ainsi que les différents modèles de déplacement dédiés aux DTN.

Nous décrivons la norme 802.15.4, qui est un protocole de communication défini par l'IEEE et destiné aux réseaux de capteurs sans fil. Une partie de l'état de l'art sera consacrée à la problématique de la synchronisation dans les réseaux de capteurs qui reste une contrainte considérable dans ce domaine.

Dans le chapitre 2, nous présentons de manière générale le principe d'échange et de colportage de l'information atomique dans une population mobile. Ce mécanisme de diffusion nécessite des stratégies de filtrage et de routage de l'information en situation de saturation de mémoire. En effet, chaque nœud dans le réseau exerce la fonction de routage en fonction de l'application visée avec un but commun d'acheminer une connaissance accumulée au fil du temps vers un point de collecte.

Les détails de notre proposition feront partie du troisième chapitre où nous présenterons les différentes entités du réseau, les structures de données, les stratégies de diffusion et le mécanisme de routage. La topologie adoptée et la mobilité seront illustrées dans cette partie afin de représenter des trajectoires d'entités mobiles dans un milieu confiné. Une partie expérimentale sera utilisée dans ce chapitre comme première étape de validation de notre mécanisme de routage avec comme perspectives envisagées de déployer à grande échelle cette solution de localisation a posteriori, dans la mine expérimentale CANMET au Québec. L'aspect graphique de l'application de localisation sera expliqué par la suite. Enfin, nous finirons cette partie avec une solution de synchronisation adoptée dans notre implémentation.

Le chapitre 4 proposera quant à lui une évaluation de notre démarche de validation du modèle générique par des simulations et par des mesures sur maquettes. Une partie de ce chapitre sera consacrée aux définitions des modèles de propagation du signal dont celui choisi dans notre approche. Le chapitre 5 vient conclure ce rapport de thèse en résumant nos contributions et en identifiant des pistes d'amélioration possibles de certains aspects partiellement couverts. Nous présenterons enfin, des perspectives pour enrichir davantage ce travail.

# Chapitre 1 : Etat de l'art

## 1.1. Introduction

Dans le premier chapitre de cette thèse, un état de l'art est dressé sur les principales thématiques nécessaires à notre domaine d'étude. La première partie de ce chapitre est consacrée à la technologie des réseaux de capteurs sans fil (RCSF), leurs domaines d'application et les problématiques ainsi que les défis liés à cette technologie. Les différentes méthodes de localisation dans le domaine des réseaux de capteurs sans fil feront l'objet de la seconde partie de ce chapitre. Ensuite, une description est proposée sur les applications tolérantes aux délais et quelques applications destinées à ce type de réseau seront citées par la suite. Ce chapitre décrit également les méthodes de transport de l'information dans des réseaux où la connectivité est le plus souvent intermittente. Une liste des différents modèles de déplacement est proposée pour simuler le déplacement des nœuds mobiles. Ce chapitre bibliographique s'achève sur une partie consacrée aux couches basses des réseaux de capteurs sans fil, c'est-à-dire, la couche physique et la couche MAC de la norme IEEE 802.15.4.

## 1.2. Les réseaux de capteurs sans fil

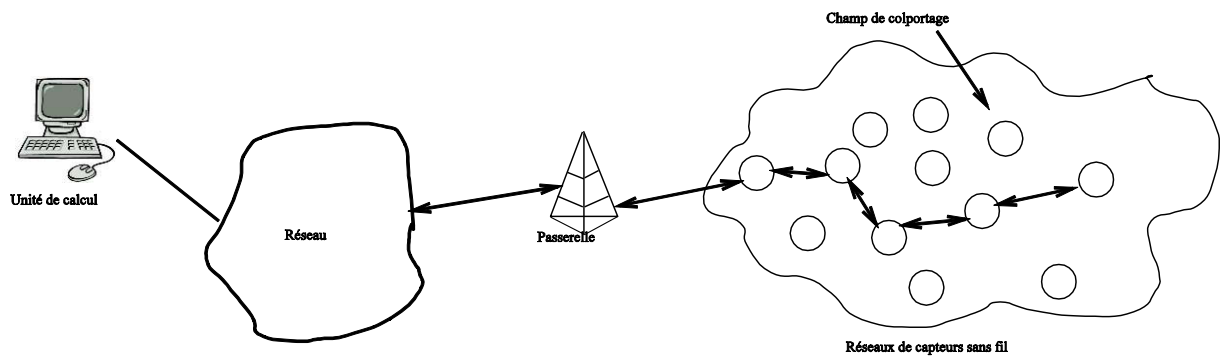
### 1.2.1. Définition

Ces dernières décennies ont vu l'avènement d'une miniaturisation du matériel informatique et de nouvelles générations de réseaux informatiques et de télécommunications qui permettent d'envisager leur usage dans des domaines applicatifs de plus en plus nombreux. Des petites entités appelées capteurs ou nœuds capteurs ont vu le jour avec le but de récolter des données physiques environnementales telles la température, la pression, les mouvements ou encore la luminosité. Un capteur est généralement composé de :

- une unité de mesure (ou capture) chargée de convertir une (ou des) grandeur(s) physique(s) en une (ou des) grandeur(s) numérique(s) ;
- une unité de traitement comprenant un processeur associé à une unité de stockage ;
- une batterie (ou une source d'alimentation) ;
- un module de transmission sans fil qui effectue toutes les émissions et réceptions sur le médium sans fil.
- et éventuellement une unité de contrôle chargée de répartir l'énergie disponible aux autres modules ;



La figure 1 représente un exemple de réseaux de capteurs sans fil ou plusieurs nœuds communiquent de proche en proche afin d'acheminer des informations à travers une passerelle pour atteindre une unité de calcul.



**Figure 1 : Exemple d'un réseau de capteurs sans fil**

La miniaturisation des capteurs sans fil, et l'optimisation de leur coût de fabrication, constituent des contraintes qui font que les solutions sont très souvent caractérisées par de ressources très limitées en matière de capacité de calcul et de stockage de données, de débit de transmission et d'autonomie énergétique. Ces limitations induisent une grande partie des problématiques de recherche dans le domaine des RCSF, en particulier la contrainte d'énergie nécessite des innovations qui ont des impacts aussi bien sur les protocoles que sur le hardware des nœuds capteurs [1].

Le nombre et la façon de déployer les nœuds dans un réseau de capteurs sans fil dépendent de l'application visée. Dans la plupart des cas, le déploiement d'un réseau de capteurs doit être précis tout en prenant en compte des objectifs définis par l'application et des contraintes du milieu. Pour des applications industrielles comme la surveillance d'un pont par exemple, où quelques centaines de nœuds suffisent, ce déploiement peut être précis et stratégiquement planifié. Néanmoins, dans de rares cas, le déploiement des nœuds peut s'effectuer de manière aléatoire.

### **1.2.2. Applications et scénarii**

Les avancées technologiques de cette décennie ont eu un impact considérable sur les micro-capteurs. La miniaturisation de ces composants, la diminution de leur coût de fabrication, la diversité des gammes de capteurs (thermique, optique, vibration, ...) ainsi que l'existence de bandes de fréquences radio, libres de droits, ont permis aux réseaux de capteurs d'intégrer plusieurs domaines d'application [1]. On trouve des applications pour la sécurité et le domaine militaire, la détection de désastres, le contrôle de l'environnement, la surveillance et la maintenance préventive des machines, la santé et la médecine, la localisation et la traçabilité, etc. Nous exposons dans cette partie quelques-unes de ces applications pour donner un aperçu de leur étendue.

### **1.2.2.1. Applications militaires**

Le domaine militaire a été l'un des moteurs essentiels pour le développement des réseaux de capteurs sans fil. Le déploiement rapide des nœuds, leur coût réduit de conception, leur capacité d'auto-organisation et leur tolérance aux pannes sont des caractéristiques qui rendent cette catégorie de réseaux marquante dans un tel domaine [2]. Une grande partie de cette croissance rapide a été provoquée par des programmes financés par l'Agence américaine pour les Projets de Recherche Avancée de Défense (DARPA pour *Defense Advanced Research Projects Agency*), notamment grâce à un programme connu sous le nom de 'SensIT' (*Sensor Information Technology*) [3].

### **1.2.2.2. Applications de sécurité**

Les applications des réseaux de capteurs liées à la sécurité [4, 5] peuvent apporter une amélioration considérable lorsqu'il s'agit de sécuriser les lieux et la protection des êtres humains. Les dégradations d'une structure de bâtiment, d'une route, d'un quai, d'une voie ferrée, d'un pont ou d'un barrage suite à un séisme ou à un vieillissement, pourraient être détectées à l'aide de capteurs sans fil placés dans les murs ou à l'intérieur du béton, sans alimentation électrique câblée ou autres connexions filaires.

### **1.2.2.3. Applications environnementales**

Il existe de nombreuses applications des RCSF liées au domaine environnemental, les principales traitent de la surveillance et de la protection des milieux biologiques et des écosystèmes, comme, la détection des débuts d'incendies de forêt, la pollution, les épidémies ou les aléas météorologiques. De telles applications permettent d'améliorer la connaissance de l'environnement et l'efficacité des moyens de lutte contre certains sinistres [6].

### **1.2.2.4. Applications médicales**

Avec la miniaturisation des composants électroniques, l'utilisation des réseaux de capteurs sans fil dans le domaine médical améliore considérablement la surveillance et le traitement des patients. Grâce à leur taille réduite, ces composants peuvent actuellement fonctionner à l'intérieur du corps humain tout en collectant des données physiologiques de meilleures qualités par exemple, pour contrôler le rythme cardiaque d'un patient, la mesure de glycémie, du taux de cholestérol ou bien la détection de cancer [7].

### **1.2.2.5. Applications industrielles et commerciales**

Les RCSF sont utilisés dans les applications industrielles et commerciales afin de surveiller la qualité des produits, le processus de fabrication et d'automatisation, le diagnostic de machines, etc. Les RCFS peuvent ainsi faciliter la construction de centres de contrôle, la commande de robots et la gestion des processus de fabrication automatisés [8]. À titre d'exemple, le contrôle de la climatisation dans les bâtiments est souvent fait par le biais de commandes centralisées.

### **1.2.2.6. Applications de localisation et de traçabilité**

La localisation et la traçabilité d'entités mobiles sont devenues des besoins importants dans diverses applications lorsqu'il s'agit par exemple, de localiser des personnes dans des zones à risque et de faciliter aussi le sauvetage des victimes. Les solutions proposées par des systèmes de localisation basée sur le GPS [9] ne sont plus opérationnels dans des milieux d'intérieur comme des galeries de mines. Ce travail est centré sur les solutions complémentaires capables de combler cette lacune. Les réseaux de capteurs sans fil peuvent apporter des solutions très utiles pour localiser ou retracer des entités mobiles en milieu confiné. Pour illustrer ce qui vient d'être dit, une problématique médicale qui préoccupe beaucoup les chercheurs sont les infections nosocomiales qui se contractent dans des établissements hospitaliers. Retracer le parcours d'un agent contaminant dans un hôpital grâce à des capteurs positionnés judicieusement peut aider à maîtriser ce fléau [10].

## **1.2.3. Problématiques et défis**

Les perspectives et les défis des réseaux de capteurs sans fil sont très motivants, mais plusieurs problématiques doivent être prises en compte. Parmi celles qui sont cruciales, nous pouvons citer : l'économie d'énergie, le routage, la qualité de service, l'auto-configuration et l'auto-réparation, la sécurité, la mobilité et la localisation.

### **1.2.3.1. Economie d'énergie**

La consommation d'énergie est une contrainte qui doit être prise en compte lors de la conception et la réalisation de solutions basées sur les réseaux de capteurs sans fil. La recharge ou le remplacement des batteries est une opération difficile, voire impossible. En conséquence, la durée de vie d'un nœud capteur est très souvent définie par la durée de sa batterie, ce qui confirme l'intérêt de cette contrainte dans la plupart des applications impliquant des RCSF.

### **1.2.3.2. Routage**

La problématique du routage est un facteur essentiel dans les RCSF, car elle doit prendre en compte les capacités de nœuds de ce type de réseau : faible débit, trames généralement courtes, nœud amené à dormir pour économiser de l'énergie. Les spécificités des réseaux de capteurs sans fil sont telles qu'il est impossible d'offrir un routage optimal en réutilisant les algorithmes de routage des réseaux filaires et ad-hoc.

### **1.2.3.3. Qualité de service**

Les caractéristiques particulières des réseaux de capteurs sans fil dues aux contraintes de ressources (énergie, calcul, mémoire), en font des systèmes peu fiables ou stables, dont le comportement est difficilement prédictible. Garantir la qualité de service est donc un défi important à relever pour certaines applications pour lesquelles le trafic échangé sur le réseau peut être constitué d'informations critiques pour le bon déroulement de l'activité de l'application.

### **1.2.3.4. Auto-configuration**

La propriété ou la capacité d'auto-configuration ou d'auto-réparation permet à un système informatique d'assurer automatiquement sa mise en fonctionnement et le traitement de certaines défaillances comme l'indisponibilité d'une ressource. Dans le cas des réseaux de capteurs, faciliter les tâches d'installation des nœuds et rendre la configuration, la maintenance et l'extension du réseau plus pratique en les automatisant, est un besoin marqué.

### **1.2.3.5. Sécurité**

Les limitations de la capacité mémoire et de calcul des nœuds d'un réseau de capteurs en font des entités extrêmement vulnérables et faciles à compromettre. Garantir la sécurité et la confidentialité des données est une tâche importante et difficile. Plusieurs techniques de protection contre les attaques des réseaux de capteurs ont été proposées par les chercheurs [11]. La transposition des techniques éprouvées pour d'autres types de réseaux n'est pas facilitée par les performances des nœuds. Ainsi, bien que l'intégrité des données soit assurée, les méthodes proposées sont parfois difficilement réalisables car trop difficiles à implémenter sur la technologie disponible. Ces considérations rendent cette nouvelle thématique des RCSF très ouverte et compétitive.

### **1.2.3.6. Mobilité**

Bien que la gestion de la mobilité dans un réseau de capteurs reste une thématique assez récente, elle engendre néanmoins de nouveaux défis tels que la communication entre nœuds capteurs mobiles qui nécessite l'adaptation de la stratégie de routage, or les caractéristiques des nœuds capteurs ne facilitent pas des échanges fréquents de tables de routage. Certaines applications nécessitent le déplacement de certaines entités du réseau pour effectuer des tâches comme relever des informations ou mettre à jour une version logicielle d'un capteur ou d'un appareil. En plus des déplacements des utilisateurs, certains nœuds capteurs doivent être mobiles surtout lorsque ces nœuds sont placés dans des machines elles-mêmes mobiles rendant dans ce cas l'utilisation de solutions câblées très difficile.

### **1.2.3.7. Localisation**

La problématique de la localisation dans les RCSF est particulièrement importante, dans la mesure où les nœuds capteurs sont dans un environnement évolutif et où ils peuvent éventuellement changer de position. Deux aspects se distinguent dans cette thématique : la localisation des nœuds capteurs et la localisation d'autres objets. Les applications de localisation en espace confiné disponibles actuellement reposent sur des solutions imprécises ou coûteuses en énergie ou matériel. Les solutions basées sur le système GPS sont une alternative pour obtenir avec précision la position d'un nœud. Cependant, ces dernières ne sont pas applicables lorsqu'il s'agit de localiser des entités mobiles dans des milieux confinés ou d'intérieur.

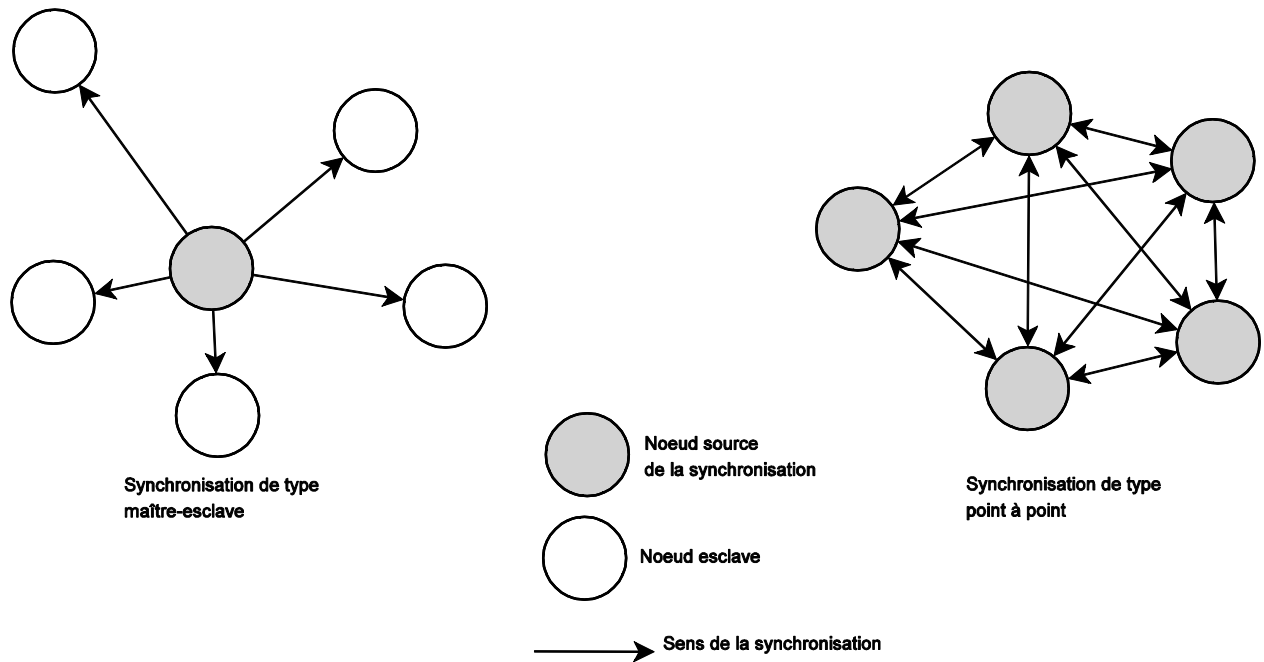
## **1.2.4. La synchronisation dans les réseaux de capteurs sans fil**

Le processus de synchronisation des nœuds dans un réseau de capteurs sans fil est un challenge important et coûteux (en débit). Les méthodes de synchronisation classiques des réseaux filaires ne sont pas applicables aux domaines des RCSF à cause de leurs caractéristiques et contraintes très différentes [12]. Parmi ces caractéristiques, on peut citer :

- L'énergie : la durée de vie limitée de la source d'énergie des nœuds capteurs rend les méthodes de synchronisation difficiles à mettre en place. En effet, ces méthodes nécessitent des échanges de messages trop longs et nombreux ce qui induit des périodes d'écoute excessivement longues.
- La bande passante : les méthodes de synchronisation sont contraintes à ne pas saturer le réseau avec un nombre de transmissions important.

- La capacité de calcul : certaines approches de synchronisation nécessitent une capacité de calcul importante que les nœuds capteurs ne peuvent pas satisfaire à cause de leur architecture généralement simplifiée.
- La mobilité : les méthodes de synchronisation doivent prendre en considération les changements de topologie.
- La latence : l'un des problèmes des méthodes de synchronisation est le non-déterminisme de ses solutions causé par la latence et par l'impression des instants d'émission et de réception (inconvenient plus marqué pour les réseaux de capteurs qui sont basse énergie et bas débit)
- La précision : pour des solutions efficaces, les méthodes de synchronisation doivent avoir une précision de l'ordre de la microseconde ( $\mu s$ ).
- La durée de validité de la synchronisation : pour économiser leur énergie, les capteurs doivent impérativement avoir de longues périodes de sommeil. Les méthodes de synchronisation doivent s'adapter à ce mode de fonctionnement et proposer des solutions valides hors des périodes de synchronisation.
- Le coût de déploiement : l'un des avantages des réseaux de capteurs est leur coût assez faible, les solutions de synchronisation doivent elles aussi être suffisamment économiques.

En se basant sur les travaux de [12], deux catégories de méthodes de synchronisation se distinguent dans la littérature pour synchroniser un réseau de capteurs sans fil : les méthodes de type maître-esclave et les méthodes point-à-point. Ces deux catégories sont représentées dans la figure 2 :



**Figure 2 : Synchronisation de type maître-esclave et point-à-point**

Dans les méthodes de type maître-esclave, tous les nœuds esclaves se synchronisent aux dépens du nœud maître. Tandis que dans les approches point-à-point, aucune distinction entre les nœuds n'est nécessaire et tous les nœuds se synchronisent entre eux. Les avantages et inconvénients de ces deux méthodes peuvent se résumer comme suit :

- Synchronisation maître-esclave :
  - Facilité de mise en place.
  - Nécessité de désigner un maître.
  - Indisponibilité lors de la perte du maître.
  
- Synchronisation point-à-point :
  - Robustesse aux aléas de la connectivité dans le réseau.
  - Complexité de mise en place.

Nous citons dans ce qui suit quelques exemples de méthodes de synchronisation basées sur les deux approches définies auparavant :

- Synchronisation maître-esclave :
  - NTS (*Network-wide Time Synchronization*) [13] : le réseau est construit en topologie arborescente, la racine de l'arbre représente le maître qui initie et propage la synchronisation à tous les niveaux inférieurs.
  - LTS (*Lightweight Tree-based Synchronization*) [14] : dans cette méthode, deux approches

peuvent être utilisées pour synchroniser le réseau. La première se base sur le principe de l'arborescence et la propagation de la synchronisation comme défini dans NTS. La seconde utilise des nœuds références, les nœuds qui désirent se synchroniser s'adressent directement aux nœuds références les plus proches.

- IEEE 1588 [15] : Le protocole de précision temporelle (PTP) IEEE 1588 est méthode standard de synchronisation d'horloges sur des réseaux tels qu'Ethernet avec une précision inférieure à la microseconde. Lorsqu'un système est initialisé, le protocole IEEE 1588 utilise l'algorithme Best Master Clock (meilleure horloge maître) afin de déterminer automatiquement l'horloge la plus précise dans le réseau. Cette horloge devient alors l'horloge maître et les autres horloges deviennent esclaves qui doivent se synchroniser auprès du maître.
- Synchronisation point-à-point :
  - RBS (*Reference Broadcast Synchronization*) [16] : cette approche nécessite que chaque nœud diffuse son horloge locale dans le réseau afin de réaliser une synchronisation globale.
  - TS (*Time Synchronization*) [17] : le but escompté dans cette méthode est de réaliser une synchronisation faible en effectuant un ordonnancement entre les événements plutôt qu'un ordonnancement des horloges des nœuds.
  - *Tiny Synchronization* [18] : la synchronisation s'effectue par le calcul des dérivées d'horloges au fur et à mesure des transmissions des nœuds dans le réseau.

La plupart des méthodes abordées ici présente des avantages et des inconvénients. Même les approches de type maître-esclave qui proposent des solutions centralisées, se révèlent d'une utilisation pas forcément aisée en raison des contraintes des réseaux de capteurs sans fil (connectivité intermittente, topologie dynamique, etc.). D'autre part, synchroniser un réseau de capteurs sans fil à l'aide de solutions dites faibles (point-à-point) s'avère être une opportunité intéressante lorsque le but est seulement d'ordonnancer ou synchroniser des événements entre eux.

### **1.3. La localisation dans les réseaux de capteurs sans fil**

Lorsqu'il est question de localiser des entités mobiles dans un réseau de capteurs sans fil, les solutions proposées se basent généralement sur les caractéristiques des ondes radio reçues par les nœuds du réseau. Parmi les solutions existantes orientées signal, les dispositifs GPS [9] offrent des résultats précis pour répondre aux besoins de la localisation en extérieur. Néanmoins, la localisation dans des milieux confinés ou d'intérieur n'est plus faisable à l'aide de dispositifs GPS qui nécessitent la visibilité directe de plusieurs satellites. Quand ces conditions sont satisfaites, il est possible d'équiper



les nœuds capteurs des fonctionnalités du GPS ce qui a pour principale conséquence d'ajouter un composant et des algorithmes de calcul sur un dispositif de petite taille, faible en capacité de calcul, de mémoire et en énergie. Quand ces conditions ne sont pas satisfaites ou que l'intégration de la fonctionnalité du GPS n'est pas compatible avec les capacités des nœuds capteurs de nombreuses autres solutions sont envisageables. Elles se distinguent par l'exploitation des caractéristiques du signal radio et/ou des informations échangées entre nœuds (topologie, table de routage, voisinage, etc.). Les particularités des ondes radio (réflexion, diffraction et absorption) font que les solutions pour les environnements confinés sont complexes et que leur précision reste faible. Dans le cas de milieu d'intérieur, on emploie souvent le terme de localisation par zone ou par cellule [19], qui est une solution moins précise que les méthodes physiques, mais satisfaisante.

### **1.3.1. Méthodes de localisation physique**

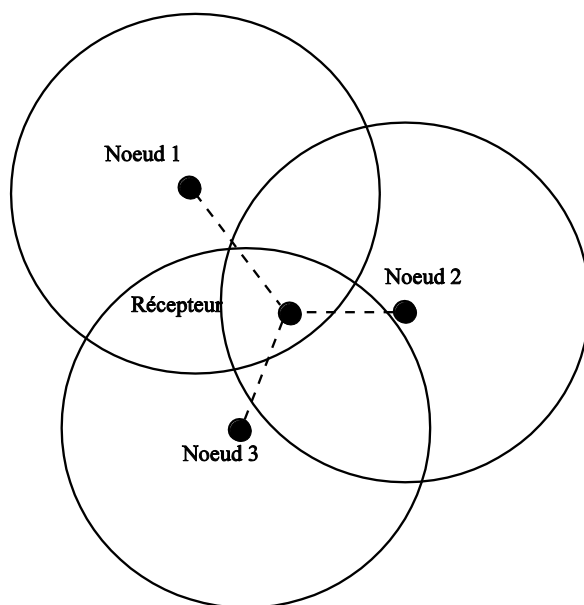
La localisation physique dans les réseaux de capteurs sans fil se base essentiellement sur les propriétés des ondes radio reçues par les nœuds du réseau, voici les principales méthodes de localisation basées sur le traitement du signal :

#### **1.3.1.1. Trilatération par calcul de temps de transit du signal**

La localisation proposée par les systèmes GPS est basée essentiellement sur le principe de trilatération, cette méthode utilise la méthode du calcul des temps de transit du signal reçu comme suit : à partir d'ondes radio reçues en visibilité directe par trois nœuds du réseau dont la localisation est connue, la position du nœud récepteur peut être calculée à partir de la formule suivante :

$$D = V * T$$

Où D représente la distance entre les nœuds émetteur et récepteur, V : la vitesse de propagation du signal, T : le temps de propagation du signal sur le médium. La figure 2 illustre une configuration avec trois nœuds références et un nœud récepteur qui a besoin de connaître sa position. Ce dernier calcule la distance qui le sépare des trois nœuds références ce qui lui permet d'estimer sa position (en appliquant par exemple des méthodes géométriques pour estimer sa position à la jonction des trois cercles de la figure 3).



**Figure 3 : Localisation par trilatération.**

Pour le calcul des distances, plusieurs méthodes sont proposées :

- ToA (Time of Arrival) [20]. Si la vitesse de propagation du signal dans l'air ainsi que l'émetteur du début de l'émission sont connus, il est possible dans ce cas de déterminer avec précision la distance qui sépare un nœud émetteur d'un nœud récepteur. En utilisant le principe de triangulation avec les nœuds références, le nœud récepteur peut déterminer sa position. Néanmoins, cette méthode présente malgré cela une faille car elle nécessite une forte synchronisation des nœuds ainsi qu'une horloge interne stable et précise. Bien que des solutions existent pour pallier les problèmes de synchronisation, ces solutions nécessitent quand même un nombre important de nœuds références de calculs, ce qui conduit à des coûts en équipements et en énergie [21].
- TDoA (Time Difference of Arrival) [22] [23]. Pour pallier la contrainte de forte synchronisation de la méthode ToA, TDoA a une approche différentielle et utilise des signaux de natures différentes, l'ultrason et la radio par exemple. La différence des temps d'arrivée de ces signaux permettra de calculer la distance entre deux nœuds sans qu'il y ait la nécessité d'une forte synchronisation. Il faut cependant connaître les temps de propagation des deux types de signaux. La seule contrainte de cette méthode, est que chaque nœud dans le réseau soit pourvu d'antennes pour les différents signaux et d'une horloge précise, ce qui n'est pas forcément le cas dans les produits disponibles actuellement. De plus, la communication à travers les deux signaux ainsi que la gestion des deux modules sans fil entraînent un surcoût énergétique.

- AoA (Angle of Arrival) [24]. Dans cette méthode, chaque nœud capteur est doté de plusieurs antennes (ou d'une antenne sectorielle). Il ainsi est capable d'estimer l'angle d'arrivée du signal reçu. Cette information reçue par plusieurs nœuds voisins permettra d'appliquer une triangulation et d'obtenir une distance relative. L'inconvénient de cette méthode est le même que la précédente en terme de consommation d'énergie due à la présence de plusieurs antennes.
- Combinaison TDoA et AoA [25]. La solution proposée est une combinaison des deux approches TDoA et AoA, afin d'accroître la précision de la localisation et de l'étendre à des environnements 3D. Cette solution consiste en une amélioration des techniques utilisées en 2D en y combinant l'exploitation du temps et des angles d'arrivée.

### 1.3.1.2. Trilatération par mesure de puissance du signal

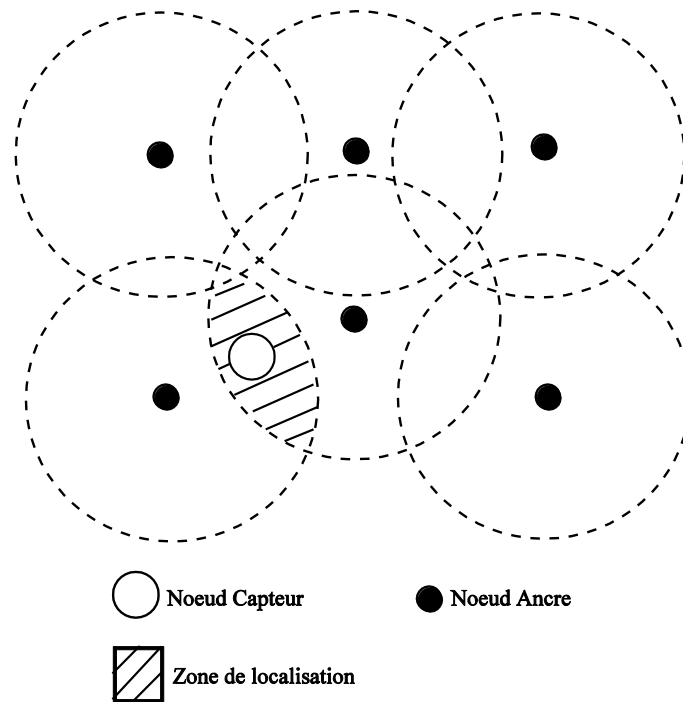
Cette approche utilise le principe de l'atténuation progressive du signal au fur et à mesure de la distance parcourue [26]. La mesure de la puissance ou RSSI (*Receive Signal Strength Indicator*) du signal reçu par nœud permet d'estimer la distance entre les nœuds. La trilatération est ensuite utilisée par le nœud récepteur avec au moins trois nœuds de référence afin de déduire sa position. L'avantage de cette solution est son indépendance à la forte synchronisation des nœuds. Néanmoins, dans un environnement confiné et/ou fortement bruité, la mesure du RSSI est fortement perturbée. En effet, les obstacles de types murs, personnes ou objets métalliques altèrent considérablement le signal durant son parcours. De plus, les nœuds mobiles dans le réseau impliquent des imprécisions lors de l'estimation de la distance.

### 1.3.2. Méthodes de localisation logique

La localisation logique repose sur l'échange d'informations de voisinage entre les nœuds du réseau telles que les tables de routage ou la topologie du réseau. Plusieurs solutions sont proposées dans la littérature :

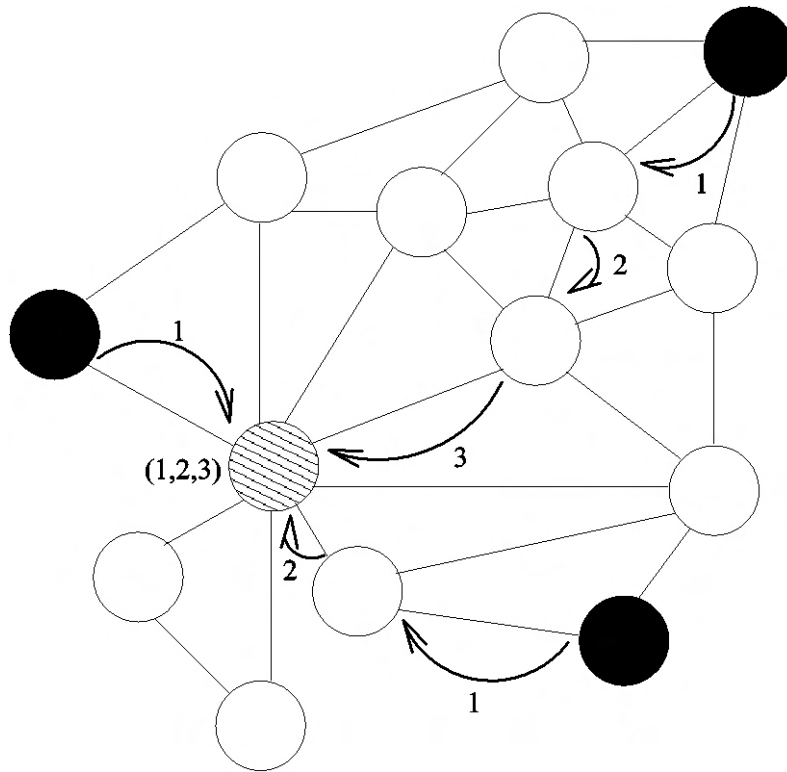
- Méthode des badges actifs [27]. Dans cette méthode, chaque nœud du réseau transmet périodiquement (chaque 10 secondes par exemple) des messages de type *hello* à destination de nœuds statiques appelés ancres (ou balises) placés à des endroits précis. Par exemple, dans un bâtiment, le rôle principal d'une ancre est de relayer les informations jusqu'au puits de données. La position du nœud sera associée à celle du nœud ancre qui a reçu le message *hello* en dernier.

- Méthode des intersections des signaux [28]. Le même principe des nœuds ancrés est repris dans cette approche. Chaque nœud capteur tient compte du nombre d'ancres à sa portée, de leur portée radio et de leur position afin d'estimer sa position dans une zone (voir figure 4). Cette zone correspond à l'intersection des zones couvertes par les ancres qu'il reçoit. Par conséquent, il est nécessaire d'avoir un nombre important d'ancres à portée des nœuds pour augmenter la précision de la localisation.



**Figure 4 : Localisation par intersection des signaux.**

- Méthode de Monte-Carlo (MCL) [29]. Dans cette approche, la localisation est réalisée sous forme de filtrage des positions impossibles des nœuds dans le réseau en utilisant aussi le principe des nœuds ancrés. La méthode Monte-Carlo calcule d'abord les distributions des positions possibles des nœuds et en fonction des positions précédentes. En fonction des positions précédentes des stations mobiles et des observations obtenues à l'aide des nœuds ancrés déployés dans le réseau, l'algorithme filtre les positions impossibles.
- Méthode des coordonnées virtuelles [30]. En réutilisant le principe des ancres, les nœuds capteurs calculent leur distance par rapport aux ancres en nombre de sauts de proche en proche entre ce nœud et les ancres. En appliquant une telle démarche, le routage de paquet devient possible en suivant le système des coordonnées déduites. Dans l'exemple de la figure 5, le nœud calcule sa distance par rapport à trois nœuds ancrés : 1 saut pour le premier, 3 sauts pour le second et 2 sauts pour le dernier ancre.



**Figure 5 : Localisation par coordonnées virtuelles.**

Une autre approche de localisation basée sur le traitement du signal reçu et dédiée au départ aux réseaux GSM et le Wifi, est la technique *fingerprint* [31], [32]. Cette méthode est composée de deux phases : durant la première phase, les caractéristiques des signaux reçus par le mobile sont enregistrées à des positions connues, ce qui revient à établir une carte radio de la zone de couverture. Pendant la seconde phase, appelée aussi phase d'exploitation, le mobile effectue une comparaison entre ses caractéristiques et celles enregistrées dans une ou plusieurs bases de données. La position attribuée au mobile sera la plus similaire des mesures enregistrées dans la base de données, ceci en tenant compte de la position précédente de ce modèle.

Parmi les solutions de localisation pour les réseaux de capteurs sans fil où le système GPS est inutilisable, nous citons les travaux décrits dans [33] qui propose une nouvelle approche basée sur le nombre d'ancres des nœuds à localiser. Deux classes de nœuds sont définies dans cet algorithme. Les nœuds dits normaux de la première classe possédant moins de trois ancres parmi ses voisins alors que les nœuds normaux de la seconde classe ayant au moins trois ancres parmi ses voisins. L'algorithme est applicable par la suite pour les deux classes de nœuds comme suit :

- *Méthode Checkout DV-hop* pour la première classe de nœuds : Le principe de base de cette méthode est de calculer la distance approximative entre un nœud de type normal et chaque ancre en multipliant le nombre de sauts minimum par la distance moyenne par saut.

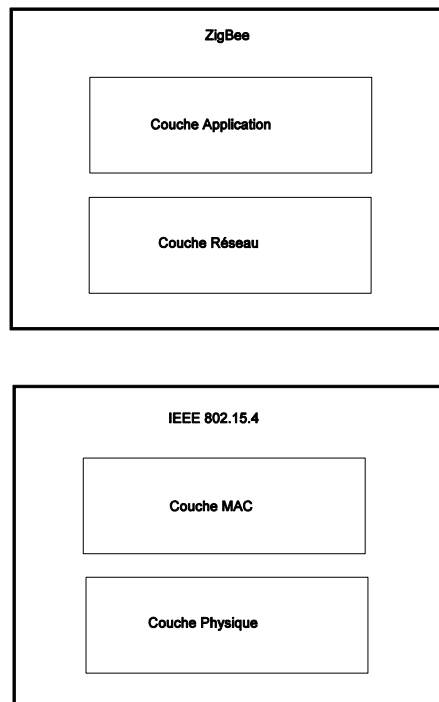
- *Méthode Mid-Perpendicular* pour les nœuds de la seconde classe : Le fonctionnement de l'algorithme pour la seconde classe de nœuds se base sur la zone de recouvrement (portée) des nœuds ancrés et la position de nœud à localiser est estimée à l'intersection des bissectrices des zones de recouvrement.

L'inconvénient majeur des méthodes logiques est qu'elles n'aboutissent pas à une localisation très précise. Cependant, ces méthodes de localisation des capteurs ont l'avantage d'être un bon compromis pour une grande partie des applications finales visées. Dans le chapitre 3 nous présenterons notre mécanisme de routage qui inclut une approche de localisation d'entités mobiles a posteriori. Cette méthode est basée sur la mise en place de nœuds fixes (appelés ici balises qui sont la signature d'une position) et sur l'échange d'informations de localisation. Une des applications visées dans notre domaine d'étude est la localisation en milieux confinés.

## **1.7. Les couches basses des réseaux de capteurs sans fil**

### **1.7.1. Caractéristiques**

Nous allons à présent exposer les caractéristiques et le fonctionnement de la norme 802.15.4/ZigBee. Cette norme est destinée à des réseaux de type personnels sans fil à bas débit, appelés également LR-WPAN (Low Rate Wireless Personal Area Network). La norme 802.15.4 [34] définit les couches basses de ce type de réseau, c'est-à-dire la couche MAC et la couche physique du modèle OSI. Cette norme répond aux besoins des réseaux de capteurs sans fil en proposant une solution aux contraintes d'énergie et de faible coût. La norme 802.15.4 est fortement attachée au protocole ZigBee [35] qui représente les couches hautes d'un réseau LR-WPAN. Le protocole ZigBee est défini et normalisé par l'organisation internationale ZigBee Alliance [36]. La figure 6 présente un découpage simplifié en couches de la pile protocolaire 802.15.4/ZigBee.

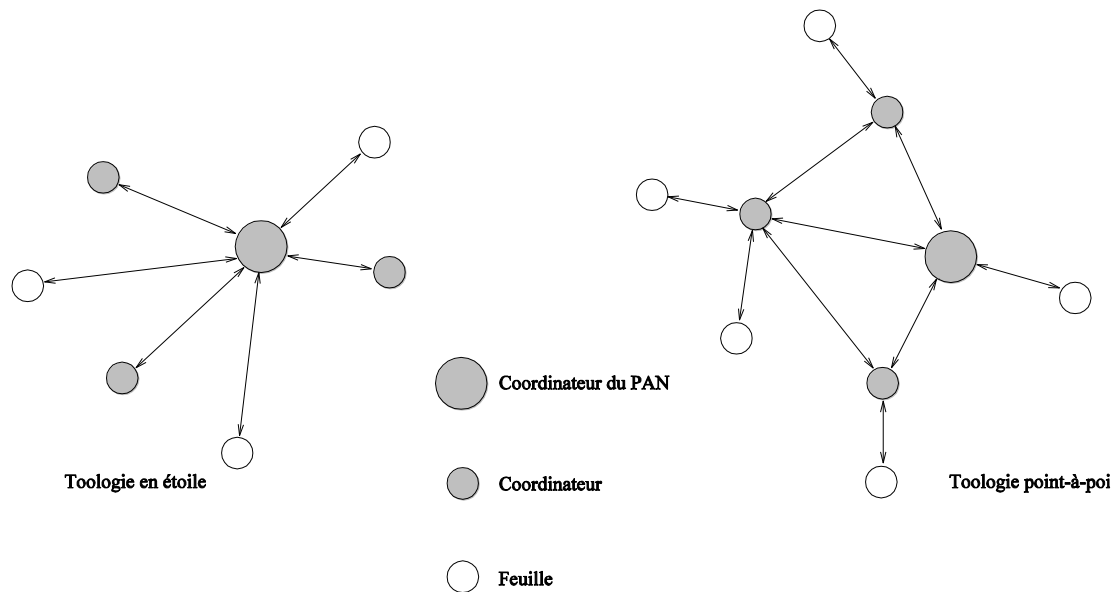


**Figure 6 : Pile protocolaire IEEE 802.15.4/ZigBee.**

La norme 802.15.4 propose deux types d'entités pour les réseaux :

- Les FFD (*Full Function Device*) : Les FFD représentent des dispositifs possédant toutes les fonctionnalités possibles. Trois modes de fonctionnement sont possibles pour de tels nœuds : Coordinateur du réseau personnel PAN (Personal Area Network), coordinateur ou bien feuille. Il faut noter qu'un coordinateur peut transmettre et recevoir des informations avec n'importe quelle entité dans le réseau alors que les RFD ne communiquent qu'avec un FFD.
- Les RFD (*Reduced Function Device*) : Un RFD ne peut jouer le rôle que d'une feuille dans un réseau. En effet, les nœuds RFD n'implémentent qu'une partie de la norme 802.15.4. La partie routage est absente pour optimiser l'empreinte mémoire et l'énergie consommée.

Les topologies prises en compte dans la norme 802.15.4 sont les topologies en étoile et les topologies point-à-point (en anglais *peer-to-peer*). La figure 7 illustre un exemple de ces deux types de topologies.



**Figure 7 : Exemple de topologie en étoile et de topologie point à point.**

Lorsque la topologie en étoile est utilisée, toutes les stations communiquent seulement avec le PAN coordinateur qui est la seule entité à autoriser une nouvelle association dans le réseau et sa portée détermine la taille du réseau. Cette catégorie de réseaux est dédiée souvent à des applications de sécurité, de confort, de gestion d'énergie, etc. Dans la topologie point-à-point, même si le PAN joue toujours le rôle de coordinateur du réseau personnel, tous les autres dispositifs du réseau peuvent communiquer directement avec n'importe quel type de voisin. Cette topologie augmente la taille et la complexité du réseau comme les réseaux maillés. Les applications de surveillance, maintenance industrielle, localisation et traçabilité sont quelques applications visées par cette topologie.

### 1.7.2. Couche physique IEEE 802.15.4

La couche physique (PHY) de la norme 802.15.4 est principalement chargée de la transmission et réception des données sur le support physique. Cette couche propose quatre débits différents.

Dans cette étude nous n'exploiterons que le débit de 250 kbps offert dans la bande ISM des 2.4 GHz. Cette solution est associée à un seuil de réception de -92 dBm et à l'existence de 16 canaux sans recouvrement espacés de 5 MHz les uns des autres.

Les points suivants résument les principales fonctionnalités de la couche physique de la norme 802.15.4 :

- Activation et désactivation du module radio : Trois états sont possibles pour le module radio, un état de transmission, de réception et de veille. Le passage en mode sommeil est indispensable pour économiser l'énergie.



- Indication de la qualité radio : A la suite d'une réception de trame, il est essentiel pour des protocoles de couche réseau et d'application de mesurer la qualité du lien. Par exemple, pour identifier le meilleur chemin, un algorithme de routage doit avoir une indication sur la qualité du lien disponible. Cette qualité est donnée par l'indicateur LQI (*Link Quality Indication*).
- Test de l'occupation du médium CCA (*Clear Channel Assessment*) : La détection de l'occupation du médium est assurée par la procédure CCA. En effet, cette information est nécessaire pour l'algorithme CSMA/CA de la couche MAC qui est une méthode d'accès pour éviter les collisions sur le médium.
- Sélection du canal : La couche physique offre 27 canaux de transmission, il est important de pouvoir basculer d'un canal vers un autre. La couche physique est responsable du choix d'abord du canal et du changement de canal. L'opération qui consiste à explorer les canaux est appelée scan.

### **1.7.3. Couche MAC IEEE 802.15.4**

Deux modes de fonctionnement sont proposés par la couche MAC 802.15.4, le premier appelé mode beacon et le second mode appelé non beacon. Dans le mode beacon (parfois appelé suivi de beacon), le coordinateur du réseau envoie périodiquement des trames beacon pour la synchronisation des nœuds qui lui sont associés. La trame correspondante à cette association représentée dans la figure 8, est appelée supertrame. Dans le mode non beacon (ou non suivi de beacon), les trames beacon n'ont plus la même fonctionnalité et ne servent qu'à la découverte du réseau. Les principales fonctionnalités de la couche MAC 802.15.4 sont donc l'accès au médium, les scans, la synchronisation du coordinateur et l'échange de données. Pour les besoins de cette thèse, nous nous attardons seulement sur l'explication de la fonctionnalité d'accès au médium de la couche 802.15.4.

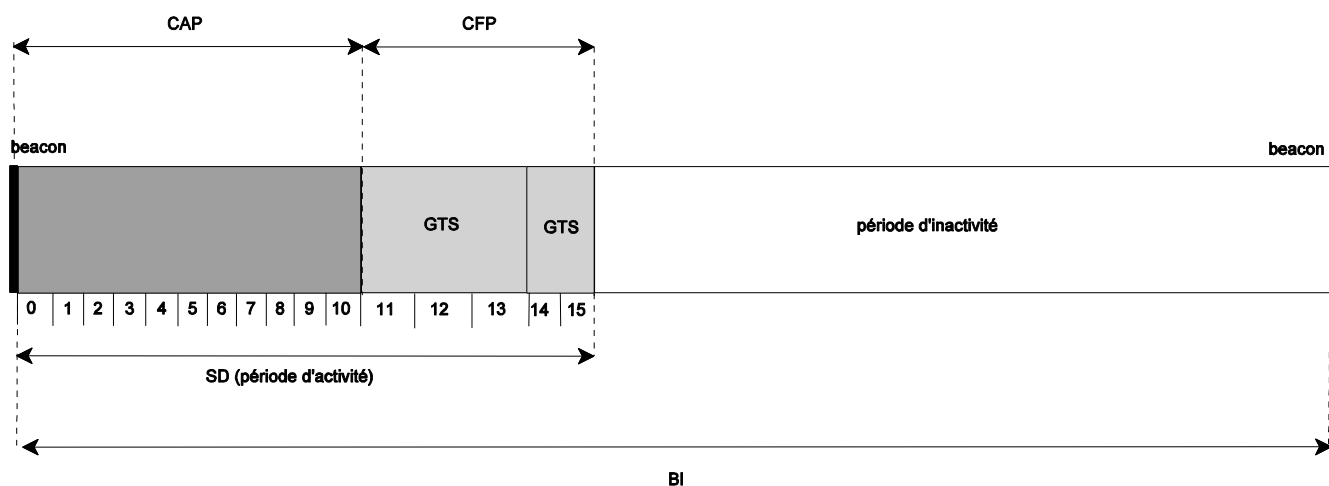


Figure 8 : Structure d'une supertrame 802.15.4.

### 1.7.3.1. L'accès au médium

Dans le mode suivi de beacon, l'entité responsable du partage du médium est le coordinateur du réseau. Tous les nœuds associés à ce coordinateur reçoivent en même temps le *beacon*. La structure de la supertrame (figure 8) est définie par un premier paramètre BI (*Beacon Interval*) qui donne l'intervalle de deux *beacons* consécutifs et le paramètre SD (*Superframe Duration*) qui indique la durée de la supertrame. Cette dernière est découpée en deux périodes : une période active et une période inactive.

La première période active est divisée en 16 slots de temps de durées égales. Elle est aussi découpée en deux parties, une période de contention CAP (*Contention Access Period*) où l'accès au médium est géré par l'algorithme CSMA/CA slotté. Durant cette période, les nœuds du réseau qui n'ont pas fini leur transmission doivent attendre le prochain cycle CAP de la prochaine supertrame. Cette période active est suivie par une seconde période active appelée CFP (*Contention Free Period*). Cette période est optionnelle et l'accès au médium est assuré par les slots de temps appelés GTS. Ces derniers sont alloués par le coordinateur à la demande des nœuds durant la période CAP. Lorsqu'un GTS est alloué à un nœud, il accède au médium sans entrer en compétition avec les autres nœuds.

La deuxième partie de la supertrame correspond à une période inactive où tous les nœuds se mettent en veille pour économiser l'énergie.

#### *CSMA/CA slotté*

La version slottée de l'algorithme CSMA/CA est utilisée pour la transmission des données durant la période CAP de la supertrame. L'unité de temps utilisé dans cet algorithme est la période de *backoff* dont la valeur est égale à  $320 \mu s$ . Trois variables sont utilisées dans l'algorithme CSMA/CA :

- NB : représente le nombre de tentatives pour accéder au médium. Elle est initialisée à 0 avant chaque nouvelle transmission. Si NB dépasse la valeur de la constante *macMaxCSMABackoffs* (dont la valeur initiale est fixée à 4), l'accès au médium est considéré comme un échec.
- CW : représente la durée en nombre de périodes de *backoff* du test du canal. Elle est initialisée à 2, et réinitialisée à cette valeur à chaque fois que le médium est trouvé occupé.
- BE : représente l'exposant de la fenêtre de tirage du *backoff*. Cette variable permet de déterminer le nombre *backoff* qu'un nœud doit attendre avant de tenter d'accéder au médium.

La figure 9 résume les différentes étapes de l'algorithme CSMA/CA slotté. Ces étapes sont aussi résumées comme suit :

1. Initialisation des variables : l'initialisation des deux variables NB à 0 et CW à 2 se fait dans cette première étape. Un test est effectué sur une variable aléatoire BLE (*Battery Life Extention*), si BLE est égale à vrai alors BE prends la valeur minimale entre 2 et *macMinBE*. Sinon, BE prend la valeur de *macMinBE* (égale à 3 par défaut). Enfin, la couche MAC identifie les limites de la prochaine période *backoff*.
2. Attente : avant qu'un nœud accède au médium, il doit attendre un délai aléatoire. Ce dernier est chiffré en période de *backoff* et tiré au sort entre  $[0 : 2^{BE} - 1]$ . Si le nombre de *backoff* est supérieur au nombre de *backoff* disponible dans la période CAP, l'algorithme CSMA/CA décrémente le nombre de *backoff* jusqu'à la fin de la période CAP et reporte le nombre restant au prochain cycle.
3. Test du canal : la troisième étape de l'algorithme CSMA/CA est consacrée au test du canal avant la transmission. La couche MAC revendique les services de la couche physique pour sonder l'état du canal. Si le canal est occupé l'algorithme passe à l'étape 4, sinon, l'étape 5 est exécutée.
4. Canal occupé : Si la couche physique détecte que le médium est occupé, la variable CW est réinitialisée à 2 et les deux variables NB et BE sont incrémentées. NB ne doit pas dépasser la constante *macMaxCSMABackoffs* (égale à 4) et BE doit être inférieur ou égal à *aMaxBE* (égale à 5). Si NB est inférieur à *macMaxCSMABackoffs*, l'algorithme CSMA/CA doit retourner la seconde étape, sinon, la transmission est considérée comme un échec.

5. Canal libre : si la couche physique détecte que le canal est libre, la variable CW est décrétementée. Dans le cas où CW est égale à 0, alors la transmission peut s'effectuer avec succès. Sinon, l'algorithme retourne à l'étape 3.

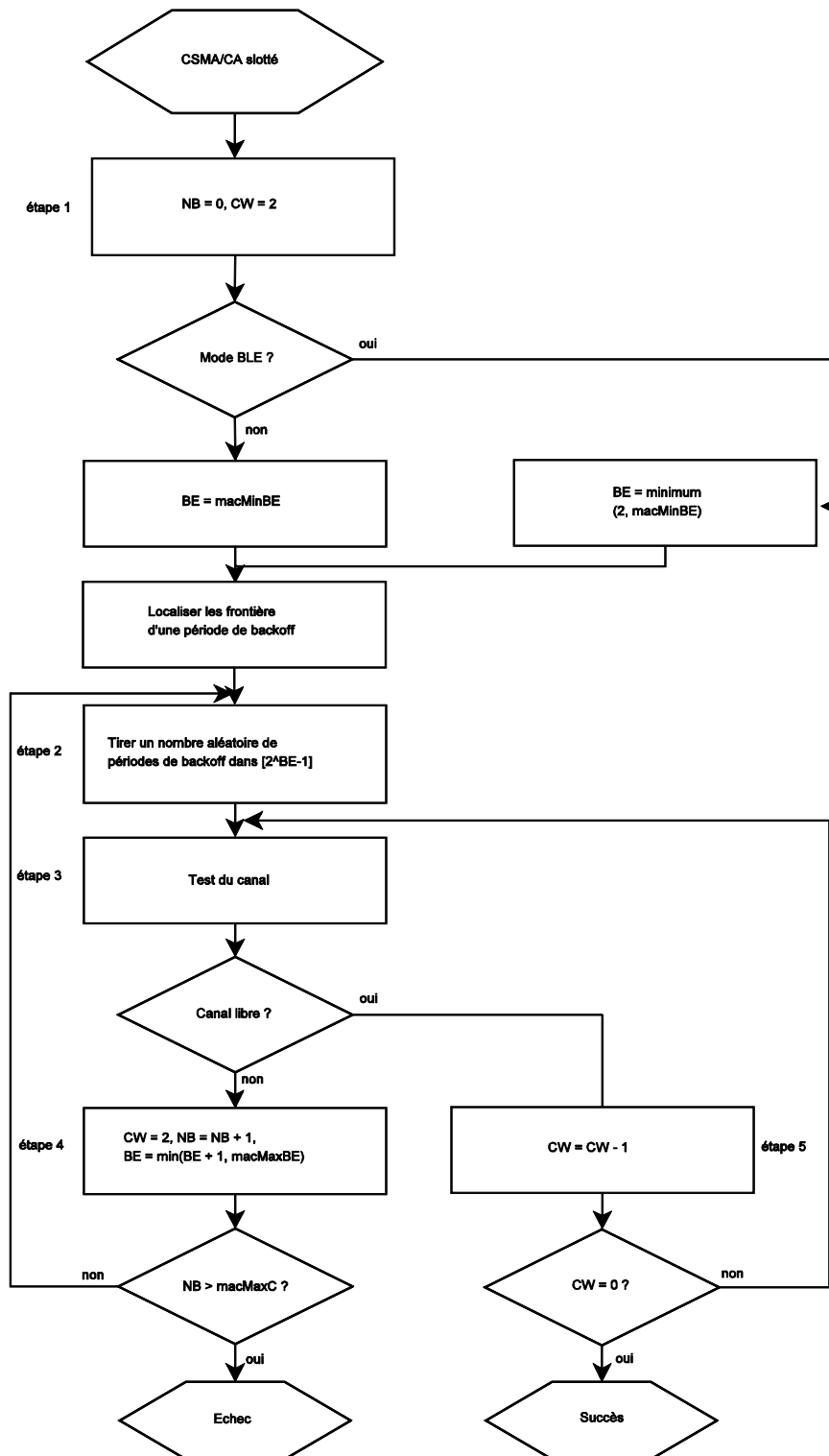


Figure 9 : Algorithme CSMA/CA slotté.

## ***CSMA/CA non slotté***

Dans le mode non suivi de beacon, l'algorithme CSMA/CA non slotté est appliqué. Aucune synchronisation entre les nœuds n'est nécessaire et l'unité de temps utilisée reste le *backoff*. La transmission est effectuée dès que le canal est jugé libre sans une synchronisation avec le début d'un *backoff*. Les étapes de l'algorithme CSMA/CA sont résumées à la fois dans la figure 10 et par les étapes suivantes :

1. Initialisation des variables : la couche MAC initialise d'abord les variables NB à 0 et BE à 3.
2. Attente : cette étape consiste à tirer une variable aléatoire de période *backoff* entre  $[0 : 2^{BE}-1]$  et attendre cette durée avant de tenter d'accéder au médium.
3. Test du canal : après expiration du délai aléatoire de l'étape précédente, la couche physique teste le canal. Si le médium est occupé, alors l'algorithme CSMA/CA passe à l'étape 4 sinon, il passe à l'étape 5.
4. Canal occupé : si le canal est considéré occupé, les variables NB et BE sont incrémentées sans que NB ne dépasse la valeur maximale de *macMaxCSMABackoffs* (égale à 4), et sans que la variable BE ne dépasse *aMaxBE* (égale à 5). Si la valeur de NB est inférieure ou égale à *macMaxCSMABackoffs*, l'algorithme CSMA/CA retourne à la deuxième étape. Sinon, la transmission est considérée comme perdue.
5. Canal libre : si le canal est considéré libre par la couche physique, la couche MAC accède au médium.

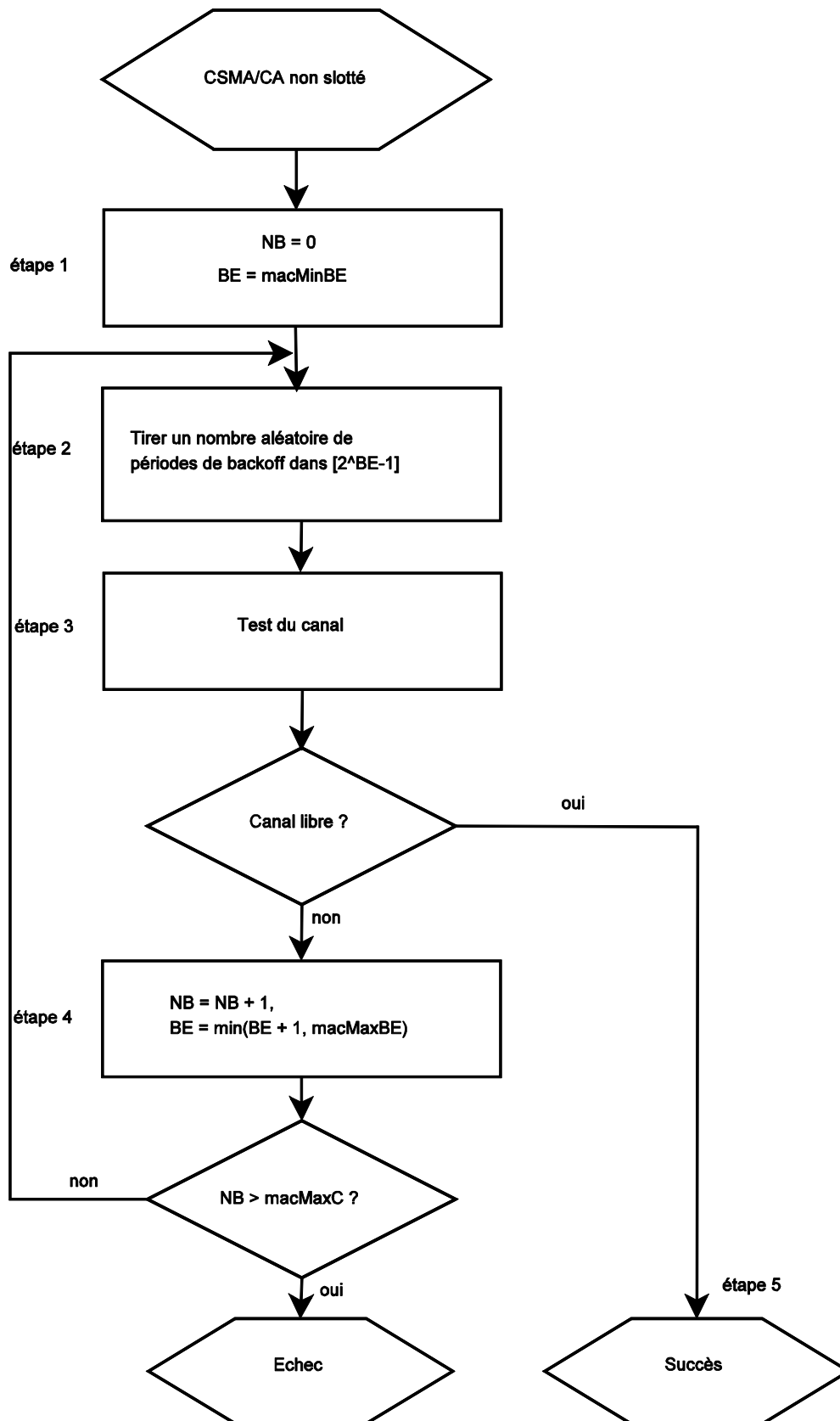


Figure 10 : Algorithme CSMA/CA non slotté.

## 1.4. Les réseaux tolérants aux délais

### 1.4.1. Définition

Les réseaux traditionnels, tels qu'actuellement Internet, doivent satisfaire des contraintes pour assurer leur bon fonctionnement : existence d'un chemin de bout en bout, bande passante suffisante, alimentation permanente en énergie. Néanmoins, ces conditions ne sont pas toujours satisfaites pour certaines applications, comme des communications longues distances, spatiales ou interplanétaires. Les réseaux tolérants aux délais (en anglais DTN : Delay Tolerant Network) [37] proposent des architectures et des protocoles pour faciliter ces communications. Les réseaux DTN désignent des réseaux globaux interconnectant des sous-réseaux régionaux. Les réseaux de ce type sont conçus pour supporter des connexions intermittentes entre nœuds, ce qui induit des latences souvent longues et variables. Ces réseaux ont été conçus initialement pour fonctionner sur de très longues distances et où la latence peut atteindre des heures, voire des jours. La figure 11 illustre un exemple d'une communication DTN.

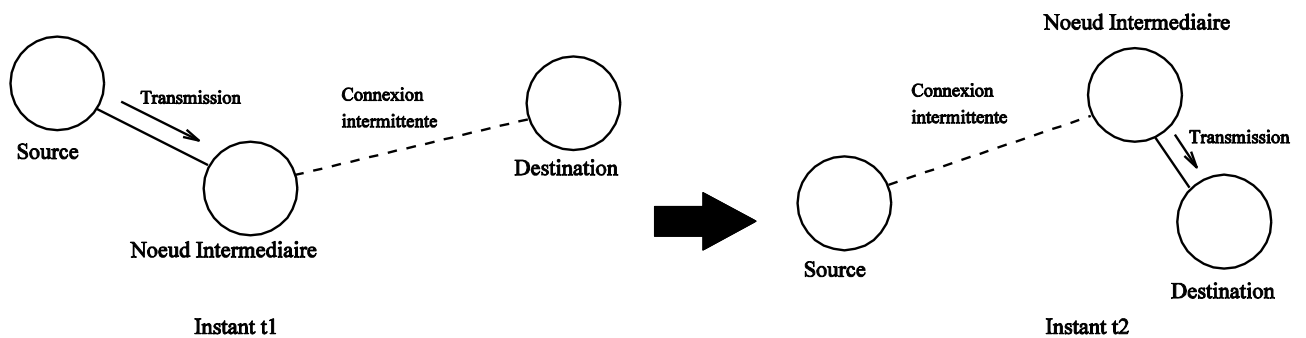


Figure 11 : Exemple d'une communication DTN.

Si nous considérons l'exemple de la figure 6, en fonction de la mobilité du nœud intermédiaire, il peut être connecté soit au nœud source, soit au nœud destination, mais jamais aux deux en même temps. En conséquence, aucune connexion de bout en bout n'est présente en permanence et des protocoles comme TCP/IP, ne sont plus fonctionnels dans ces conditions.

Ces réseaux ont une forte contrainte matérielle, car il nécessite une capacité mémoire importante pour un stockage persistant des données. En effet, quand un nœud est temporairement sans connexion avec le nœud à destination duquel il doit émettre, les paquets doivent être conservés dans des buffers jusqu'au prochain changement de topologie où ce nœud sera accessible. Ce procédé est assuré par une couche supplémentaire appelée Bundle située juste au-dessous de la couche applicative [37].

## 1.4.2. Caractéristiques des DTN

Les caractéristiques d'un réseau DTN sont les suivantes :

- Réseaux très étendus: Les distances parcourues par les données dans l'espace sont très importantes et peuvent atteindre des milliers de kilomètres. Cette technologie est capable d'offrir une solution pour des réseaux de grande envergure. Ces grandes distances associées au nécessaire stockage des paquets en attendant une connectivité favorable génèrent des très grands et très variables temps de transit.
- Connectivité intermittente : Dans les réseaux DTN, il n'y a pas forcément de chemin de bout en bout entre le nœud source et le nœud destination en raison des difficultés rencontrées dans l'environnement à savoir : le climat, les pannes d'énergie, la mobilité, etc. Pour parvenir à livrer les données, une modification de la pile protocolaire usuelle a été proposée. Cette dernière consiste en l'ajout d'une couche supplémentaire appelée couche Bundle, dont le rôle principal est le stockage des données jusqu'à ce qu'une occasion de les transmettre se présente. Le concept utilisé ici fait référence au paradigme '*Store & Forward*' [38].
- Délais longs et variables : Les problèmes de propagation de l'information dû au temps d'attente dans les files des nœuds intermédiaires, font que les protocoles basés sur des acquittements instantanés de bout en bout sont difficilement fonctionnels.
- Débits asymétriques : Le bon fonctionnement des protocoles conversationnels peut être altéré lorsque l'asymétrie est importante.
- Taux d'erreurs élevé : L'intermittence des liens et leur éventuelle grande longueur ont pour conséquence de nombreuses erreurs de bit sur une liaison. La correction ou la retransmission d'un paquet corrompu sont aussi rendues plus complexes en raison de ces caractéristiques.
- Multiplicité des protocoles : L'affluence d'une grande variété de protocoles de communication dans les DTN peut être une problématique à ne pas négliger.

## 1.4.3. Applications des réseaux tolérants aux délais

Dans les réseaux tolérants aux délais, nous pouvons distinguer trois grandes familles d'applications :

- Applications spatiales : Les besoins en télécommunications spatiales et les contraintes liées à



ces environnements (distance, délais, vitesse, mobilité) ont été à l'origine des réseaux tolérants aux délais. En 2008, des tests sur les DTN ont été réalisés par la NASA [39] pour des communications spatiales. Ce projet consistait en un envoi d'un message d'une sonde spatiale équipée d'une implémentation du Bundle Protocol en direction de la Terre, ce message a été acheminé par neuf nœuds-relais en utilisant un réseau de type DTN. Ces tests ont rencontré un franc succès et ont permis de mettre en avant l'intérêt des DTN dans ce type de communication. Cependant, les travaux menés dans ce domaine sont encore au stade expérimental.

- Applications militaires : Une autre organisation qui suit de plus près les avancées des réseaux tolérants aux délais est l'armée américaine ou plus précisément la DARPA [40]. Sur un champ de bataille, les communications entre soldats et bases militaires sont souvent perturbées voire perdues (mobilité, brouillage, etc.). Cette situation de grand danger pour des soldats déployés souvent pour effectuer des tâches périlleuses, peut justifier l'usage d'une telle technologie.
- Applications biologiques : Dans le domaine de la biologie, plusieurs projets utilisent des réseaux de capteurs pour le suivi d'espèces animales.
  - ZebraNet [41]. Le but de ce projet scientifique est d'étudier le comportement de zèbres équipés de colliers disposant de systèmes GPS (figure 12). Les données sont transférées automatiquement lorsqu'une voiture ou un avion passe à proximité. L'application effectue un échantillonnage des données GPS chaque trois minutes et la durée totale de l'expérimentation est d'une année sans aucune intervention humaine. A travers cette expérimentation, les biologistes tentent de répondre à des questions sur les impacts résultant de l'interaction entre des espèces animales et les activités humaines.



**Figure 12 : Des étudiants équipant des zèbres avec des dispositifs GPS.**

- DakNet [42]. Ce projet tente de pallier le manque de moyens financiers nécessaires à l'installation des câbles téléphoniques dans des villages en Inde, Rwanda, Cambodge et Paraguay, entre autres, afin de permettre la communication entre ces villages et le monde extérieur (figure 13). DakNet est une solution ad-hoc sans fil fournissant une connectivité asynchrone avec un coût abordable pour laquelle des bus et des motos sont équipés de dispositifs Wi-Fi pour récupérer et faire circuler des courriels et des messages vidéo à travers les villages traversés.

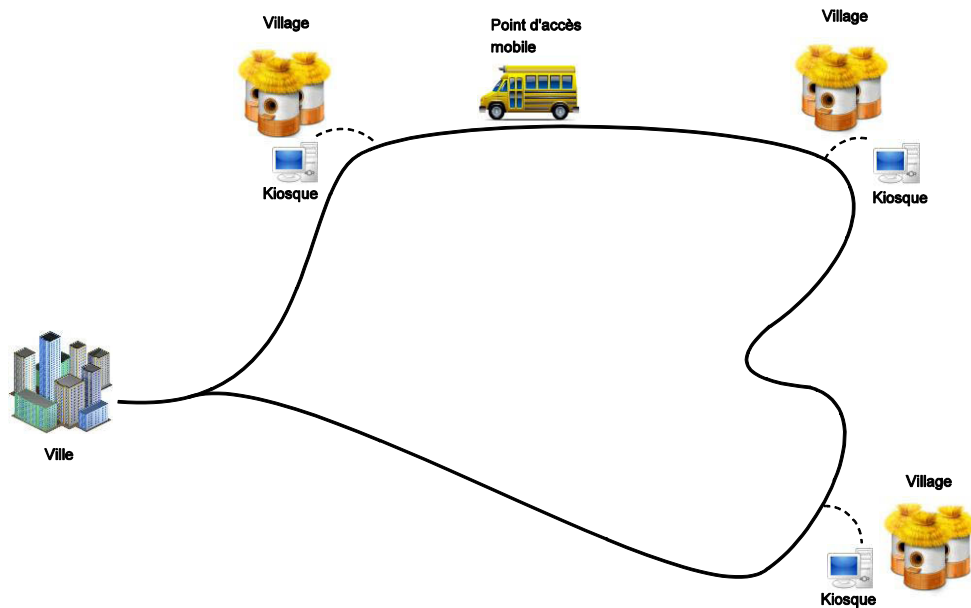


Figure 13 : Concept DakNet.

## 1.5 Le transport de l'information dans un réseau à connectivité intermittente

Un réseau de type DTN s'avère une technologie très prometteuse dont les spécificités ont un impact fort sur les mécanismes de routage. Les protocoles de routage traditionnels (réactif, proactif) [43] sont rendus inutilisables à cause de la faible densité des nœuds et des connexions intermittentes. Plusieurs méthodes de routage sont proposées dans la littérature pour résoudre cette problématique. Deux grandes catégories de protocoles de routage pour les réseaux DTN se distinguent : les protocoles basés sur la connaissance du réseau et les protocoles basés sur des nœuds additionnels.

### 1.5.1. Les méthodes basées sur la connaissance du réseau

Les protocoles proposés dans cette première approche reposent principalement sur l'échange de connaissances (liste des rencontres, topologie, table de routage, etc.) accumulées par les nœuds du réseau. Chaque nœud calcule ensuite le meilleur chemin pour atteindre la destination. Nous dressons à présent une liste des principales méthodes existantes dans la littérature.

### 1.5.1.1. Epidemic Routing.

L'*Epidemic Routing* [44] (en français routage épidémique) est l'un des plus connus et des plus simples des algorithmes de routage dans le domaine des DTN. Cette méthode est introduite comme approche de routage dans des réseaux denses avec des connexions intermittentes. Chaque nœud adopte le modèle '*store-carry-and forward*' pour stocker et diffuser les paquets qu'il reçoit des nœuds qu'il rencontre tout au long de son parcours. Par analogie à la propagation d'une maladie, chaque nœud 'infecte' d'autres nœuds en transmettant une copie de ses messages. La figure 14 montre un exemple de routage épidémique.

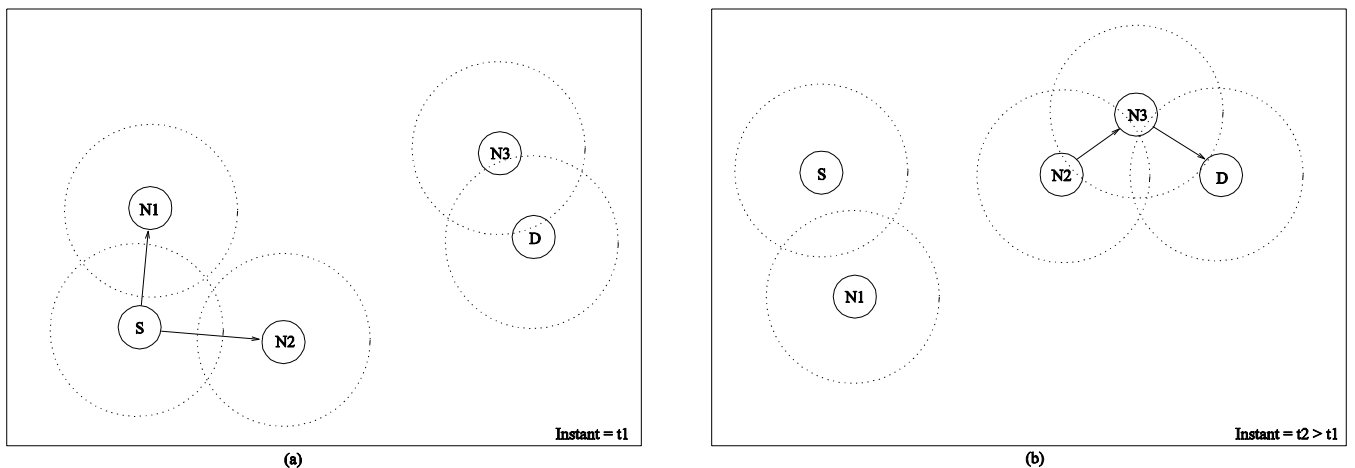


Figure 14 : Exemple de routage épidémique.

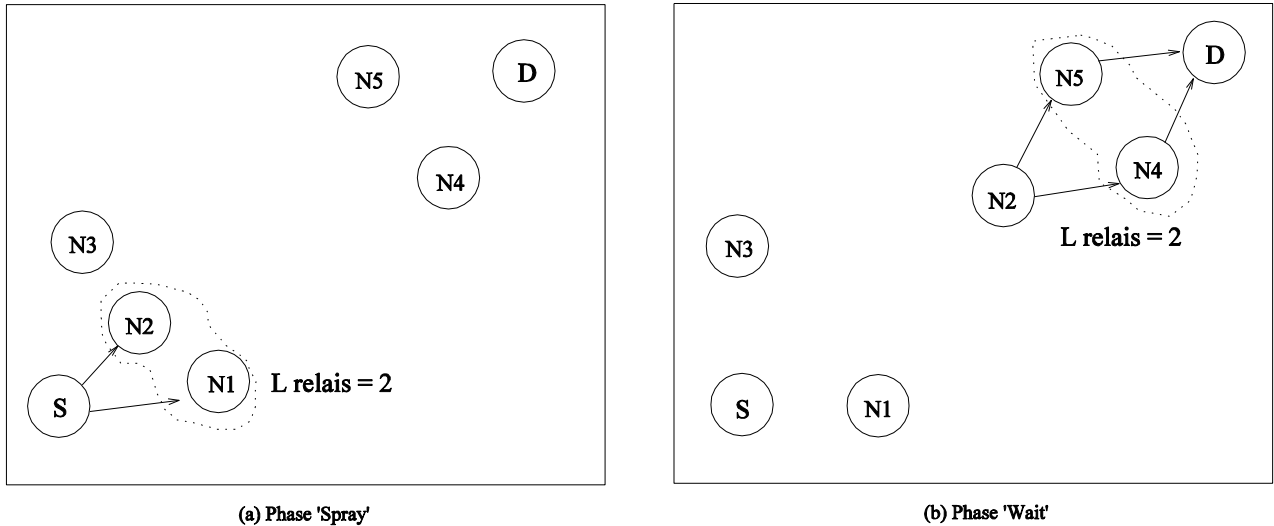
Dans la figure 14, le mobile source 'S' tente d'émettre un message vers le nœud destination 'D' mais aucun chemin n'existe entre ces deux nœuds (figure 14, partie a). La source S envoie alors son message aux deux voisins à portée N1 et N2. Plus tard, lorsque le nœud N2 rejoint le nœud N3, il lui transmet une copie du message de D. Comme N3 est à portée de la destination D, une copie de l'information générée par S est envoyée à son ultime destination (figure 14, partie b).

L'inconvénient de cette approche de routage est le nombre de ressources consommées pour atteindre la destination. En effet, la propagation des messages à tous les voisins provoque une explosion combinatoire de la taille mémoire de stockage nécessaire. Les performances de ce mécanisme de routage sont considérablement réduites s'il n'y a pas suffisamment de ressources, notamment en matière de files d'attente pour porter les copies des paquets.

### 1.5.1.2. Controlled flooding Spray and Wait.

Cette méthode de routage combine la rapidité du routage épidémique et la simplicité d'un routage à transmission directe. Elle consiste en deux phases : la phase *Spray* et la phase *Wait* [45]. Dans la première phase, pour chaque message généré à la source,  $L$  copies sont diffusées à  $L$  relais distincts

(figure 15, partie a). Si la destination n'est pas atteinte durant la première phase, alors dans la seconde, chacun des  $L$  relais propagent à leur tour le message à leurs voisins jusqu'à atteindre la destination (figure 15, partie b). Le paramètre  $L$  est choisi en fonction de la densité du réseau et le délai moyen désiré.



**Figure 15 : Routage épidémique en deux phases.**

Dans la figure 15, le nœud source  $S$  désire router une information vers la destination (nœud  $D$ ). Il transmet d'abord  $L$  copies (ici  $L = 2$ ) vers deux de ces nœuds voisins dans la première phase *Spray* (figure 15, partie a). Ensuite, dans la seconde phase *Wait*, lorsque le nœud  $N2$  rejoint de nouveaux voisins, il propage à son tour  $L$  copies vers eux qui appliqueront le même principe de propagation pour atteindre la destination. Dans cette approche, l'efficacité du routage des paquets à travers les nœuds relais est en corrélation avec la mobilité des nœuds. En effet, lorsqu'il y a une source qui propage une information vers les voisins relais, il se peut qu'aucun des nœuds de ce groupe n'atteigne la destination. Une amélioration est apportée à cette méthode appelée *Spray and Focus* [46], qui remplace la période d'attente (phase *Wait*) par une période durant laquelle les paquets sont routés à travers les nœuds relais basés cette fois non pas sur le nombre de mobiles, mais plutôt sur un ensemble de *timers* qui enregistrent l'heure des dernières rencontres. Chaque nœud maintient une liste de *timers* pour chaque mobile rencontré, il les utilise comme un cache pour garder trace des rencontres les plus récentes. Les résultats obtenus avec ce procédé présentent de meilleures performances comparées à celles obtenues avec *Controlled flooding Spray and Wait* en ce qui concerne le temps de transit.

### 1.5.1.3. Deleting obsolete messages.

La méthode *Deleting obsolete messages* [47] s'inspire de l'approche SWIM (Shared Wireless Infostation Model) [48] pour laquelle les paquets sont gardés en mémoire et ne sont pas supprimés après diffusion. Dans ce cas de figure, un paquet est considéré délivré si un nœud ayant une copie de

ce message rencontre la destination. Pour éviter une surcharge et une redondance de l'information, les auteurs soumettent différentes méthodes de suppression de paquets transmis et non nécessaires en mémoire, à savoir :

- JUST TTL : tous les paquets restent en mémoire jusqu'à expiration d'un certain temps calculé depuis leur création,
- FULL ERASE : les paquets sont supprimés juste après qu'une copie est diffusée à la station destination,
- IMMUNE TX : les paquets sont supprimés lorsqu'une copie est transmise à la destination de la même manière que FULL ERASE tout en gardant les identifiants des paquets transmis pour savoir traiter les mêmes paquets reçus plusieurs fois. Ces identifiants sont aussi partagés avec les nœuds voisins qui transportent une copie des messages déjà diffusés.
- VACCINE : en plus d'appliquer le même principe de suppression des paquets précédant lorsqu'une copie est diffusée, tous les paquets et la liste des identifiants des messages sont diffusés à tous les nœuds voisins afin de les informer, même s'ils ne possèdent pas une copie du message en mémoire.

Les résultats obtenus diminuent considérablement la consommation d'énergie des nœuds et réduisent la capacité mémoire nécessaire de 50 %.

#### **1.5.1.4. Packet dropping policy.**

Les travaux de [49] proposent également un protocole de routage basé sur la suppression des paquets avec en plus des acquittements propagés sur le réseau, tout en évitant la propagation des messages déjà diffusés. Les hypothèses posées dans cette méthode sont les suivantes :

- La mémoire est supposée infinie pour les messages reçus et limitée pour les messages générés,
- La durée et le débit de transmission sont limités.
- Aucune information sur la topologie, la position et les mouvements des nœuds n'est connue à l'avance par les nœuds du réseau.

Le fonctionnement de l'algorithme est le suivant :

- Chaque nœud garde en mémoire tous ses messages jusqu'à la prochaine rencontre,

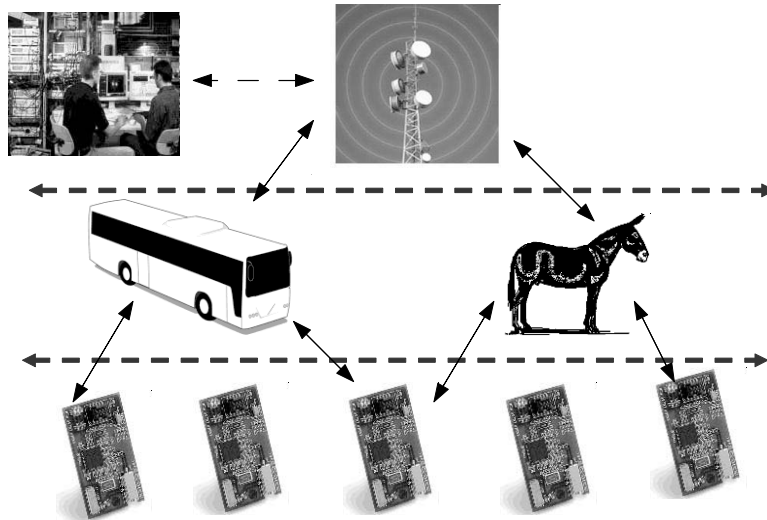
- Les messages sont propagés jusqu'à la réception d'un accusé de réception envoyé par la destination,
- A chaque nœud est assigné un coût qui représente une probabilité que les paquets émis parviennent jusqu'à cette destination,
- Lorsqu'un message arrive vers sa destination, ce dernier propage à tous ses voisins un message d'acquiescement,
- Une priorité est attribuée aux paquets les plus récents afin d'éviter les réceptions multiples d'un même message, cette priorité est déterminée selon des probabilités de rencontre entre nœuds.

L'évaluation de ce mécanisme a été réalisée à l'aide de traces de mobilité réelles pendant 60 jours et avec un déploiement de 30 bus dans un campus universitaire (Amherst, Etats Unis). Ces bus sont équipés d'interfaces WiFi (802.11 via USB) et de dispositifs GPS. Les expérimentations réalisées donnent des résultats encourageants en matière de latence et de débit

## **1.5.2. Les méthodes basées sur des nœuds additionnels**

### **1.5.2.1. Data MULE**

Comme défini dans la première partie de ce chapitre, un réseau de capteurs sans fil peut recouvrir une zone géographique étendue dans le but de récolter des données environnementales. Il faut donc rapatrier ces mesures vers des centres de traitement avec un minimum d'infrastructure câblée possible. Pour répondre à cette demande, des mobiles équipés de modules communicants peuvent servir au rapatriement des données collectées en parcourant les zones dans lesquelles les capteurs sans fil sont positionnés. De telles stations mobiles sont appelées *Data MULEs* [50] (en anglais Mobile Ubiquitous LAN Extensions) qui transportent des informations jusqu'à des points relais ou des centres de traitement. Généralement, le mouvement des mobiles se fait de manière aléatoire ou de façon non optimum vis-à-vis d'un l'objectif de collecte de l'information. Cette approche d'utilisation de nœuds mobiles comme mules est aussi désignée architecture trois étages (figure 16). La figure 16 montre les trois différents niveaux qui correspondent à la récolte de l'information, son transport par des mules, et leur centralisation vers leurs destinations finales.



**Figure 16 : Architecture trois étages dite data MULE.**

L'avantage des data mules est la facilité de mise en place de telles solutions, il n'est pas nécessaire d'appliquer des algorithmes de routages complexes pour router l'information, ou d'avoir une connaissance particulière de la topologie. Cependant, étant donné le déplacement « aléatoire » des mules, la collecte d'information n'est pas forcément complète. La figure 17 illustre un exemple de déplacement d'une mule à travers un réseau de capteurs. Durant son déplacement, la mule entre en contact avec les capteurs et collecte les données fournies par ces nœuds. Cependant, dans les cas pour lesquels le déplacement des mules est effectivement aléatoire, la latence est un paramètre difficile à borner avec pertinence. Cet inconvénient peut entraîner rapidement une saturation de la mémoire des nœuds relais. Ainsi, les données en transit peuvent être supprimées avant d'être acheminées vers la destination, remettant en cause la fiabilité de la couche transport.

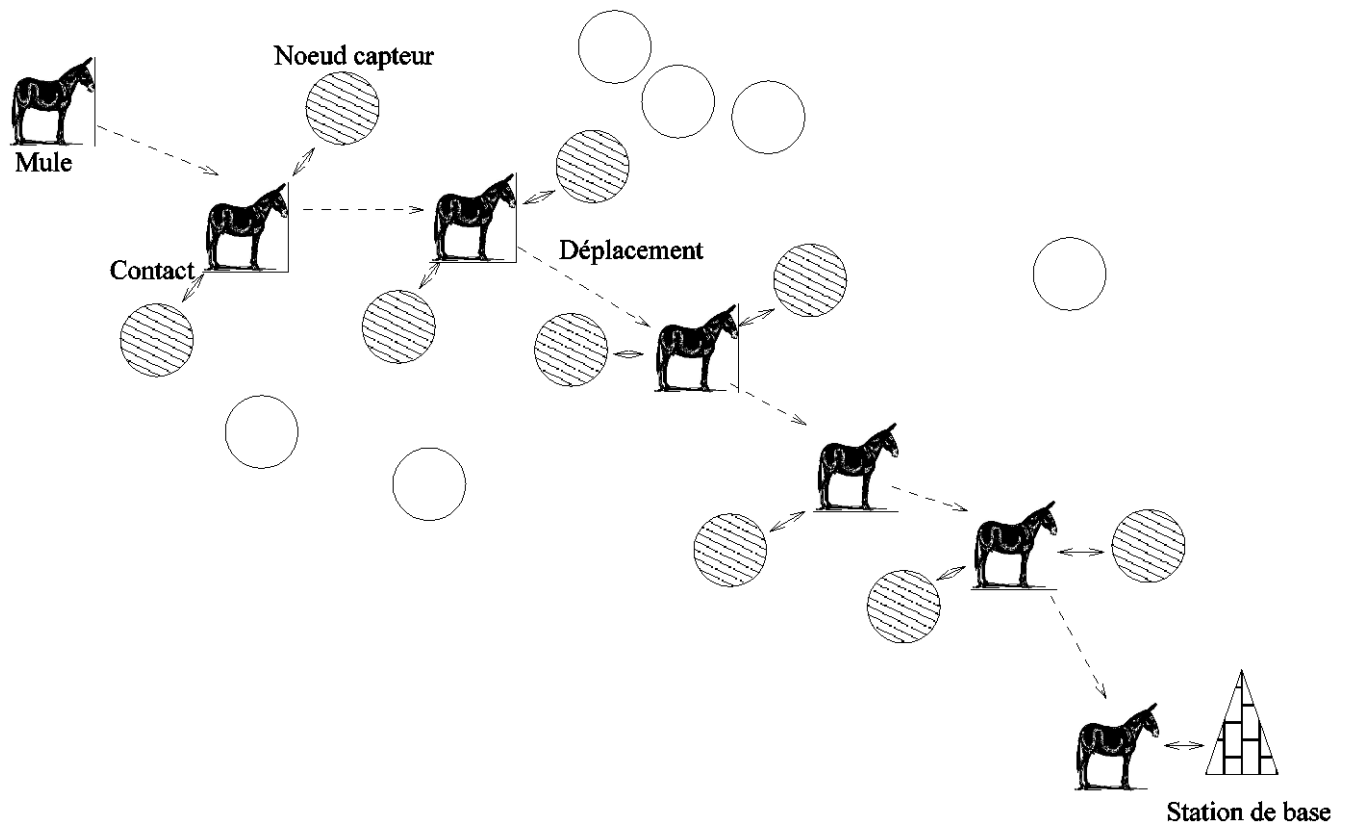


Figure 17 : Déplacement aléatoire d'une mule dans un champ de capteurs.

### 1.5.2.2. Message Ferry

Contrairement aux déplacements aléatoires des data mules dans [50], les auteurs dans [51] utilisent des nœuds mobiles appelés *message ferrys* adoptant un modèle de déplacement plus ou moins planifié souvent déduit d'une activité de l'être humain. Le rôle des nœuds *ferrys* est de collecter et de transporter des messages entre les nœuds du réseau appelés nœuds réguliers. La collecte d'information est effectuée selon deux approches possibles :

- L'approche initiée par les nœuds réguliers (*Node-initiated message ferrying*) : Pour récolter les données, les nœuds *ferrys* suivent un modèle de déplacement aléatoire sur des trajectoires connues des nœuds réguliers. Lorsque ces derniers désirent envoyer une information vers une destination, ils se rapprochent du chemin d'un *ferry*.
- L'approche initiée par les nœuds *ferrys* (*Ferry-initiated message ferrying*) : Les nœuds *ferrys* adaptent leurs déplacements en fonction des nœuds fixes qui désirent envoyer des données vers une destination. Un nœud *ferry* commence d'abord par diffuser sa position périodiquement aux nœuds fixes. Ensuite, les nœuds réguliers intéressés par l'envoi de messages répondent avec une requête. Une fois la requête reçue par un *ferry*, ce dernier adapte sa trajectoire en fonction des demandes.



Cette méthode améliore les performances d'acheminement des paquets comparée au modèle des mules. En effet, en connaissant les différentes positions et en adaptant les déplacements à ces positions, la latence pour transporter un paquet de bout en bout peut être diminuée considérablement ce qui peut être avantageux lorsque les nœuds sont contraints en énergie et en capacité mémoire. La seule difficulté engendrée dans cette solution est lorsque les nœuds se déplacent arbitrairement. Cette problématique est due aux positions incertaines des nœuds fixes qui rendent le parcours des mules délicat et imprécis.

### 1.5.2.3. Throwboxes

Le routage proposé dans cette méthode est basé sur l'ajout de stations fixes appelées *throwboxes* [52]. Ces nœuds fixes jouent le rôle de relais en créant plus de contacts dans le réseau. Ces nœuds sont supposés avoir une grande capacité de stockage, de calcul et d'énergie. La figure 18 montre un exemple d'un réseau DTN utilisant des *throwboxes*. Le nœud source envoie d'abord un message vers le *throwboxe 2* lorsqu'il est à sa portée (figure 18, partie a). Ensuite, lorsque la destination se rapproche à son tour du *throwboxe 2* récupère ce message (figure 18, partie b).

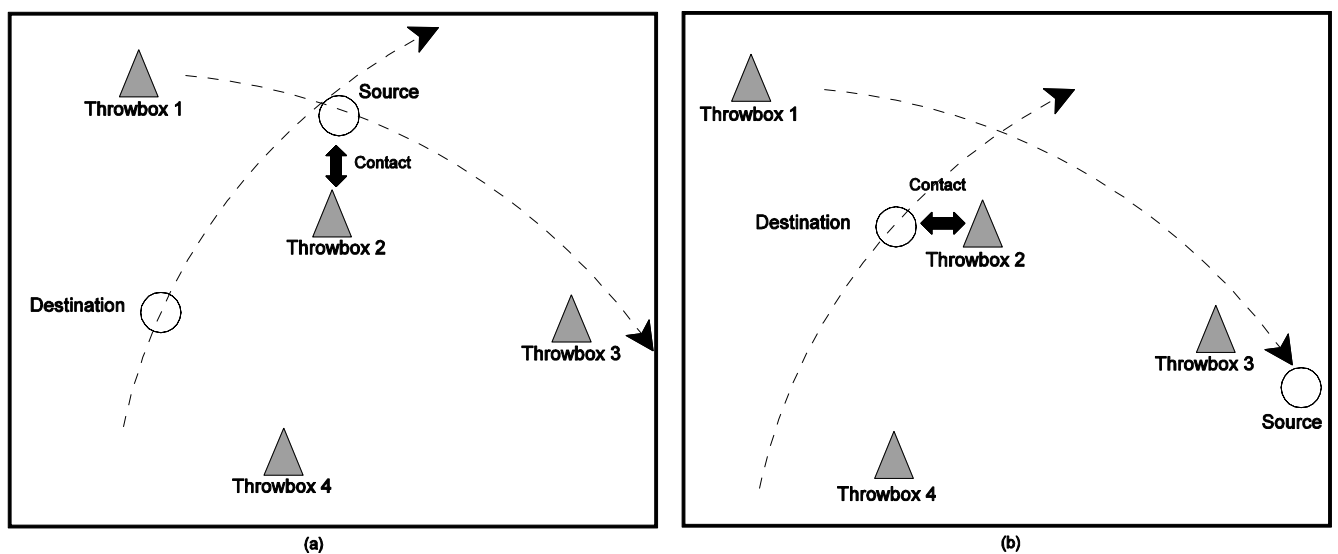


Figure 18 : Exemple d'un DTN utilisant des throwboxes.

Cependant, l'utilisation des *throwboxes* sans une gestion efficace de l'énergie rend ce type de réseaux inopérants. En effet, les auteurs de [53] justifient que si les nœuds n'adoptent aucune gestion énergétique, la durée de vie du réseau risque de diminuer considérablement. Toutefois, si les nœuds optent pour une gestion drastique de l'énergie, une perte importante d'information aura lieu, due aux périodes d'économie d'énergie, augmentant ainsi la durée de vie du réseau sans pour autant améliorer les performances.

Comme nous l'avons vu précédemment, de nombreux protocoles de routage pour les réseaux tolérants aux délais existent. Chaque méthode propose un compromis entre trois aspects importants d'un réseau de capteurs et qui sont : la gestion de la mémoire, l'économie de l'énergie et la mobilité des nœuds. Un autre compromis important pris en compte est la complexité des algorithmes utilisés pour assurer le bon fonctionnement de la tâche de routage des messages. Plus les algorithmes sont simples, plus leur rendu en matière de performances est faible.

## **1.6. Les modèles de déplacement**

Un modèle de déplacement exprime dans notre étude les évolutions « géographiques » d'un nœud dans un réseau. Les principaux paramètres pris en compte pour définir le parcours d'un nœud mobile sont la position, la vitesse et l'accélération. Ces modèles de déplacement peuvent être répartis en deux catégories : les modèles analytiques et les modèles basés sur des traces. Les modèles basés sur des traces fournissent des informations précises, particulièrement lorsqu'ils résultent de l'observation des déplacements d'un grand nombre de mobiles sur de longues périodes. Cependant, les nouveaux environnements des réseaux ad-hoc et les réseaux tolérants aux délais ne sont pas faciles à modéliser si les traces n'ont pas été créées au préalable. Dans ce type de situation, il est nécessaire d'utiliser les modèles analytiques. Les modèles analytiques tentent de représenter le comportement des nœuds mobiles plutôt à partir de règles que de traces. Cette partie expose une liste de modèles analytiques et décrit quelques exemples de traces utilisés dans les DTN.

### **1.6.1. Les modèles analytiques**

Deux catégories de modèles se distinguent dans la littérature : les modèles individuels et les modèles de groupe.

#### **1.6.1.1. Les modèles individuels**

Dans cette catégorie de modèle individuel, chaque nœud se déplace individuellement et adopte un comportement autonome par rapport aux autres mobiles. Cette classe en elle-même se divise en deux sous-classes : les modèles à déplacement totalement aléatoire où les nœuds n'ont aucune connaissance des positions antérieures, et les modèles souples où les positions, vitesses et directions sont gardées en mémoire et influent sur les prochaines destinations. Nous dressons dans ce qui suit une liste des modèles individuels proposés pour simuler le déplacement des nœuds dans un réseau DTN.

### ***Random Walk [54]***

Les paramètres pris en considération dans cette méthode sont la direction et la vitesse de déplacement. Lorsqu'un nœud se déplace d'un point vers un autre, il choisit d'abord un angle de direction entre 0 et  $2\pi$ . Ensuite, il doit choisir une vitesse aléatoire entre deux valeurs [ $V_{min}$ ,  $V_{max}$ ]. Lorsque le nœud a atteint sa destination après un temps  $t$  ou une distance  $d$ , il peut éventuellement changer les valeurs de ces deux paramètres. On constate que cette méthode fait partie des approches où le déplacement des nœuds est complètement aléatoire et indépendant des anciennes positions.

### ***Random Waypoint [55]***

Ce modèle contrairement au précédent ajoute une période de pause aléatoire lorsqu'un nœud atteint un point de destination avant d'éventuellement changer de direction et de vitesse. Ce modèle a été présenté d'abord comme outil d'évaluation du protocole de routage DSR [56] pour les réseaux ad-hoc. Etant donné sa simplicité d'implémentation, ce modèle est inclus dans divers outils de simulations des réseaux ad-hoc notamment dans l'un des plus connu qui est NS-2 [57].

### ***Random Direction [58]***

Le modèle *Random Waypoint* a pour inconvénient de créer des effets de concentration des nœuds mobiles au centre de la zone des déplacements utilisée dans le processus de simulation. Pour pallier cet effet de concentration, la méthode *Random Direction* force les mobiles à se déplacer jusqu'à atteindre les bords de la zone de déplacement, juste avant d'avoir à changer de direction après un éventuel temps de pause. Pour les directions et les vitesses, ce modèle adopte le même principe que les deux premiers, c'est-à-dire un angle entre 0 et  $2\pi$  pour la direction et une vitesse entre une valeur minimale et maximale. La figure 19 illustre un exemple de ce type de déplacement.

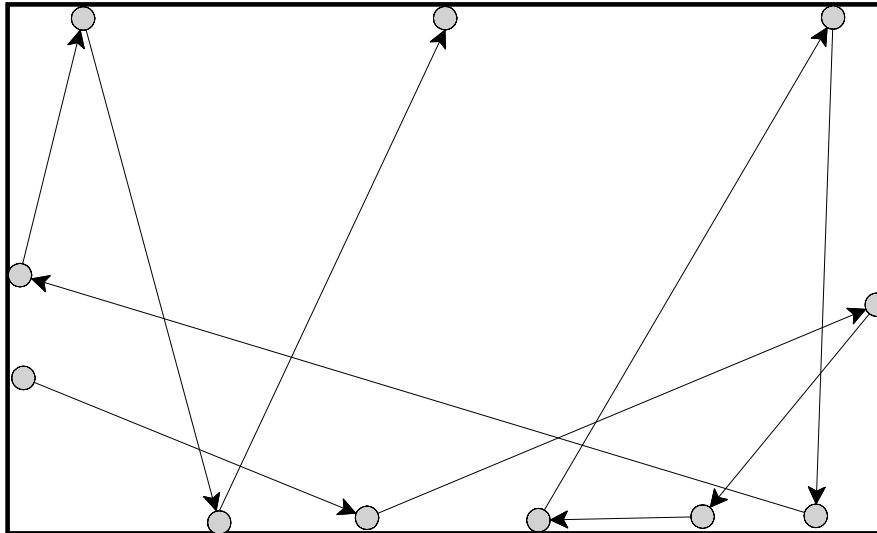


Figure 19 : Exemple d'un déplacement aléatoire *Random Direction*.

### *Restricted Random Waypoint* [59]

Ce modèle s'inspire des déplacements d'une population réelle où la zone de simulation contient des rectangles représentant des villes reliées par des autoroutes. Chaque nœud applique le *Random Waypoint* pour se déplacer d'une ville à une autre un certain nombre de fois spécifié par un paramètre. Les nœuds mobiles peuvent éventuellement se déplacer à l'intérieur d'une même ville, le nombre de déplacements effectués à l'intérieur d'un rectangle est aussi limité par un autre paramètre. La figure 20 illustre le déplacement d'un nœud utilisant le modèle *Restricted Random Waypoint*. Les villes sont représentées par des rectangles et le nœud effectue trois déplacements dans la même ville avant de changer sa destination vers une autre.

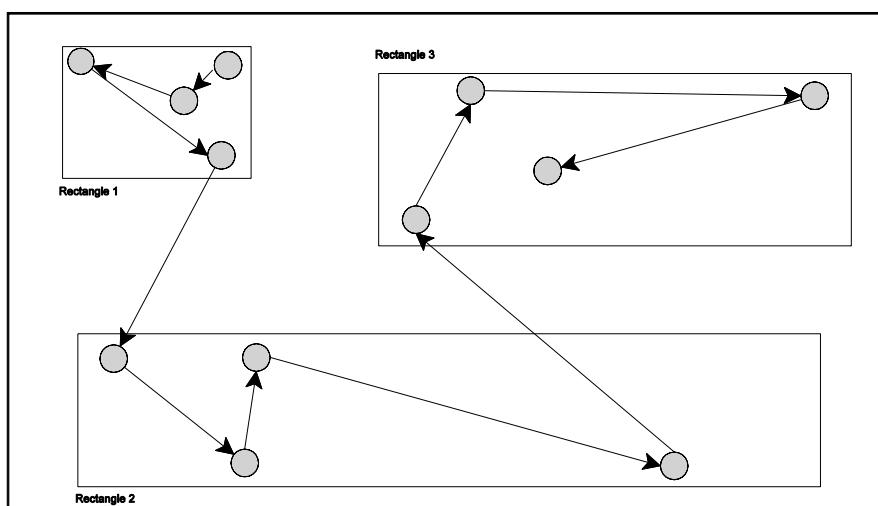
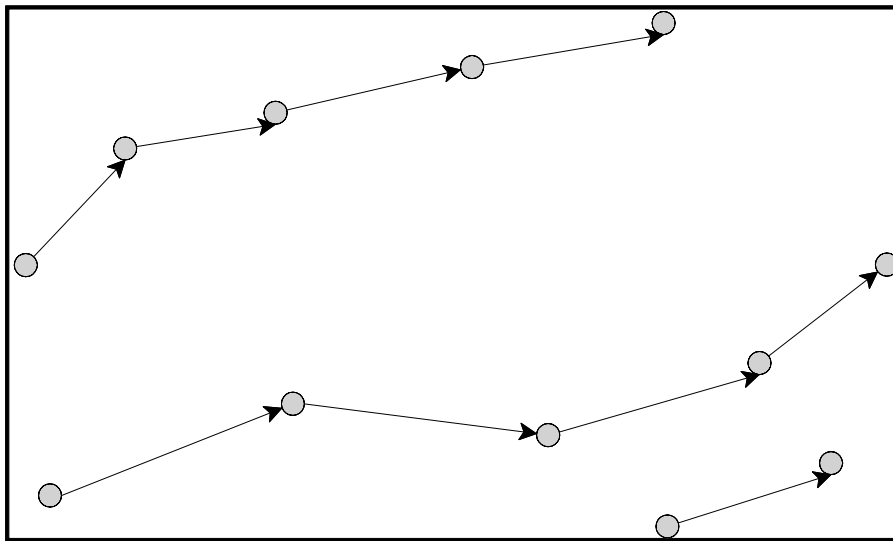


Figure 20 : Exemple d'un déplacement aléatoire *Restricted Random Waypoint*.

### ***Boundless Simulation Area [60]***

Cette méthode fait partie des modèles lissés pour lesquels une relation existe entre l'ancienne direction et la direction du tronçon du parcours suivant des nœuds, leur vitesse courante de déplacement et celle du prochain tronçon. Cette relation s'exprime avec la fonction  $V = (v, O)$  où  $v$  représente la vitesse du nœud et  $O$  la direction prise. La position du mobile est donnée par le couple  $(x, y)$  et la vitesse et la direction du nœud sont mises à jour à chaque étape du parcours selon plusieurs formules. Dans la figure 21, un exemple est donné du parcours d'un nœud mobile dans une zone de simulation. Lorsque le nœud atteint la limite de cette zone, il réapparaît ensuite dans la position opposée.



**Figure 21 : Exemple d'un déplacement aléatoire *Boundless Simulation Area*.**

#### **1.6.1.2. Les modèles de groupe**

Les modèles de groupe s'inspirent d'exemples réels de la vie courante pour la modélisation de la mobilité comme le déplacement par exemple, de citoyens d'une station de bus ou de métro vers leur lieu de travail. Une liste non exhaustive des modèles existants dans la littérature est dressée ci-après.

#### ***Exponential Correlated Random [61]***

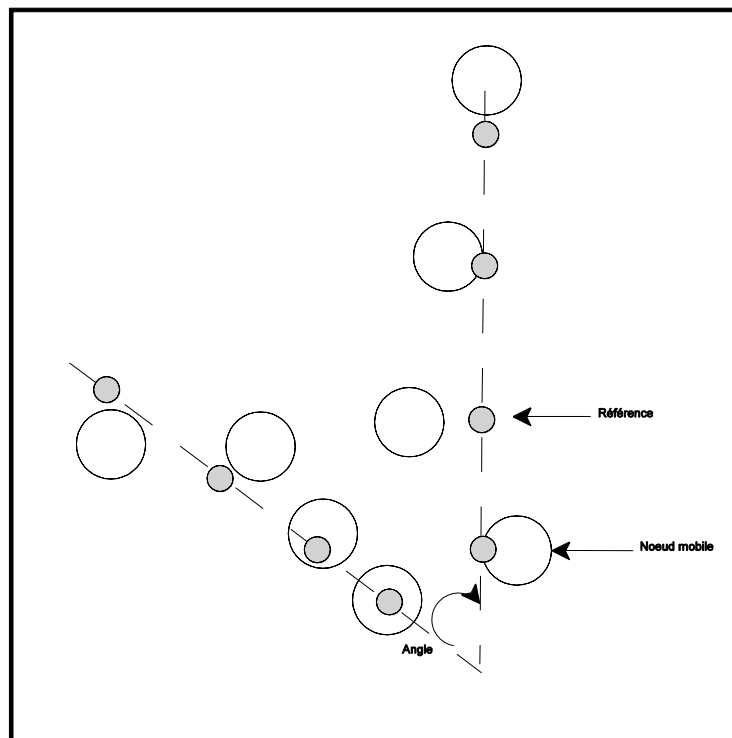
Ce modèle est assez simple et le mouvement d'un groupe de nœuds est indépendant des autres. Chaque groupe se déplace de manière aléatoire d'un point vers un autre. Les coordonnées d'un mobile sont aussi choisies de manière aléatoire à l'aide d'une fonction qui détermine les prochaines positions en fonction des positions antérieures. L'inconvénient de cette méthode est la complexité de sa fonction.

## ***Column [62]***

Les mouvements des mobiles au sein d'un groupe sont définis par des points de référence, ces points sont en réalité les mobiles du groupe. Ce dernier peut avoir un ou plusieurs points références. Les nœuds références sont d'abord placés de manière à former une colonne. Ensuite, les mobiles du même groupe sont positionnés autour de ses références, où ils peuvent se déplacer en adoptant un des modèles de mobilité singuliers. Les variations des axes de références sont définies comme suit :

Nouvelle position (Référence) = Ancienne position (Référence) + Vecteur anticipé.

Le vecteur anticipé est un angle aléatoire qui prend ses valeurs entre 0 et  $2\pi$ . La figure 22 montre un exemple de déplacement groupé avec ce modèle. Les points en gris représentent les références entourés par des mobiles. Lors du changement d'angle, les mobiles modifient leur position mais restent toujours à proximité de leurs références.

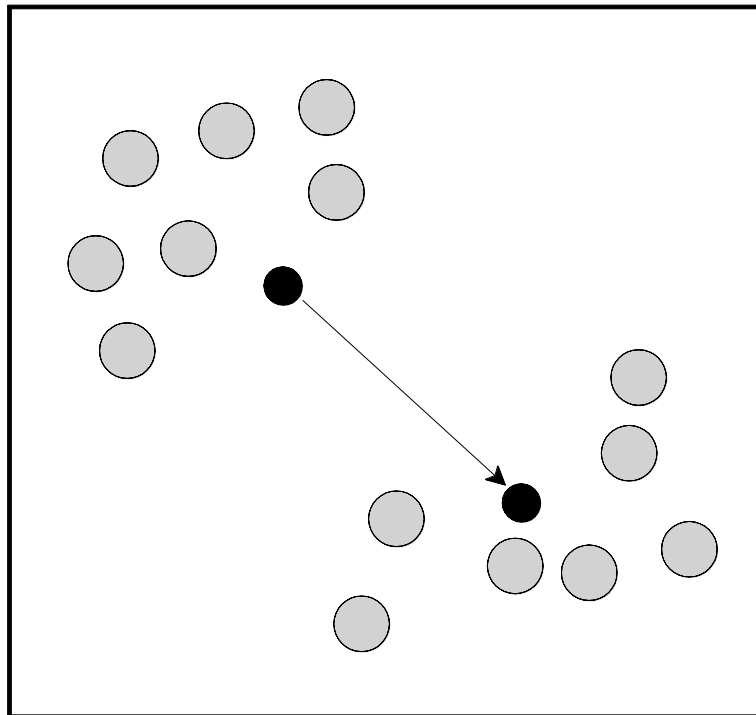


**Figure 22 : Mouvement d'un groupe de mobiles avec la méthode Column.**

## ***Nomadic Community [63]***

L'idée principale de cette approche s'inspire d'une population nomade qui se déplace d'une région à une autre au rythme des saisons. La différence entre l'approche Column est que chaque mobile dans un groupe n'a qu'une seule référence. Ainsi, le mouvement des nœuds à l'intérieur d'un groupe est

restreint par une distance limitée qui sépare le nœud de sa référence. Enfin, pour les déplacements au sein du groupe, les mobiles adoptent une méthode singulière comme définie précédemment, et il en est de même pour les mouvements des points références. De nombreuses applications existent pour ce type de scénario. Par exemple, considérons une classe d'étudiants en visite d'un musée. La classe restera groupée d'une région à une autre. Toutefois, les étudiants au sein de la même classe voudront se déplacer individuellement pour visiter une œuvre particulière. La figure 23 donne une illustration de sept nœuds mobiles avec le modèle nomade. Le point référence (représenté en noir dans la figure) se déplace d'un point à un autre, et les autres mobiles suivent son parcours.



**Figure 23 : Mouvement de sept mobiles utilisant le modèle nomade.**

### ***Purse [62]***

De la même manière que le modèle nomade, Purse est un modèle de déplacement groupé où les nœuds mobiles n'ont qu'un point de référence dans un groupe. La différence avec le premier modèle est que les nœuds suivent la référence comme des policiers qui poursuivent un bandit. Les nouvelles positions des mobiles sont calculées à l'aide de la fonction suivante :

$$\text{Nouvelle position} = \text{Ancienne position} + \text{Accélération (cible)} + \text{Vecteur aléatoire}$$

Accélération (cible) représente une information sur la prochaine cible et le Vecteur aléatoire, le modèle de déplacement singulier choisi par un mobile (Random Walk par exemple).

## **1.6.2. Les modèles basés sur des traces**

Les modèles basés sur des traces sont des modèles obtenus à partir de résultats réels. Ces modèles sont considérés par certains comme plus réalistes que les modèles analytiques. Les traces fournissent des informations précises particulièrement lorsqu'il s'agit d'un nombre de nœuds mobiles important et de longues périodes d'observation. Plusieurs études récentes ont été réalisées sur des traces collectées à partir de réseaux mobiles ad-hoc. L'extraction d'informations à partir de traces à l'aide d'outils spécifiques permet en effet soit directement de les réutiliser pour reproduire le mouvement de nœuds mobiles dans des simulations, soit de développer de nouveaux modèles de déplacement. Les traces obtenues permettent également d'évaluer l'impact de la mobilité pour des applications tolérantes aux délais. Les points qui vont suivre décrivent brièvement quelques projets se basant sur des traces réelles.

### **1.6.2.1. ZebraNet**

Cette étude a donné lieu à une collecte d'informations sur le comportement de zèbres dans une réserve naturelle (Sweetwaters) au Kenya [41]. Les informations collectées (traces) par les dispositifs électroniques placés sur des zèbres sont transférées automatiquement vers des stations de base lorsqu'une voiture ou un avion passe à proximité.

### **1.6.2.2. DakNet**

Les données collectées dans ce projet permettent la communication entre plusieurs villages dans des pays en voie de développement. Les messages (emails, vidéos, etc.) sont acheminés par des bus et des motos à travers tout le village [42].

### **1.6.2.3. Cambridge/Haggle**

Ce projet inclut plusieurs traces de communication récupérées à l'aide de dispositif Bluetooth portés par des groupes d'utilisateurs et observés durant des conférences ou dans des bureaux [64].

### **1.6.2.4. The One**

The One [65] est un simulateur spécialement conçu pour l'évaluation de protocoles de routage et d'applications dédiés à réseaux DTN. Il permet aux utilisateurs de créer des scénarii basés sur différents modèles analytiques ou bien sur des traces réelles.



### **1.6.2.5. The SUVNet**

Ce projet consiste à récupérer des données GPS collectées à partir de plus de 4000 taxis déployés dans la ville Shanghai (Chine) où chaque voiture signale sa position en temps réel [66].

### **1.6.2.6. VehILux**

VehILux [67] est un modèle de mobilité de véhicules basé sur des traces réelles obtenues en s'appuyant sur deux sources d'informations disponibles au Luxembourg. La première source de données provient du nombre de véhicules situés dans une autoroute luxembourgeoise. Tandis que la seconde source est issue d'information géographique concernant des localisations : centre commercial, industriel, résidences privées, etc.

Tout au long de cette partie de l'état de l'art consacrée aux réseaux tolérants aux délais, nous avons pu constater le nombre important de protocoles de routage dédiés à ce type de réseau où les conditions de communications sont quelquefois extrêmes. Ces protocoles proposent des compromis entre plusieurs aspects que ce soit la gestion de la mémoire, l'économie d'énergie ou bien la complexité des algorithmes. Un autre aspect abordé en détail dans les réseaux DTN est la mobilité des nœuds au sein du réseau. Là aussi, les chercheurs se sont penchés sur cette problématique pour proposer plusieurs méthodes de modélisation de la mobilité, certaines sont basées sur des modèles analytiques et d'autres, basées sur des traces réelles. Dans le chapitre 4 de ce rapport de thèse, nous proposerons notre mécanisme de routage pour des réseaux tolérants aux délais et nous développerons en détail notre approche de modélisation de la mobilité des nœuds dans des milieux confinés telle qu'une galerie de mine.

## **1.8. Conclusion**

Ce chapitre de l'état de l'art a permis d'évoquer les principaux domaines liés à notre étude. Nous avons d'abord consacré une première partie aux réseaux de capteurs sans fil qui est le premier domaine visé par cette recherche. Après avoir défini leurs caractéristiques et leurs applications, nous avons tenté de dresser une liste des principales méthodes de localisation dédiées aux RCSF. Ensuite, vu l'importance des délais dans un réseau de capteurs, il était important de consacrer une partie aux réseaux tolérants aux délais où nous avons abordé en plus des caractéristiques des DTN quelques approches de routage et comment l'aspect de la mobilité est pris en compte dans ce type de réseaux. Par la suite, nous avons développé la norme 802.15.4/ZigBee qui est une norme destinée aux réseaux de type personnel bas débits. La problématique de la synchronisation dans les RCSF nous a insisté à discuter des solutions existantes dans la littérature afin de synchroniser des nœuds dans un réseau de

capteurs. Le chapitre qui va suivre sera une transition entre cet état de l'art et notre proposition où nous allons rappeler le contexte de notre étude, les problématiques et les défis. Nous présenterons également une modélisation globale de notre proposition : un modèle générique de colportage d'informations atomiques dans une population mobile.

# Chapitre 2 : Colportage de connaissance dans une population mobile

## 2.1. Introduction

Dans ce second chapitre, nous développons le concept du colportage de connaissance dans une population mobile et proposons un modèle générique en se basant sur les constatations faites au premier chapitre dans le but de recouvrir un ensemble d'applications pour des réseaux mobiles de type capteurs et tolérants aux délais [37]. La plupart des applications concernées par cette approche a été citée durant l'étude bibliographique présentée au chapitre précédent. Notre démarche consiste à en construire une généralisation en y apportant de nouvelles caractéristiques issues de notre réflexion. Par cette fusion nous avons ainsi pu proposer un méta-modèle du colportage de connaissance instanciable à des cas d'applications spécifiques. A travers des définitions et des schémas génériques, nous allons maintenant exposer dans une première partie le concept de colportage tout en rappelant les domaines applicatifs visés. Dans une seconde partie, à l'aide d'un langage de modélisation (UML), nous développons les différentes fonctionnalités ainsi que les acteurs impliqués dans le processus de colportage de connaissance dans de tels réseaux.

## 2.2. Le concept du colportage dans une population mobile

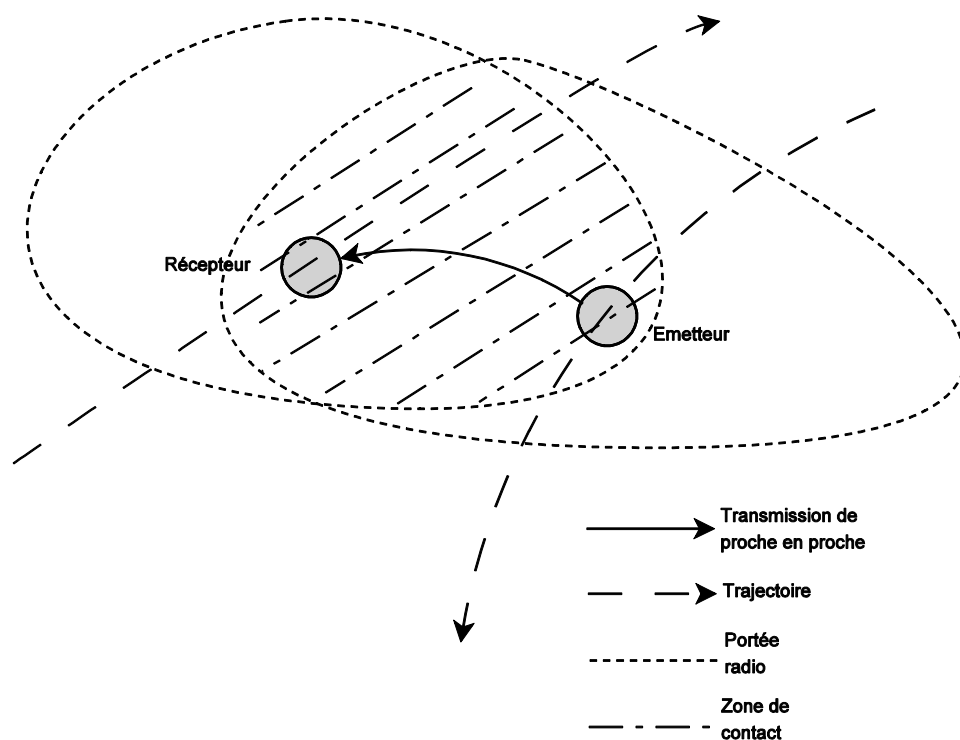
Pour comprendre l'étendue de ce sujet ou plus précisément ce qu'on nomme le concept du colportage dans une population mobile, nous allons d'abord faire un rappel sur l'idée fondamentale et initiale qui a donné jour à cette thématique : *'une analogie est envisageable entre un échange de quelque nature que ce soit dans une population mobile (humaine ou animale) et l'échange de données dans un réseau Ad-hoc pour des entités informatiques mobiles'*. En effet, supposons que l'on veuille propager une nouvelle, une alerte, une rumeur entre citoyens éventuellement mobiles ou nomades, ceci peut être assimilé à la transmission d'une « information » de nature verbale ou textuelle dans une population mobile. Ou encore, supposons qu'un élément indépendant d'une volonté établie se propage à l'intérieur de cette même population mobile, comme un virus par exemple, ceci peut également être vu comme la transmission d'une « information » de nature virale au sein de la population mobile.

Dans le cas de transmission d'une "information" de nature verbale, une affirmation, telle que : « ce film va être nominé cette année », partant d'un membre important du comité de sélection peut vite être propagée et avoir des conséquences sur le résultat de la nomination. Chacune des personnes partageant des liens sociaux en rapport avec la critique de film, peut ou non participer à la diffusion (propagation) de cette nouvelle. On retrouve cette notion dans le principe du *bouche à oreille* (en anglais : *Word of Mouth*) [68]. Ce principe vient du fait qu'en général, lorsqu'on apprécie quelque chose la tendance est

d'en parler autour de soi, de s'exprimer sur ce sujet en participant ainsi à sa diffusion. Dans le cas contraire, lorsque l'avis des personnes touchées est neutre ou en désaccord, la nouvelle est bien moins souvent propagée (on dit parfois que l'information est enterrée).

Dans un contexte médical cette fois, un rapprochement peut être fait entre la diffusion d'une information atomique et la propagation d'un agent infectieux comme un virus. Prenons l'exemple de la propagation d'un rhume, chaque être humain atteint de cette maladie virale a un fort potentiel contaminant que ce soit par éternuement, par toux ou bien par contact de salive [10].

A partir de ces exemples, une analogie peut se faire entre la transmission de l'information de nature diverse et la transmission d'une trame dans un réseau informatique sans fil composé de nœuds mobiles : le nœud émetteur de la trame est par analogie celui qui transmet l'information comme l'élément contaminant. La trame propagée dans le réseau sous la forme d'un signal radio symbolise la propagation de la nouvelle ou le virus par voie aérienne. Enfin, l'élément récepteur de la trame chargée d'identifier le contenu de cette trame, représente la personne informée ou contaminée. Ce qui vient d'être décrit peut aussi être nommé : *échange de proche en proche* que nous désignerons aussi par échange durant un contact dans la suite. Cette description est représentée dans la figure 24 où deux entités doivent être à portée pour s'échanger une trame.



**Figure 24 : Transmission de proche en proche.**

Lorsque les deux entités sont à portée, on dit qu'il y a *contact* ou bien que les deux nœuds sont *en contact*. Une des deux entités recevant une trame de l'autre peut en déduire sa présence, nous disposons donc implicitement de la capacité de *détection de proximité* entre ces deux entités [69]. Pour des applications de traçabilité d'agent contaminant, l'information qu'une personne saine ait été

en *contact* avec une autre personne, elle-même contaminée, représente une donnée cruciale pour ce type d'application.

La figure 25 illustre un exemple de transmission d'un virus entre entités à portée. La durée d'un contact est aussi une information de première importance. En effet, dans ce même exemple, il est primordial de savoir si les deux personnes ont été suffisamment longtemps en contact pour savoir si la personne saine a été contaminée. Lorsqu'un contact est établi entre une entité communicante et une ou plusieurs autres, un *événement de contact* est créé pour chaque entité. Cet événement représente l'image de cette rencontre et va donner lieu à la création d'une structure de données. Durant toute la durée de ce contact, les nœuds peuvent échanger une partie ou la totalité de leur *connaissance*. Le terme *connaissance* fait ici référence à une collection d'événements de contact éventuellement manipulés (modifier, fusionner, supprimer,... etc.). Mais nous pouvons considérer d'autres connaissances comme :

1. La connaissance d'analyse de résultats suite à la propagation. Elle correspond aux règles mise en œuvre par un acteur, une entité pour établir un diagnostic dans le domaine considéré. Au niveau médical un médecin, entité différente des nœuds de la population peut juger d'une contamination ou non. Chaque entité peut être aussi capable de s'auto-diagnostiquer.
2. La faculté ou le savoir-faire que possède chaque entité physique de s'auto-organiser pour participer au mieux à l'action de propagation. Par exemple, l'oubli de certains commentaires pour ne publier que l'essentiel dans le cas d'une propagation d'une rumeur (utilisation de filtres).

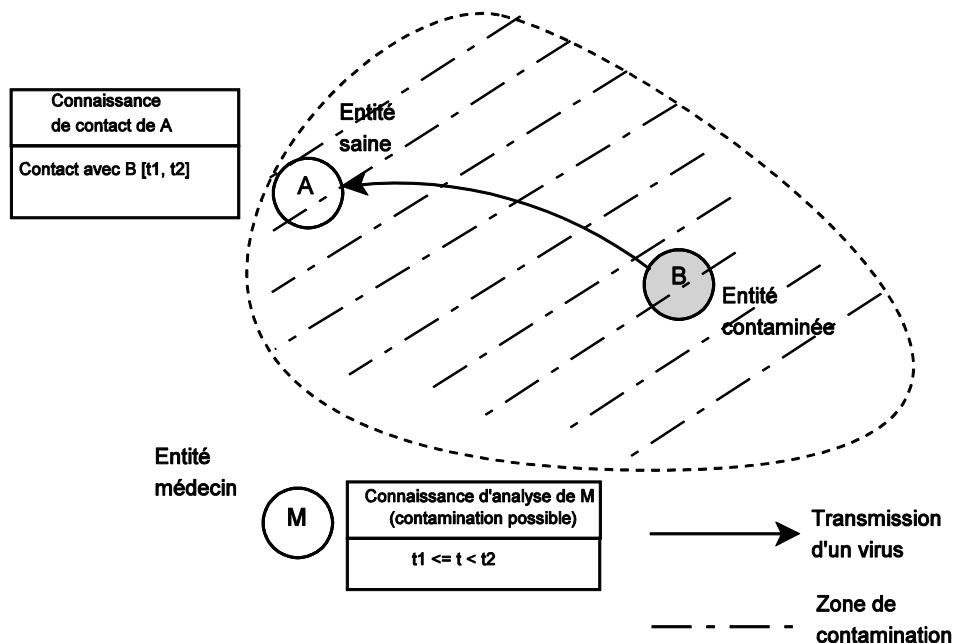


Figure 25 : Transmission d'un virus entre deux entités.

Il faut noter que le but final de cette thèse n'est pas seulement centré sur l'étude et la modélisation d'un processus de contamination dans une population mobile, mais est plutôt de proposer un modèle générique pour mettre en œuvre des réseaux de capteurs sans fil afin de répondre à des besoins de traçabilité dans des applications de diverse nature. Le potentiel de notre étude se veut le plus large possible.

Dans cette thèse, quatre grands domaines ont été identifiés en fonction des différents modes de propagation ou de comportement à la réception :

1. Médical : pour des applications de suivi de patients dans une zone d'évolution donnée,
2. Biologique : pour l'étude du comportement d'espèces animales,
3. Industriel : pour la localisation de personnes ou de matériel, dans des milieux confinés par exemple,
4. Social : pour des applications d'analyse du comportement humain en société ou dans un lieu précis, des visiteurs d'une exposition par exemple.

Notons que les types 2 et 4 diffèrent essentiellement par la densité des nœuds et par l'étendue des zones d'évolution. Il est maintenant primordial de définir à ce niveau les premiers termes qui vont être employés dans la suite de cette thèse et qui sont propres au principe de colportage.

### **2.2.1. Terminologie**

Nous définissons clairement les termes utilisés dans ce manuscrit et qui sont les suivants :

- Contact : lorsque deux entités sont à portée radio l'une de l'autre, on dit qu'il y a contact.
- Détection de proximité : on parle de détection de proximité suite à la bonne réception d'une trame d'un partenaire.
- Événement de contact : un événement de contact représente l'image d'une rencontre entre nœuds mobiles et/ou fixes. Cette structure de données définit l'information atomique du processus de colportage. Elle est composée des identifiants des nœuds partenaires, de la durée du contact et d'un champ réservé à l'application visée. Ce champ peut contenir l'énoncé d'une ou plusieurs rumeurs ou indiquer si l'entité ou nœud est porteur d'un élément

contaminant ou non. Il peut aussi servir à acheminer des données nécessaires au bon fonctionnement du réseau comme des informations de routage.

- Connaissance de contact : ensemble d'événements de contact liés à l'application qu'on notera par la suite « connaissance ». Ces événements concernent les rencontres des nœuds du réseau et peuvent éventuellement être modifiés selon les besoins de l'application.
- Connaissance d'analyse : ensemble de règles liées à l'analyse du résultat du colportage de l'information.
- Connaissance d'organisation : ensemble de règles ou de décision permettant un ajustement physique pour la propagation des événements de contact.
- Colporter : le terme colporter dans notre thématique fait référence à l'action de transporter, échanger de proche en proche de la connaissance de contact lorsque deux ou plusieurs entités sont en contact.
- Collecter : rassembler de la connaissance de contact au niveau d'un point de collecte ou une cible particulière afin d'alimenter les applications visées.

Ce mécanisme d'échanges d'informations de proche en proche conduit à un enrichissement de la connaissance des entités mais nous expose au problème d'une explosion combinatoire d'autant plus gênante si l'on tient compte du fait qu'un nœud capteur est généralement pourvu de ressources mémoires limitées.

- Filtrer : appliquer une ou plusieurs politiques de suppression, de modification (agrégation) et de choix d'événements de contact diffusés ou stockés en mémoire lorsque l'espace de stockage alloué à un nœud est saturé. Ces politiques sont également appliquées pour les besoins des applications finales.

### **2.2.2. Création de l'information élémentaire**

Au niveau signal, une détection de proximité est initiée par la bonne réception d'une trame qui déclenche le commencement du contact, celui-ci se caractérise par une date de début, une durée et l'identité des partenaires [19]. Pour se détecter, les entités du réseau doivent se signaler en générant une activité sur le médium radio à une fréquence qui dépendra de l'application visée. Lorsqu'un nœud mobile X par exemple entre dans ce qui est appelé la zone de couverture d'un autre nœud Y du réseau (fixe ou mobile), il reçoit une première trame, il note alors le début de l'évènement de contact. La durée du contact est l'intervalle de temps pendant lequel le partenaire est toujours "visible". La figure

26 représente cet événement de contact, le seuil de réception  $P_{th}$  représente la puissance minimale du signal reçu pour qu'une trame soit correctement interprétée. L'intervalle de temps  $[t_1, t_2]$  illustre la durée d'un contact entre ces entités communicantes.

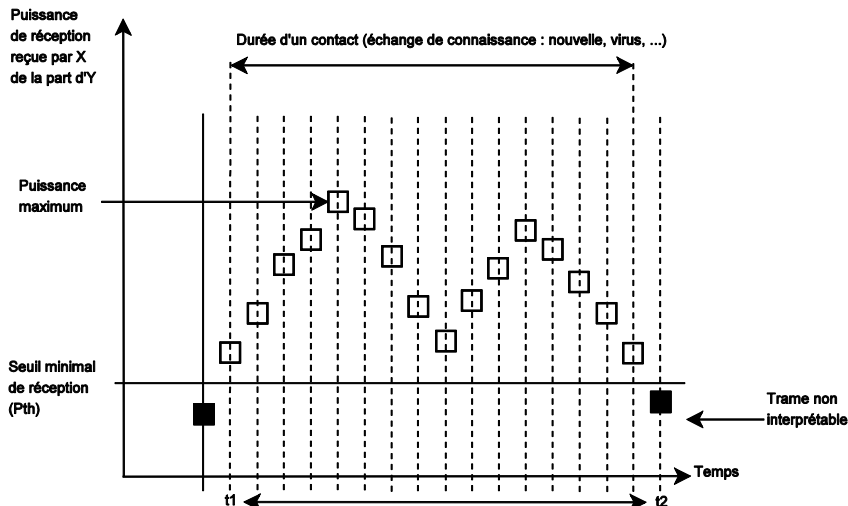


Figure 26 : Caractéristique d'un événement de contact au niveau signal.

Par la suite, à l'aide de ces informations, la structure de données 'événement de contact' est créée. L'identifiant du partenaire est inséré dans la structure suivis de la durée du contact obtenue à partir de l'intervalle de temps  $[t_1, t_2]$ . Pour les besoins de l'application visée, des données supplémentaires peuvent compléter cette structure telle que la puissance maximum de réception dans l'exemple du signal. La figure 27 illustre la structure fonctionnelle d'un événement de contact.

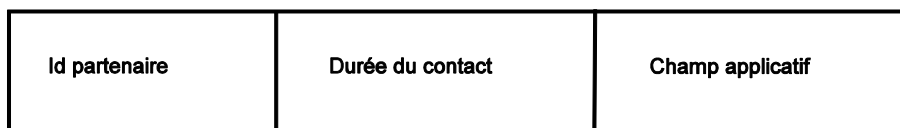


Figure 27 : Structure fonctionnelle d'un événement de contact.

### 2.2.3. Processus de colportage

Le mécanisme de colportage qui consiste à confier ou partager une information avec tous les voisins accessibles dans un réseau, peut produire une explosion combinatoire qui se traduit par une saturation du médium et/ou de la capacité de mémorisation des nœuds. Dans le domaine de l'ingénierie des réseaux [43], ceci a une incidence sur les performances de routage de l'information qui sont fortement diminuées. De ce fait, en plus du modèle générique proposé dans le cadre de ce travail, cette thèse propose un mécanisme de routage basé sur le filtrage de l'information [70]. En effet, chaque entité impliquée dans le processus de colportage est amenée à prendre des décisions localement lorsque l'espace de stockage alloué atteint un niveau de saturation (connaissance d'organisation). Dans la suite de ce mémoire, plusieurs politiques de filtrage des événements de contact propagés dans le réseau



seront spécifiées et évaluées. Ces stratégies dépendent fortement des applications concernées. Pour l'exemple d'une application de localisation de matériels dits sensibles car dangereux, la fraîcheur de l'information est un paramètre à prendre en considération en priorité. Il faut donc retenir le contact le plus récent relatif à cette entité dangereuse.

Pour illustrer ce qui vient d'être énoncé sur le principe du colportage et les applications visées, nous proposons un schéma englobant ces principales idées. Celui-ci est donné en figure 28 comme descriptif générique du principe du colportage dans une population mobile. Trois grandes parties sont identifiables :

- Le choix du domaine applicatif et du type de l'information (1), la construction du système de colportage (2) et l'analyse des données collectées (3). La première étape consiste à bien identifier le domaine d'activité ainsi que l'application finale visée. Cette spécification va impliquer des choix de comportement, d'organisation et de structuration du système modélisé. Il va être nécessaire de définir des règles dites applicatives qui vont avoir un impact sur la mise en place du système de colportage (2) mais aussi sur le mode d'analyse des données collectées (3).
- Durant la seconde étape et selon l'application finale identifiée, à partir d'une information atomique (élémentaire) créée pour ou par l'application, plusieurs nœuds dits nœuds colporteurs (fixes et mobiles) propagent cette information dans le réseau à chaque rencontre. L'objectif commun est d'acheminer ces données et de les centraliser vers des points cibles qui sont soit des points de collecte appelés collecteurs ou tout autre entité fixe ou mobile ayant un rôle de cible dans l'application concernée.
- Enfin, les puits d'information servent à alimenter durant la troisième phase diverses applications allant de la localisation dans des milieux confinés vers l'étude d'espèces animales. Dans la seconde partie de ce chapitre, nous reviendrons plus en détail à l'aide de diagramme de modélisation sur les fonctionnalités qui constituent ces trois grandes parties.

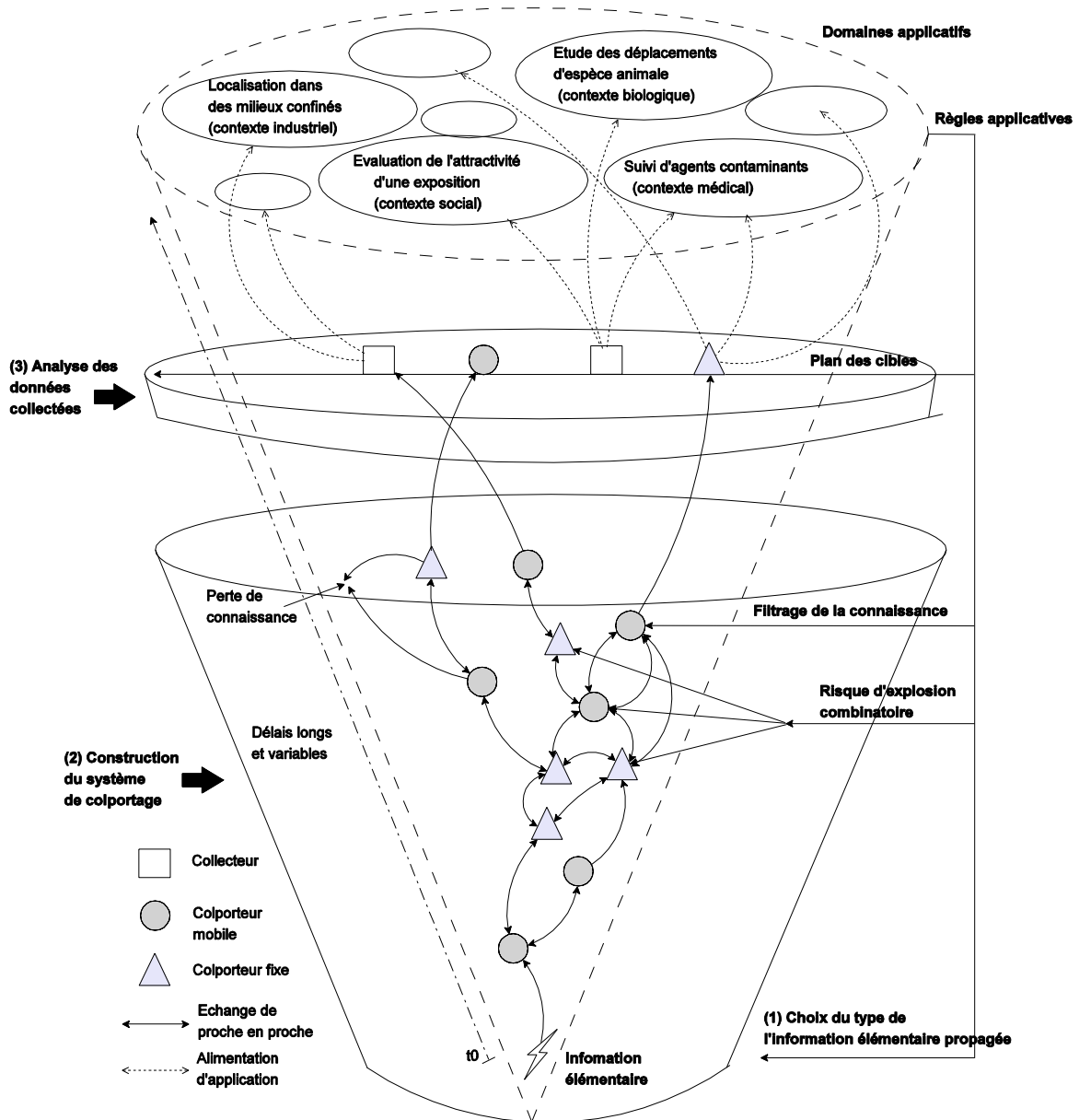


Figure 28 : Principe du colportage dans une population mobile.

### 2.2.3.1. Principe de colportage appliqué à une étude biologique

La figure 29 reprend le principe du colportage dans une population mobile adapté à un contexte biologique où l'application visée est l'étude des déplacements d'une espèce animale. L'exemple donné ici reprend le projet ZebraNet [71] où le comportement des zèbres est étudié afin de comprendre les déplacements de cette espèce. A chaque passage d'un véhicule à proximité d'un animal ou d'un troupeau de zèbres, les données sont enregistrées automatiquement (en supposant que les zèbres partagent une connaissance). Ces données sont ensuite déversées dans des points de collecte afin d'alimenter l'application qui se charge du traitement de ces informations collectées.

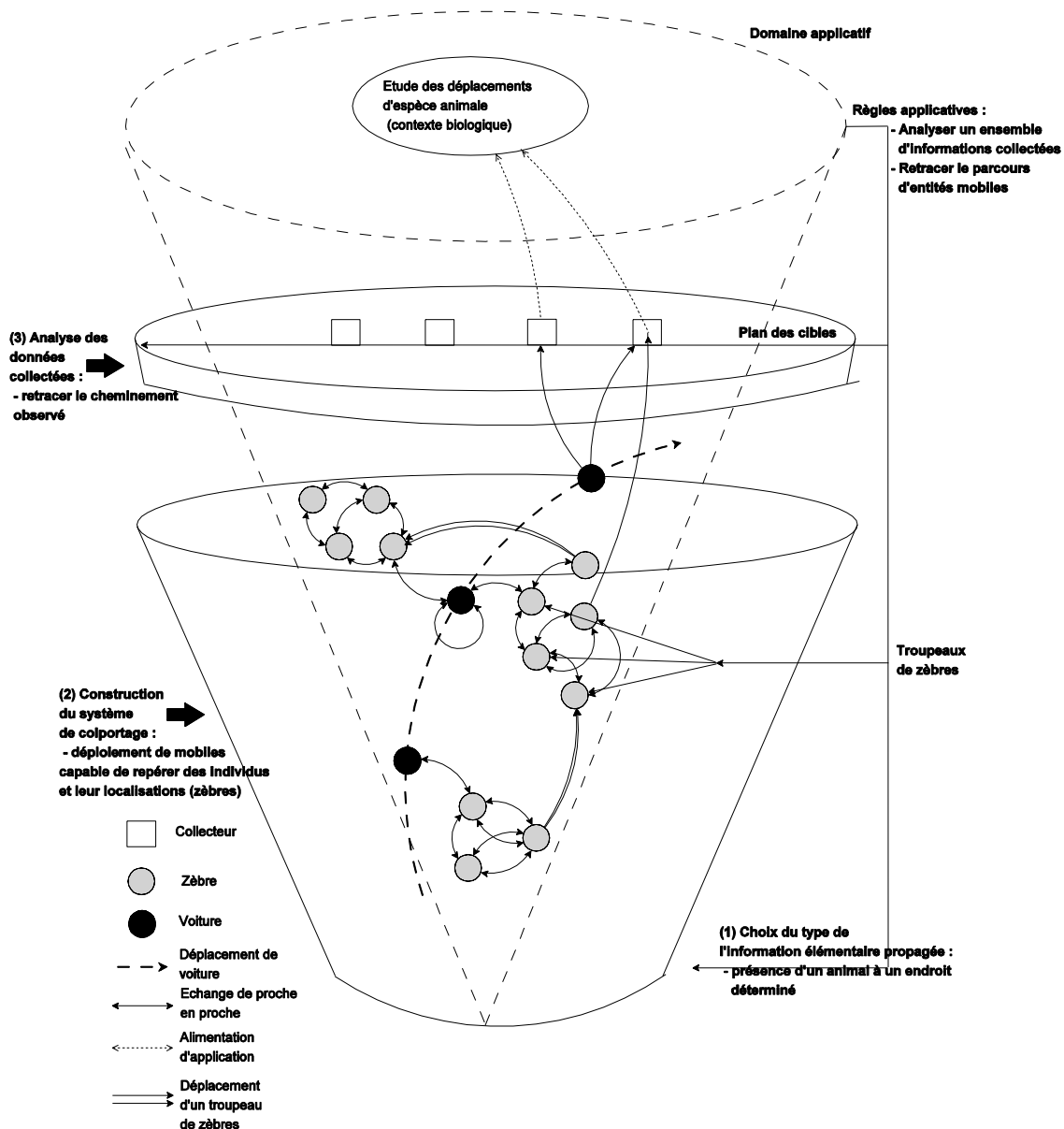


Figure 29 : Principe du colportage appliqué à l'étude des déplacements d'espèce animale.

### 2.2.3.2. Principe de colportage appliqué à un domaine médical

Un autre exemple d'application est donné dans la figure 30. Cette fois-ci le concept de colportage est adapté à une application de suivi d'agents contaminants en milieu hospitalier. L'entité mobile symbolisée en noire représente l'agent contaminant concrètement un nœud capteur porté par un malade ou attaché à un matériel. Chaque rencontre entre cet agent et une personne saine peut induire un processus de transmission de virus selon une probabilité de contamination qui dépend de plusieurs paramètres (immunité, durée d'exposition, distance entre les entités, fréquence d'éternuement... etc.). Dans ce type d'application, chaque entité déployée dans le réseau et ayant été en contact avec l'agent contaminant est potentiellement une cible impliquée dans le processus de colportage (y compris les porteurs sains). L'application finale va ensuite tenter de retracer le parcours de l'agent afin d'évaluer le nombre et l'identité des personnes possiblement contaminées. Cet exemple s'inspire fortement des infections nosocomiales contractées dans des établissements de santé [72]. Ce concept peut aussi être utilisé pour une démarche de simulation.

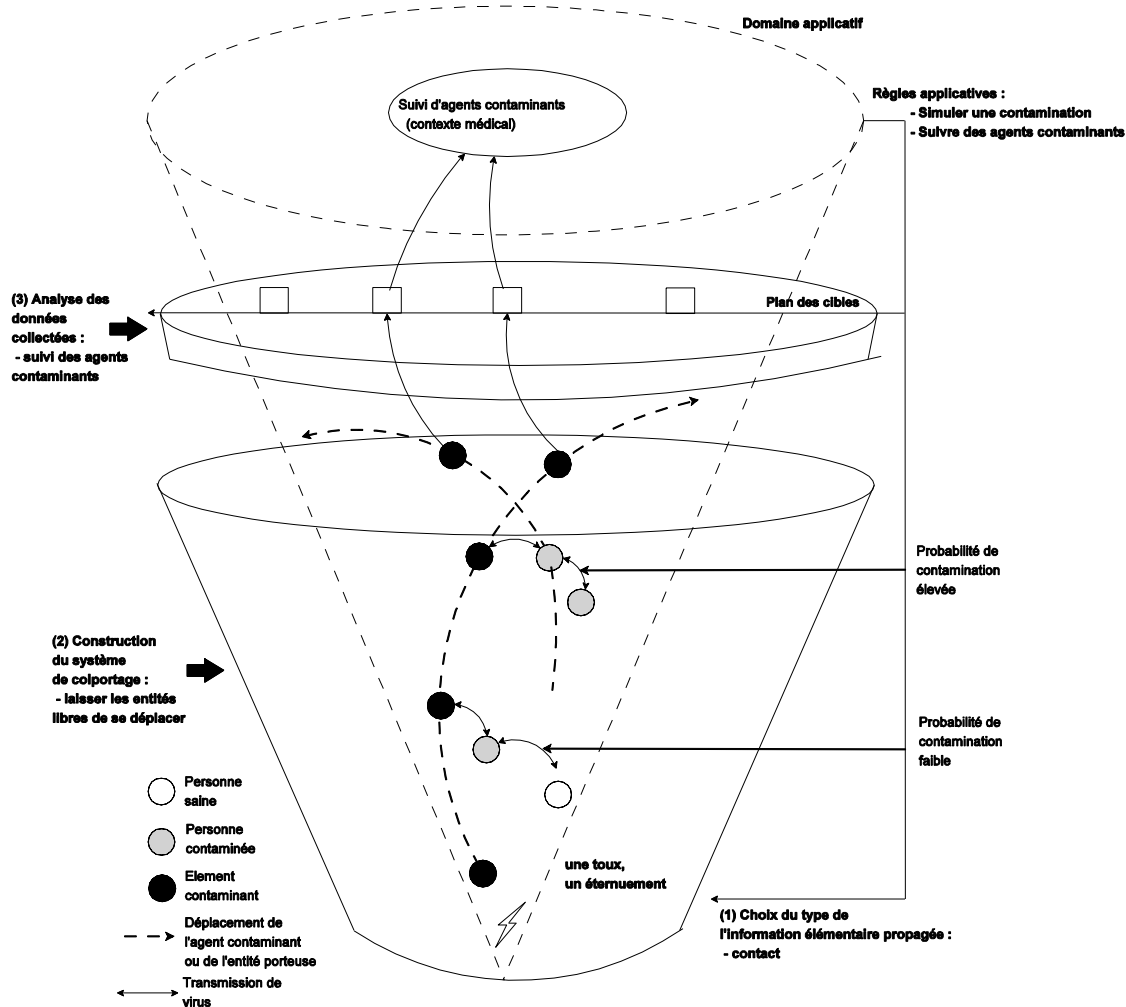


Figure 30 : Principe du colportage appliqué au suivi d'agents contaminants.

### 2.2.3.3. Principe de colportage appliqué à un domaine social

Dans un contexte social cette fois-ci, l'exemple, illustré par la figure 31, montre le déploiement des nœuds et l'exploitation d'un système de colportage pour une application dédiée à la fois à l'évaluation de l'attractivité dans un musée et à l'analyse du comportement des visiteurs d'une exposition par exemple. Plusieurs salles constituent cette exposition et proposent de nombreuses œuvres que les visiteurs peuvent contempler durant leur parcours. Chaque visiteur peut être colporteur d'information. En effet, il peut spécifier qu'il n'était pas le seul à contempler une œuvre en même temps que lui ou avant lui ou au contraire qu'elle a intéressée peu de monde.

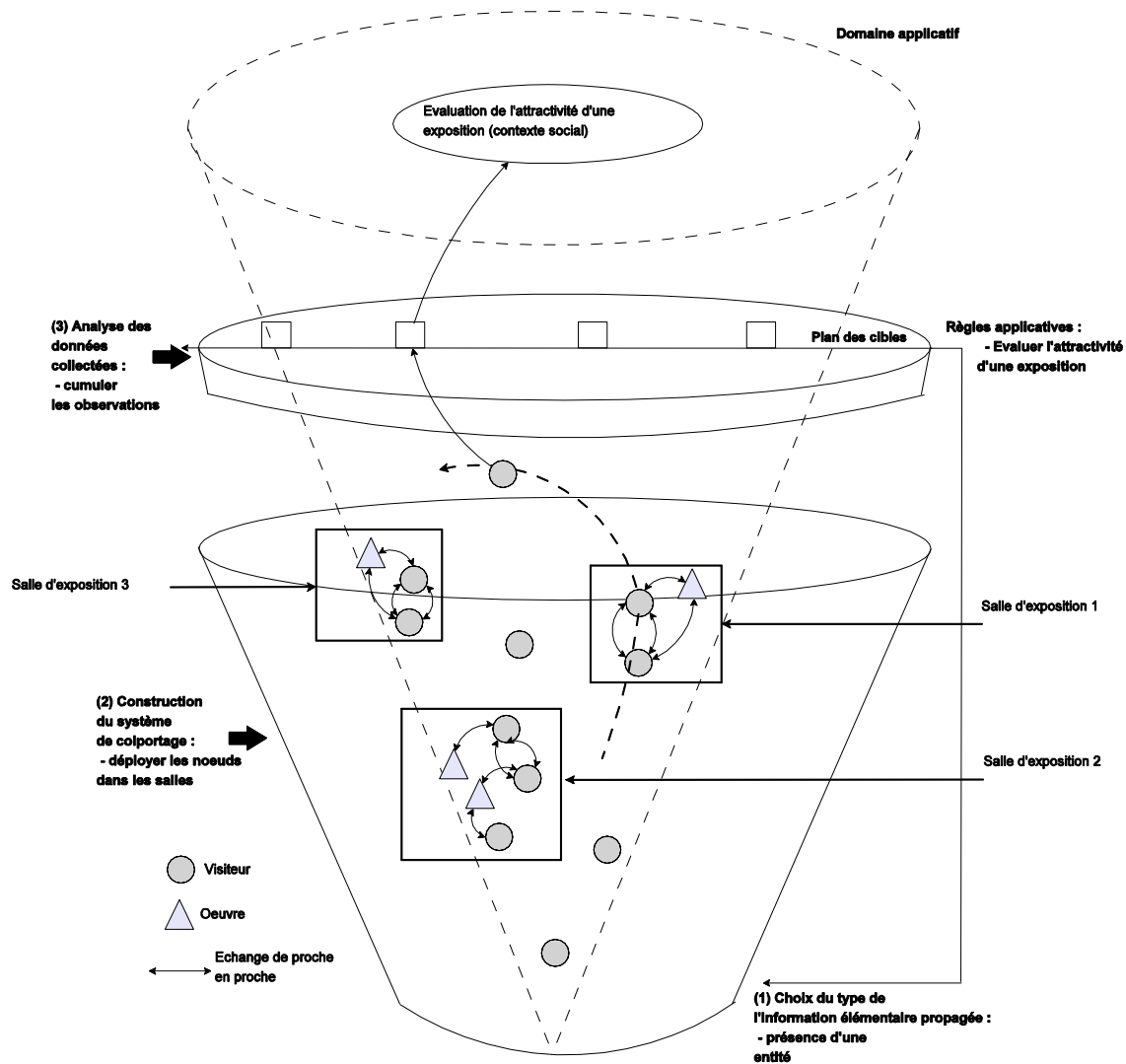


Figure 31 : Principe du colportage appliqué à l'étude de l'attractivité d'une exposition.

### 2.2.3.4. Principe de colportage appliqué à la localisation en milieux confinés

Le dernier exemple proposé dans la figure 32 expose la manière dont le système de colportage est exploité pour des applications de localisation a posteriori dans des milieux confinés.

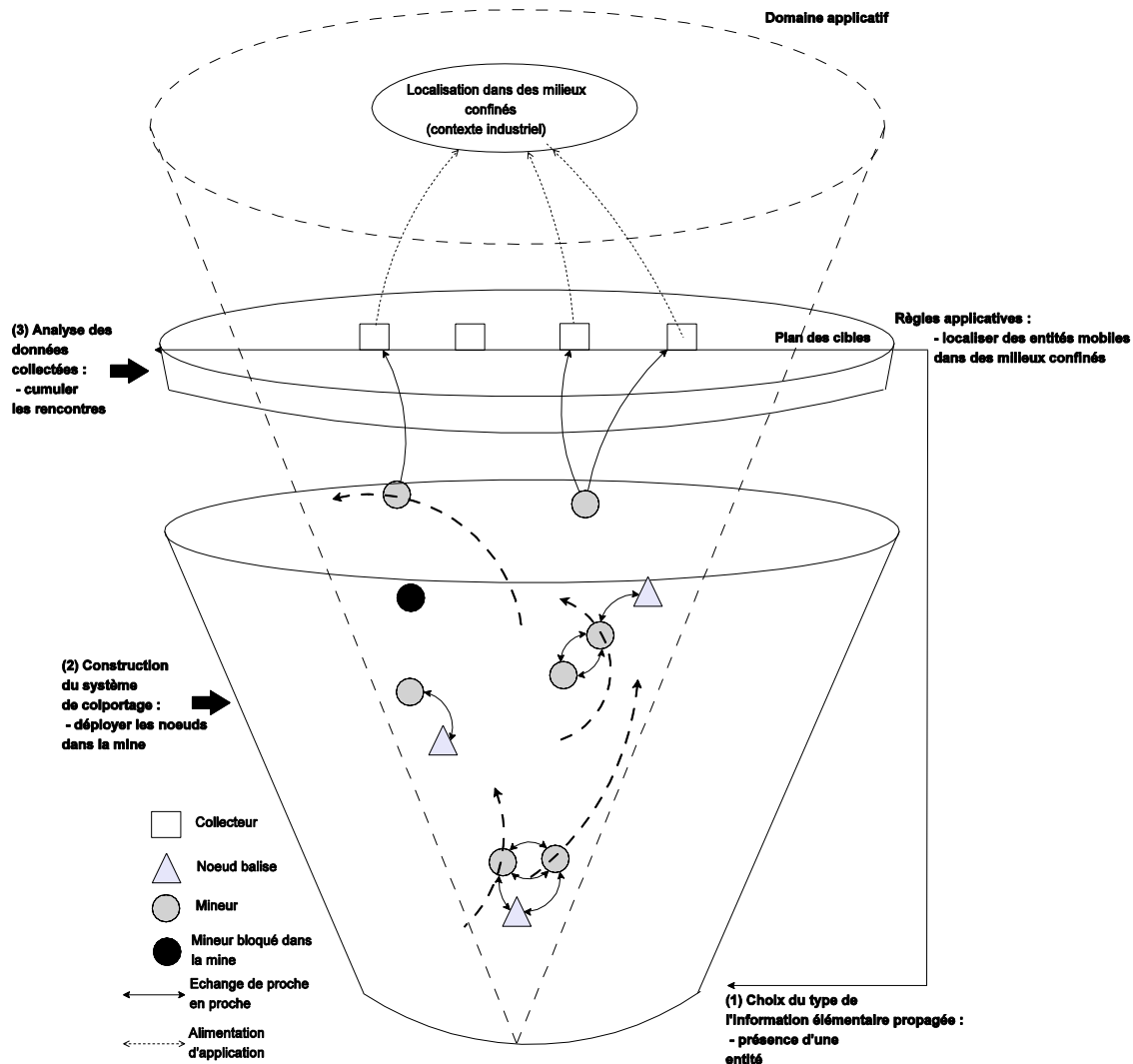


Figure 32 : Principe du colportage appliqué à la localisation dans des milieux confinés.

Des nœuds fixes représentés en triangle appelés *nœuds balises* sont placés en divers points stratégiques. Ils peuvent avoir deux rôles distincts : ils sont à la fois la signature d'une position en jouant ainsi le rôle d'un phare qui donne sa position à des bateaux et peuvent servir de points de relais pour le cheminement de l'information. Lorsqu'il s'agit d'exploiter le colportage dans une galerie de mine par exemple, leur principal rôle est de relayer aux entités en contact, des informations confiées auparavant par d'autres entités mobiles qui sont passées à proximité. Ainsi, il est possible qu'une entité ait des informations relatives à des entités avec lesquelles elle n'a eu aucun contact. Les stations mobiles représentent des mineurs en activité qui se déplacent dans la mine et peuvent échanger de la connaissance de contact afin de l'acheminer vers un ou plusieurs collecteurs. Le nœud mobile

représenté en noir symbolise un mineur bloqué à l'intérieur de la mine que l'application finale tentera de localiser a posteriori.

Ce type de réseau est aussi caractérisé par des délais de transit longs et très souvent variables, il fait partie des réseaux dits tolérants aux délais, pour lesquels l'information peut prendre des heures voire des jours pour être routée vers la destination finale. Notre stratégie s'inspire et tente d'enrichir le concept dit trois étages déjà présenté dans la bibliographie à propos du concept *data MULEs* [50]. Trois types de modules sont donc proposés dans notre stratégie :

- Les nœuds colporteurs mobiles : ces entités ont à la fois le potentiel de se déplacer, d'échanger et de stocker de la connaissance.
- Les nœuds colporteurs fixes : entités fixes appelées aussi balises et éventuellement placées en des points stratégiques. Ces nœuds jouent le rôle de signature de position et de relais d'information pour les entités mobiles qui passent à proximité.
- Les collecteurs : entités fixes ou mobiles jouant le rôle de dépôt de données lors des passages à proximité des entités de colportage et servant à alimenter les applications ciblées. Lorsque les collecteurs sont mobiles, ces entités peuvent par exemple se déplacer pour récolter la connaissance stockée au niveau des points relais (balises).

Notre architecture enrichit le concept data MULE en modifiant le rôle des entités, dans les exemples qui viennent d'être donnés la notion de MULE est répartie sur l'ensemble des colporteurs qui est sensiblement l'ensemble des nœuds du réseau de capteurs.

### **2.3. Modèle générique**

Après avoir exposé une première vision globale de l'approche du colportage. Nous allons à présent à l'aide d'un outil de modélisation, analyser plus en détail, les acteurs ainsi que leur rôle dans le système du colportage. Notre choix se dirige vers le langage UML (Unified Modeling Language) [73], un langage de modélisation du domaine du génie logiciel et couramment utilisé dans des projets logiciels. Il peut être appliqué à toutes sortes de systèmes.

Ce choix est justifié d'abord par le fait qu'UML est devenu un standard incontournable dans le domaine de la modélisation orientée objet mais aussi parce qu'il va permettre de mieux visualiser notre système en précisant les composants, leurs interactions et l'utilisation qu'il est possible d'en faire.

UML dans sa version 2.3 [73] propose treize diagrammes avec des fonctionnalités différentes, ces diagrammes sont dépendants hiérarchiquement et se complètent pour la modélisation d'un projet quelconque tout au long de son cycle de vie. Ces diagrammes peuvent être classés en trois catégories : structurels (statiques), comportementaux et d'interactions (dynamiques). Nous nous attarderons sur

l'explication de trois diagrammes seulement qui seront utiles pour la modélisation du processus de colportage et qui sont les suivants :

- Diagramme de cas d'utilisation : il permet de représenter les interactions possibles entre le système et les intervenants extérieurs à celui-ci (acteurs externes) mais aussi les utilisations internes.
- Diagramme de classes (statique) du modèle de domaine : ce diagramme permet de représenter les objets métiers qui constitue notre système. Il permet d'identifier des objets, leurs propriétés et leurs comportements par une structuration en classes.
- Diagramme de séquence (dynamique) : il fait suite au diagramme de cas d'utilisation en permettant pour chacun d'eux de spécifier une représentation séquentielle du déroulement des traitements ainsi que les interactions entre les acteurs du système.

### **2.3.1. Orientations fonctionnelles d'un système de colportage**

Nous avons choisi d'aborder la présentation d'un système de colportage par son utilisation afin de faire apparaître ses orientations fonctionnelles. Trois diagrammes de cas d'utilisation sont présentés ici afin d'illustrer le fonctionnement des trois principales couches qui constituent le système de colportage : le choix du domaine applicatif, la construction du système de colportage et l'analyse des données collectées.



### 2.3.1.1. Le choix du domaine applicatif

Nous proposons d'abord un diagramme pour présenter l'utilisation d'un système de colportage qui pourrait être faite par l'administrateur pour l'initialiser et préparer sa configuration par rapport au domaine applicatif. Comme nous l'avons défini dans la première partie de ce chapitre, quatre domaines sont visés par notre thématique : industriel, médical, biologique et social. Pour chaque contexte, un exemple d'application a été proposé en adaptant le contenu de la figure 28 illustrant le concept de colportage. L'identification du domaine se fait en tenant compte d'orientations basées sur ce qui est attendu. Un ensemble de question pourrait être posée au quelle l'administrateur devra répondre : Que simuler ? Une diffusion ? Une contamination ?

Le diagramme proposé par la figure 33 regroupe ces différentes applications selon un autre type de classification qui est la façon dont l'information est propagée dans le réseau.

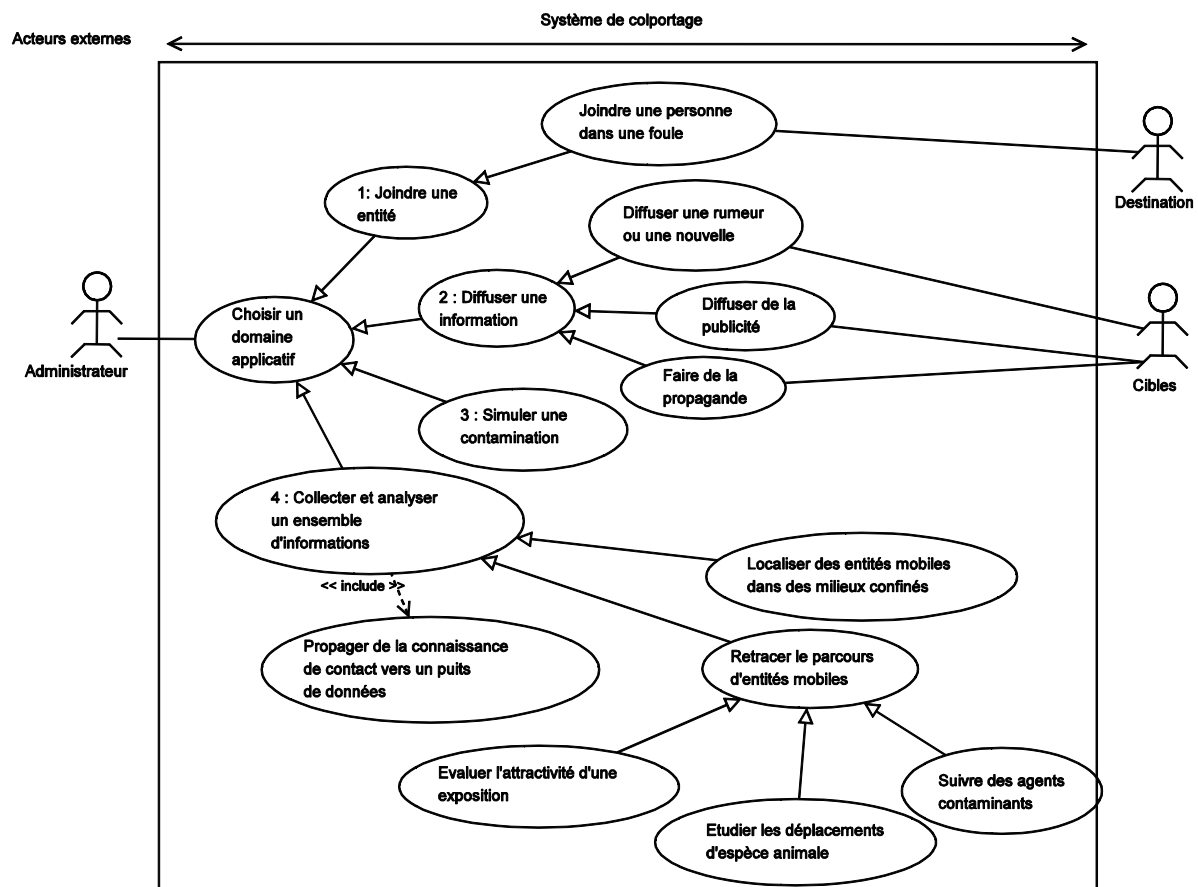


Figure 33 : Diagramme de cas d'utilisation pour le choix du domaine applicatif.

En effet, les nœuds déployés dans les quatre exemples applicatifs impliquant le colportage propagent leur connaissance au sein du réseau avec un but commun qui est d'atteindre ou de router cette information vers des points de collecte afin d'alimenter les applications visées. Il est possible de considérer une application selon son domaine mais également selon la façon d'échanger les données

entre entités, en prenant en compte la façon d'impliquer les entités réceptrices ou destinataires d'un échange. Différentes classes d'applications sont proposées ici afin d'élargir notre travail en se basant sur les modèles suivants :

1. Joindre une entité particulière : ce principe de transmission correspond à faire partir une information élémentaire d'une entité et de chercher à la transmettre à une cible bien particulière. Les nœuds impliqués dans cette catégorie d'échange contribuent à des communications de type *1 vers 1*. Un exemple est donné dans le diagramme de la figure 33 où une personne tente de joindre une personne au milieu d'une foule.
2. Diffuser une information : cette façon de communiquer est assez courante, nous pouvons citer à titre d'exemple la diffusion d'une nouvelle, d'une rumeur, faire de la publicité ou de la propagande. Les communications entre les nœuds du réseau peuvent être de type *1 vers n*.
3. Simuler une propagation : la dernière catégorie proposée ici est une catégorie particulière qui permet de jouer ou de simuler un phénomène de propagation d'une information quelconque. L'exemple le plus significatif dans cette catégorie est la simulation d'un processus de contamination par une épidémie ou les communications sont de types *n vers n*.
4. Les mécanismes proposés dans cette thèse se focalisent seulement sur des applications dont le but est de propager de la connaissance de contact afin d'atteindre un ou plusieurs puits de données. Cette connaissance peut éventuellement servir à des applications de localisation a posteriori d'entités mobiles. Aussi, nous nous appuyerons sur une application générique qui consiste à localiser des mineurs et/ou du matériel à l'intérieur des galeries d'une mine souterraine. Cet exemple d'application cible dédié à la localisation dans ces milieux confinés sera détaillé et évalué dans les deux chapitres qui vont suivre.

### 2.3.1.2. La construction du système de colportage

Le diagramme de cas d'utilisation illustré dans la figure 34 développe la seconde phase du processus de colportage qui consiste en la construction du système de colportage.

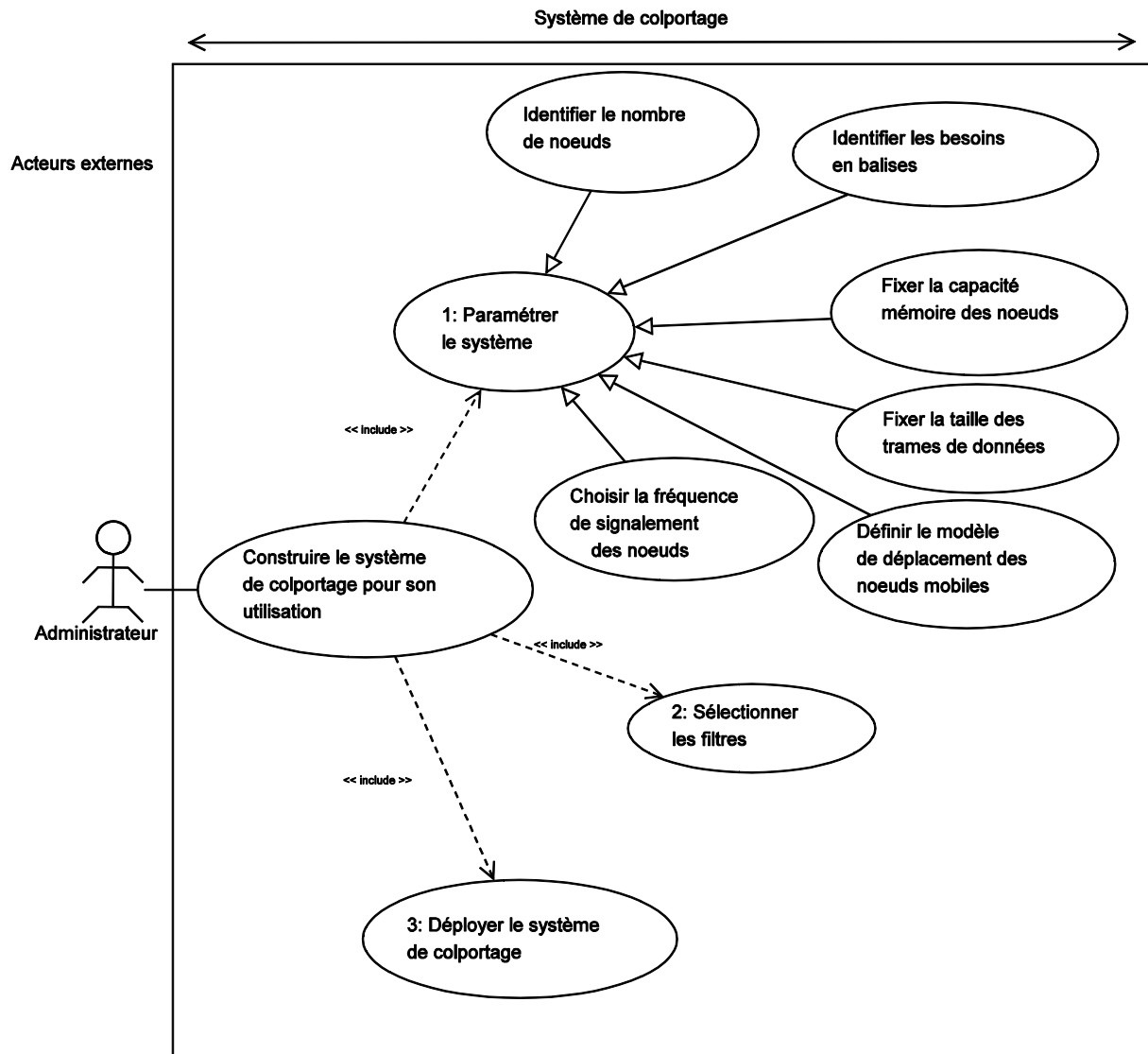


Figure 34 : Diagramme de cas d'utilisation pour la construction du système de colportage.

La construction du système va permettre sa définition et son utilisation suite au choix du domaine applicatif mais cela ne peut se faire sans certains points de passage obligatoires :

- Le paramétrage du système : cette étape est constituée de plusieurs tâches indispensables pour l'exécution du processus de colportage et qui consiste à fixer :
  - Le nombre de mobiles à déployer au sein du réseau.
  - Le nombre et la position des balises à déployer.
  - La capacité mémoire allouée pour chaque station.
  - La taille des trames de données.

- Le modèle de déplacement des stations mobiles.
- La fréquence de transmission des trames de signalement.
- La sélection des filtres : la construction du système de colportage nécessite le choix des filtres qui doivent être appliqués ensuite par les nœuds du réseau afin d'acheminer leur connaissance vers les points de collecte. Deux classes de filtres peuvent être distingués : les filtres permanents (ou anti-redondance) et les filtres applicatifs. La première classe de filtre est appliquée de manière permanente par tous les nœuds afin d'éviter de stocker des informations qu'ils connaissent déjà. La seconde classe concerne les filtres applicatifs qui seront choisis selon l'application visée. Dans le chapitre 3 de cette thèse, nous reviendrons plus en détail sur les fonctionnalités de toutes les stratégies de filtrage.
- Le déploiement du système : cette étape finalise le processus de colportage en appliquant tous les paramètres définis auparavant à la solution déployée.

### **2.3.1.3. L'analyse des données collectées**

Le dernier cas d'utilisation représenté dans la figure 35 correspond à l'étape finale du processus de colportage. S'il y a eu utilisation d'un puits de données, il faut analyser les données collectées. Cette étape doit dans un premier temps mettre en forme les données collectées. Ensuite, le système a la possibilité d'appliquer d'éventuels filtres supplémentaires avant de traiter les données collectées selon le type de l'application finale : joindre une entité, diffuser une information, simuler une contamination ou atteindre un puits de données par exemple.

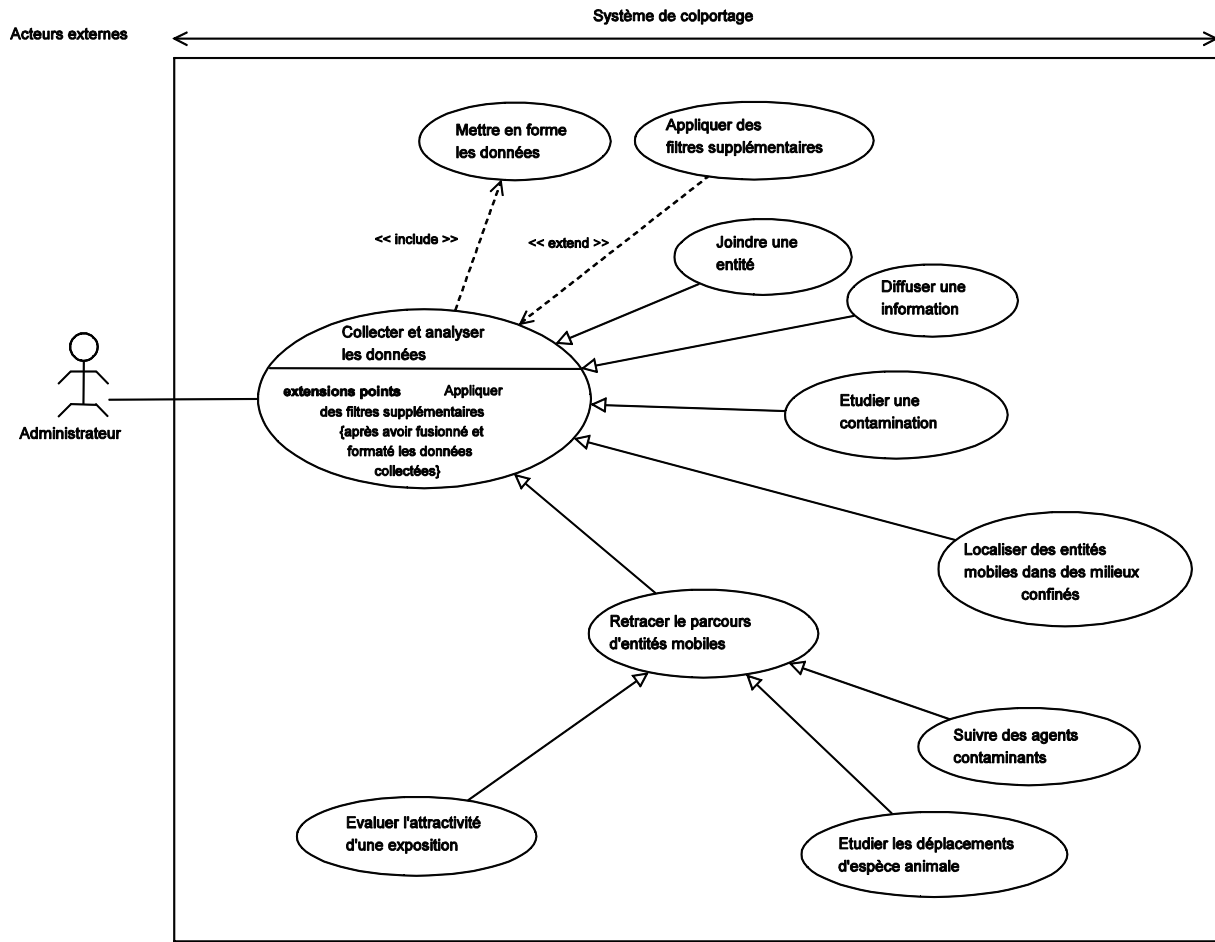


Figure 35 : Diagramme de cas d'utilisation pour l'analyse des données collectées.

Cette étape doit dans un premier temps mettre en forme les données collectées. Ensuite, le système a la possibilité d'appliquer d'éventuels filtres supplémentaires avant de traiter les données collectées selon le type de l'application finale.

#### 2.3.1.4. Cas d'utilisation pour une application de localisation a posteriori

La modélisation en trois temps présentée dans ce chapitre (choix du domaine, construction du système et analyse des données) permet d'avoir une vue générique de la mise en place d'un système de colportage. Ce modèle va pouvoir être adapté à chaque type d'application en sélectionnant certains cas en priorité. Un exemple est donné dans la figure 36.

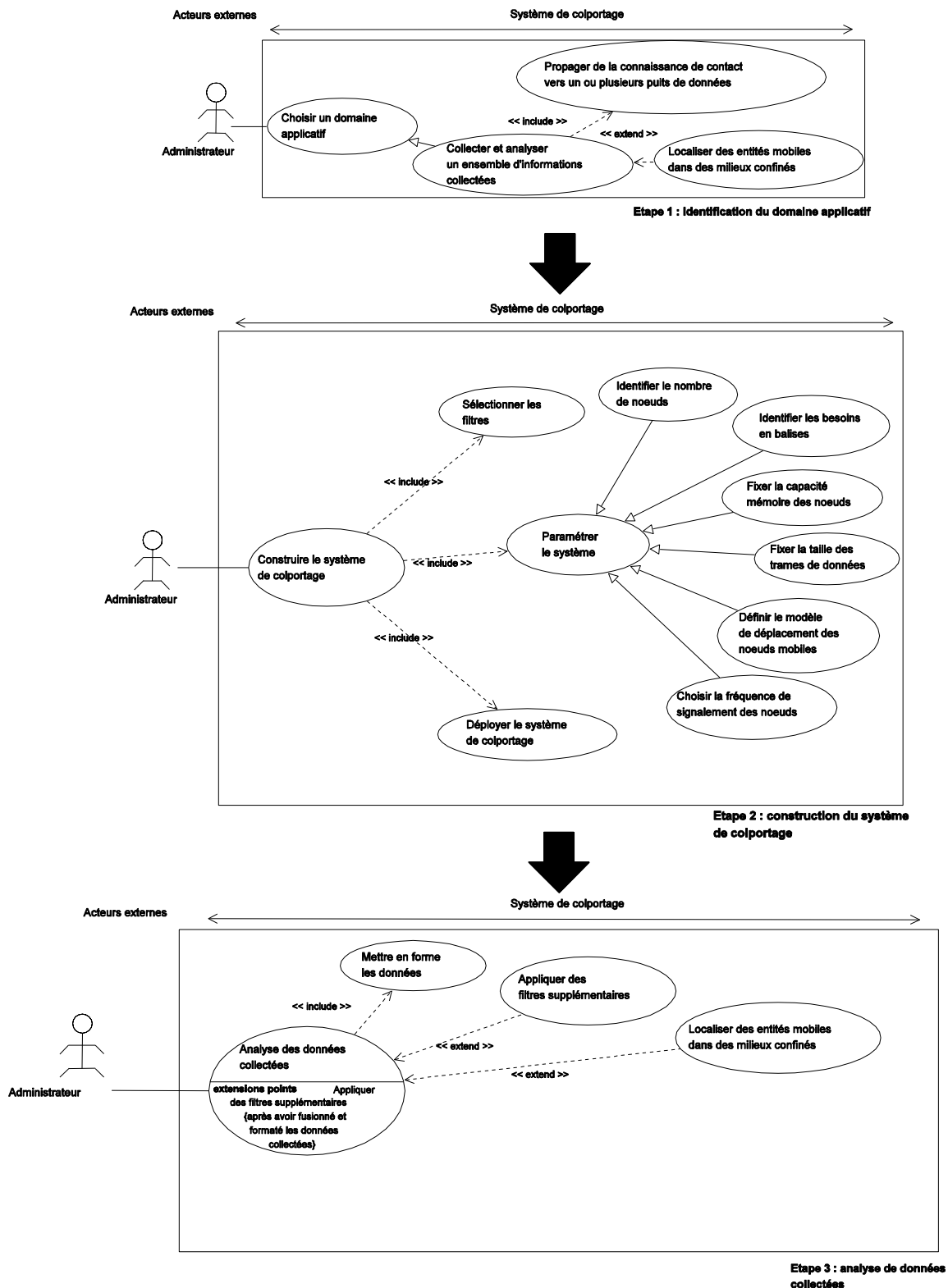


Figure 36 : Exemple d'un cas d'utilisation pour une application de localisation a posteriori.

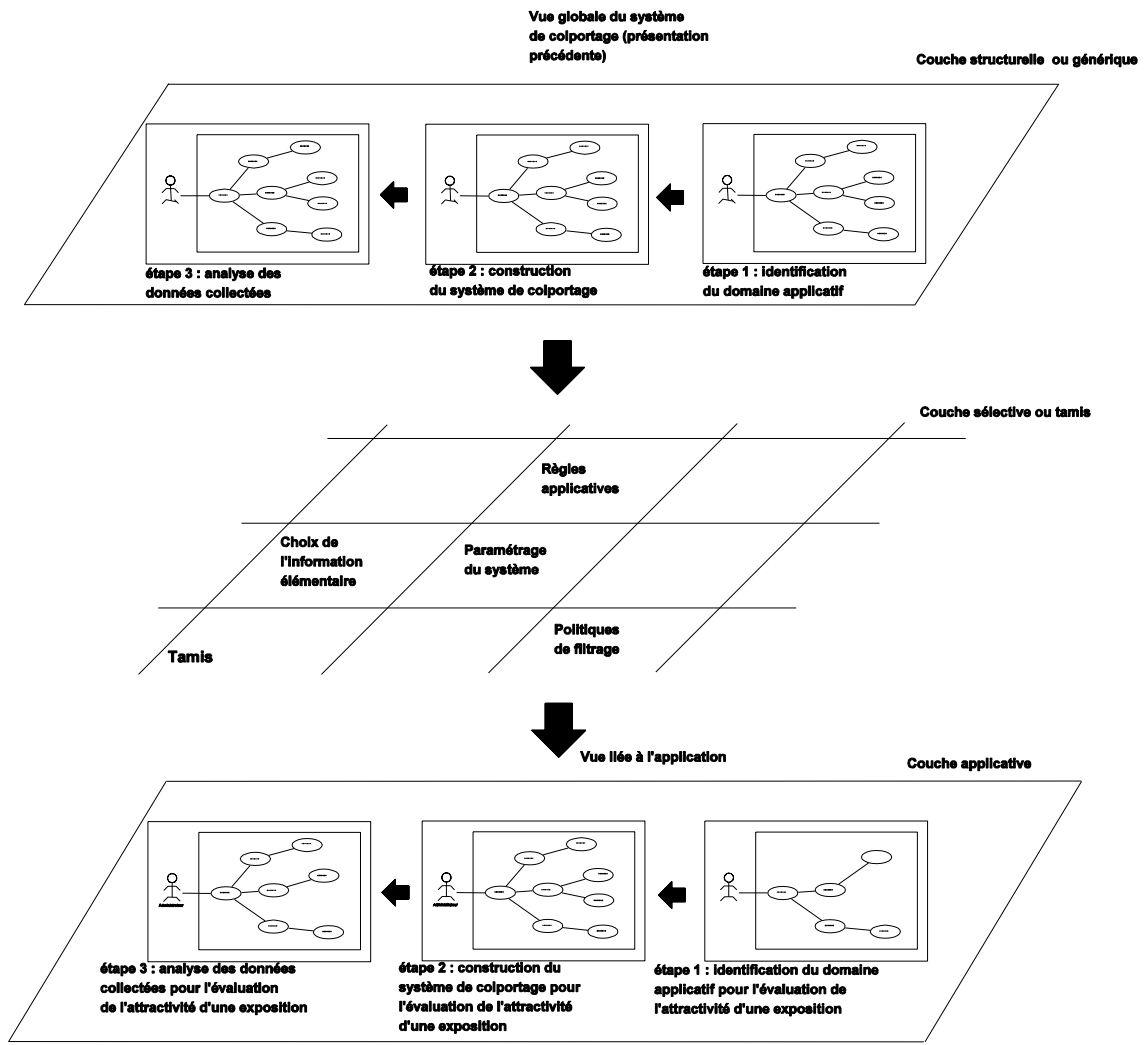
Nous reprenons dans cette modélisation notre exemple de localisation a posteriori d'entités mobiles dans des milieux confinés. Nous estimons que pour traiter notre exemple, il faut sélectionner certains cas. Chaque cas d'utilisation va adapter ses fonctionnalités selon l'application visée. Dans cet exemple, la politique de filtrage des événements de contact dans la seconde étape va prendre en

considération la fraîcheur de l'information comme paramètre prioritaire afin d'affiner le processus de localisation.

### **2.3.1.5. Représentation en couche du modèle de colportage**

En tenant compte des différents points exposés, nous proposons une représentation supplémentaire du modèle de colportage afin d'apporter une vision complémentaire du modèle générique et de son adaptation. Cette représentation qui sera appelée *représentation en couche*, est illustrée en détail dans la figure 46 et se compose de trois couches essentielles qui sont les suivantes :

- Couche structurelle : c'est la couche initiale et la plus générique dans cette représentation, son rôle est de satisfaire les cas d'utilisation définis plus haut dans ce mémoire et les considérer dans une seule étape. Le but escompté est de donner une vue globale du système de colportage en regroupant les principales fonctionnalités du système, c'est-à-dire : l'identification du domaine applicatif, la construction du système, et l'analyse des données collectées. Toutes ces fonctionnalités sont ensuite soumises à la couche sélective pour être traitées selon le contexte de l'application finale.
- Couche sélective : cette couche appelée aussi 'tamis', joue le rôle d'un tamis pour les services offerts par la couche supérieure en sélectionnant seulement les fonctionnalités indispensables au domaine applicatif ciblé (exp : choix de l'information élémentaire, règles applicatives, politiques de filtrage, paramétrage du système, ...etc.). Seuls en ressortiront les cas d'utilisation utiles au développement de l'application finale. Cette couche peut être notamment vue comme un système intelligent utilisant des règles de production basées sur un comportement à adopter selon différents contextes concernés.
- Couche applicative : le rôle de cette couche est de finaliser le processus de colportage en appliquant les besoins et les contraintes imposées par la couche supérieure au modèle de colportage. La figure 37 illustre un exemple lorsque le domaine visé se situe dans un contexte social où l'application finale tente d'évaluer l'impact du nombre de visiteurs dans un musée sur l'attractivité d'une œuvre particulière.



**Figure 37 : Représentation en couche du processus de colportage.**



### 2.3.2. Modèle du domaine

Le système de colportage s'appuie sur différents acteurs dont des colporteurs fixes ou mobiles et des collecteurs que l'on va considérer comme étant des nœuds. La figure 38 montre qu'un nœud possède un rôle dans le système de colportage. Il peut être soit colporteur, soit collecteur d'un ensemble d'événements de contact. Deux classes de nœuds se distinguent selon leur déplacement : les nœuds mobiles et les nœuds fixes. Les nœuds mobiles ont pour rôle de récolter, d'échanger et de stocker des événements de contact durant leur parcours. Les nœuds fixes sont utilisés comme des relais d'information dans le processus de colportage. Ce diagramme de classe montre aussi la relation entre un nœud et la connaissance. En effet, chaque nœud possède un ou plusieurs événements de contact stockés dans sa mémoire qu'il peut éventuellement manipuler et colporter à travers le réseau. La dernière fonctionnalité représentée dans ce diagramme de classe est la notion de filtrage de la connaissance, ou plus précisément des événements de contact stockés en mémoire. Chaque station du réseau peut appliquer des filtres pour libérer de l'espace mémoire selon plusieurs critères. Ces derniers peuvent être permanents (anti-redondance) ou bien appliqués seulement lorsque la mémoire est saturée. Certains filtres sont utilisés pour répondre aux besoins des applications visées ou tout simplement utilisés pour éviter la redondance des événements. Nous détaillerons dans le chapitre qui va suivre de manière exhaustive la liste de tous les filtres impliqués dans le processus de colportage.

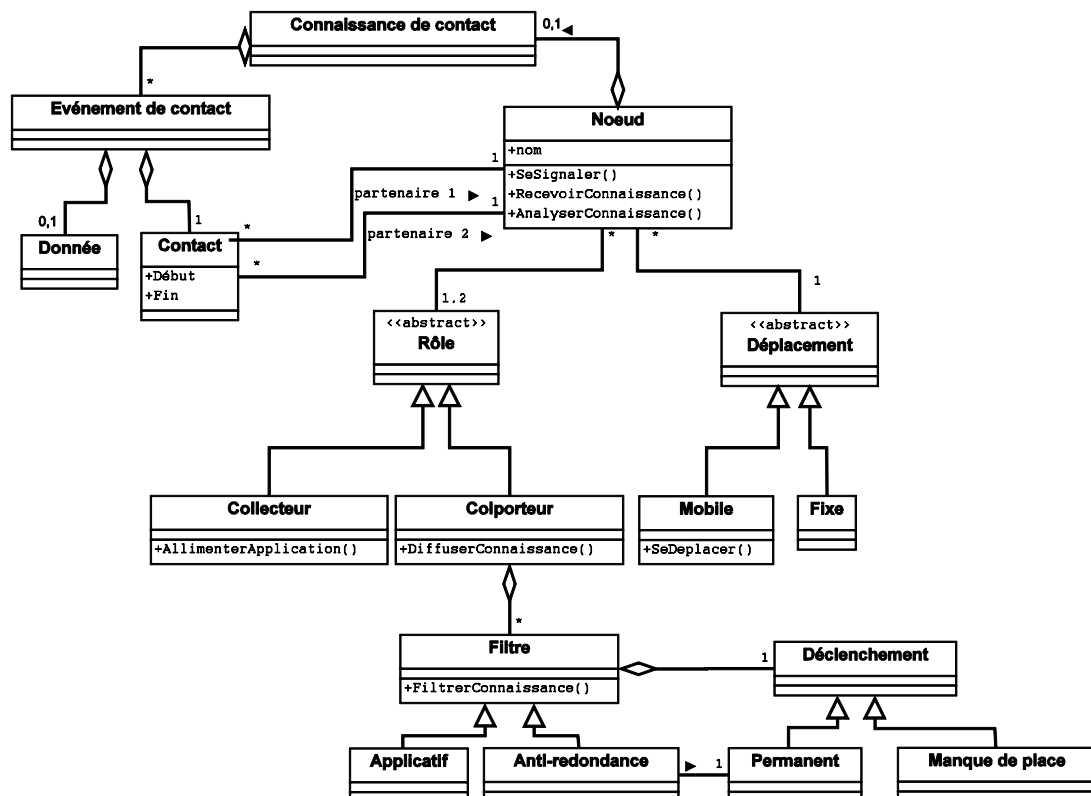


Figure 38 : Diagramme de classes d'un système de colportage.

Ce schéma montre aussi les comportements que possède chaque classe. Il en regroupe également les principales fonctionnalités :

- Se signaler : chaque nœud quel que soit son type et le type d'application visé, se signale périodiquement en transmettant son identité.
- Recevoir connaissance : lors d'une rencontre, un partenaire peut solliciter son voisin pour recevoir sa connaissance.
- Analyser connaissance : selon l'application visée, les nœuds du réseau peuvent analyser la connaissance collectée pour en optimiser l'occupation mémoire par exemple ou pour apporter certains résultats en cours de processus.
- Diffuser sa connaissance : à chaque rencontre les nœuds peuvent diffuser une partie ou la totalité de leur connaissance. Le collecteur est le seul nœud qui ne diffuse pas sa connaissance aux autres nœuds du réseau étant donné son rôle de puits de l'information. Il va alimenter les applications par tout ce qu'il a reçu.
- Alimenter application : le collecteur a pour rôle de récolter toutes les données déversées et de les transmettre éventuellement après une mise en forme des résultats à la ou aux applications exploitant ce processus de colportage.
- Se déplacer : les nœuds mobiles ont la possibilité de se déplacer selon un parcours aléatoire ou déterminé.
- Filtrer connaissance : lors du stockage en mémoire, les nœuds ont la possibilité de décider localement de filtrer les informations à leur disposition selon divers critères dépendant le plus souvent des applications visées.

### 2.3.3. Diagramme de séquence

Il est possible d'analyser chaque cas d'utilisation plus en détail. Pour illustrer cette faisabilité nous avons choisi le cas d'utilisation « construire le système de colportage » (cf. 2.3.1.2), une fois que le système de colportage a été déployé, qu'il est mis en œuvre et qu'il est en état de fonctionnement. Lorsque deux entités fixes ou mobiles sont en contact, un échange peut se dérouler. Le diagramme de séquence est ici utilisé pour représenter de manière séquentielle le déroulement d'un scénario d'échange mettant en scène deux nœuds mobiles, une balise fixe et un collecteur. Le scénario de cet échange est représenté dans la figure 39. Dans cette figure il est à noter que m1, m2, sont des nœuds de type colporteur, b1 est de type balise et c1 est un nœud de type collecteur (cf. 2.3.2). La station b1 n'effectue aucun déplacement (déplacement fixe), m1 et m2 effectuent différents déplacements (mobiles).

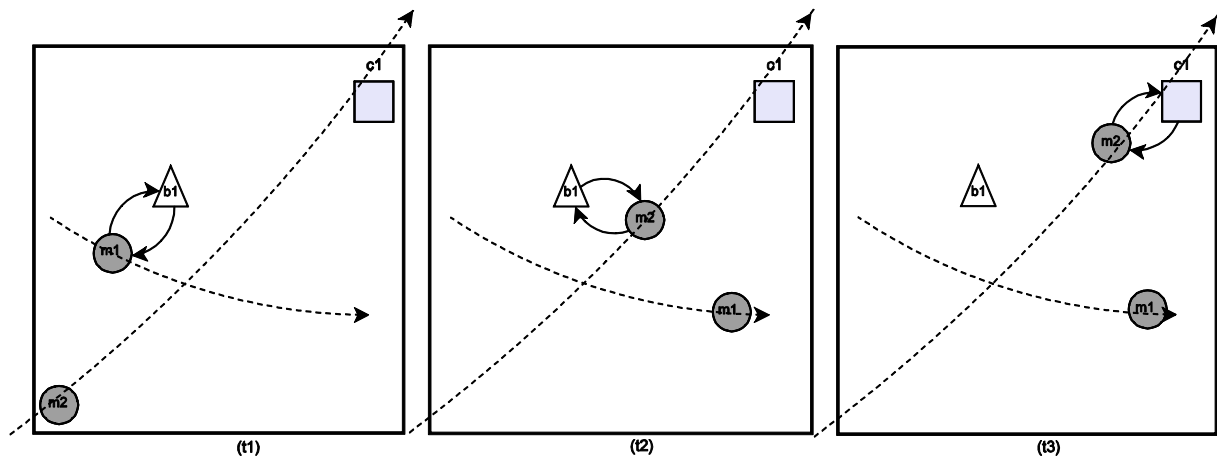
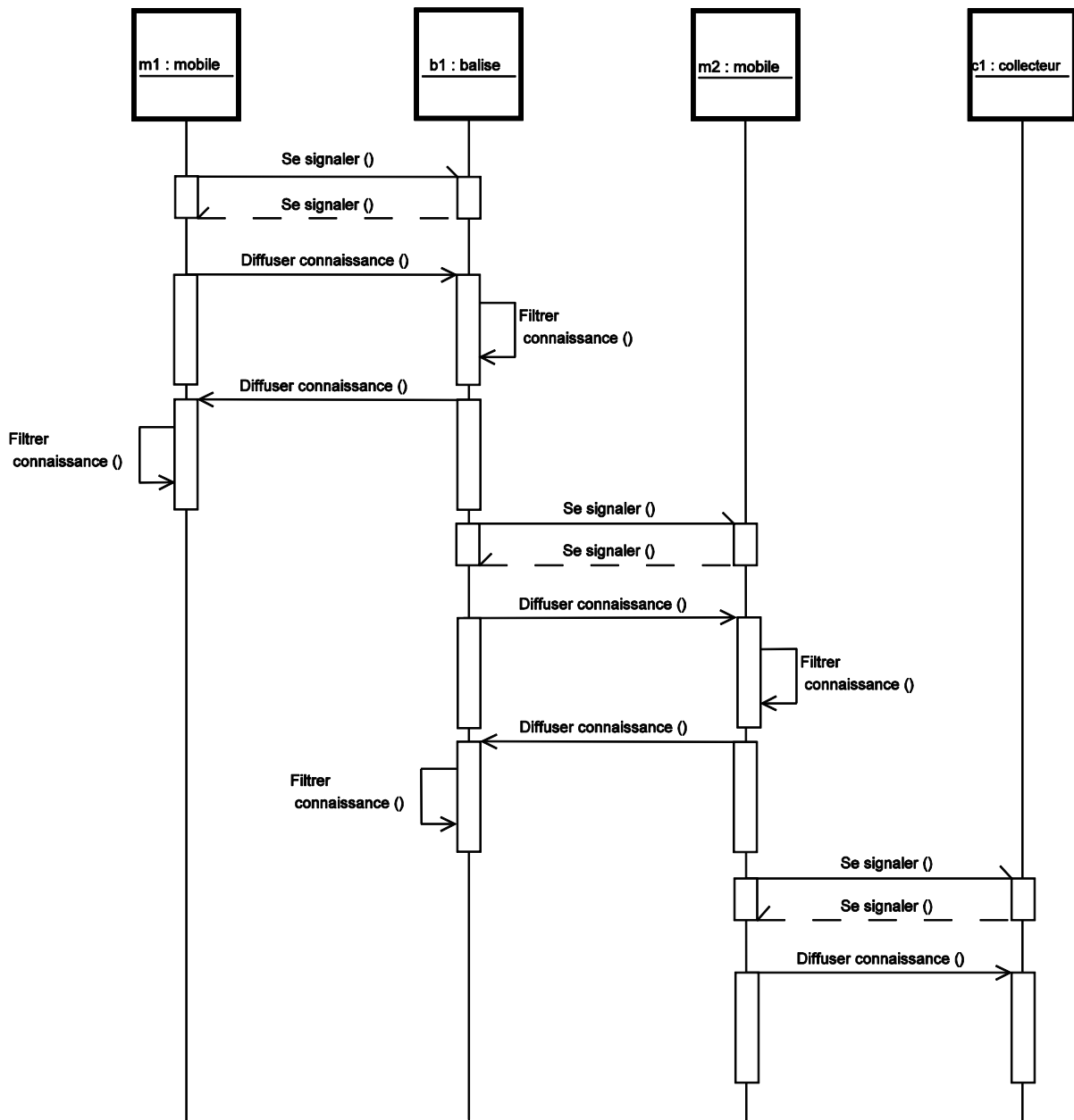


Figure 39 : Scénario d'échange.

Le scénario illustré dans cette figure montre qu'un premier échange a lieu entre l'entité mobile m1 et la balise b1 lorsque ce mobile traverse la zone de détection de b1 (instant t1), il y a création d'un événement de contact de part et d'autre. Quand le second mobile m2 passe à son tour à proximité de la balise b1 (instant t2), les deux entités créent un événement contact de part et d'autre mais aussi m2 récupère les informations échangées entre m1 et b1 lors du premier contact (le plus souvent une partie) et va ensuite les colporter vers le nœud collecteur c1 (instant t3).

Le diagramme de séquence proposé dans la figure 40 illustre les échanges de la connaissance du mobile m1 et ceux du mobile m2 suivant le scénario établi dans la figure 39.



**Figure 40 : Diagramme de séquence pour l'échange de connaissance.**

On constate sur ce diagramme de séquence que le nœud mobile m1 et la balise fixe b1 se signalent périodiquement et échangent leur connaissance lorsqu'ils entrent en contact (instant t1). Les procédures utilisées respectivement pour le signalement et l'échange de connaissance sont : *Se signaler ()* et *Diffuser connaissance ()* (figure 40). Nous définissons une trame de signalement comme étant une trame qui contient seulement l'identifiant du nœud émetteur et servant seulement à signaler sa présence. Les nœuds se signalent tant qu'ils ne détectent aucun nouveau partenaire. Ensuite, lorsque le nœud mobile m2 entre en contact à son tour avec la balise b1, il récupère les événements de contact échangés lors du premier contact, que la balise a choisi de mémoriser. Enfin, ce mobile déverse l'intégralité de la connaissance qu'il transporte au niveau du collecteur. Remarquons un point qui sera détaillé dans la suite, la nécessité de choisir la connaissance pour des raisons de capacité mémoire

notamment, fait que la connaissance collectée est un extrait de l'ensemble des événements de contact généré au gré du déplacement des entités.

## **2.4. Conclusion**

Ce chapitre avait pour but de donner une première image du concept de colportage dans une population mobile. Quelques notions fondamentales ont été présentées pour définir le vocabulaire propre à ce système. Le fonctionnement de base de ce système fondé particulièrement sur la propagation d'une information atomique et sur des décisions prises localement par des nœuds de types capteurs a été modélisé notamment via l'usage d'UML. Cette description se veut générale et non exhaustive. Le modèle proposé ici doit prendre en considération l'ensemble des domaines applicatifs traités dans le premier chapitre de cette thèse afin de proposer un modèle conceptuel générique pour les réseaux dits tolérants aux délais. Dans le chapitre suivant, nous expliquerons de manière détaillée le processus de colportage et plus précisément le mécanisme de routage basé sur des stratégies de filtrage de la connaissance en prenant comme cible notre application générique.

# Chapitre 3 : LoWCA (Exemple d'application cible)

## 3.1. Introduction

Dans ce troisième chapitre, nous abordons en détail tous les aspects du concept de colportage en nous appuyant sur une application cible appelée LoWCA (Localization and tracking techniques using a Wireless sensor network in Confined Areas). LoWCA est une adaptation du mécanisme de colportage de données destinées à des applications de localisation et de traçabilité d'une population d'entités mobiles. Cette solution s'adresse plus particulièrement aux applications pour lesquelles des entités évoluent en milieux confinés, et communiquent en utilisant des nœuds de réseaux de capteurs sans fil. Ce mécanisme repose essentiellement sur la stratégie de filtrage des données colportées par les nœuds du réseau, basée sur une panoplie de filtres selon les besoins de l'application finale.

De plus, un modèle de déplacement sera proposé dans le but de reproduire les déplacements induits par des activités en milieux confinés comme dans une mine industrielle par exemple. Enfin, une partie expérimentale et une partie applicative concluront ce chapitre pour consolider notre contribution.

## 3.2. Entités du réseau

La stratégie de partage et d'échange de connaissance par colportage est construite sur trois catégories de nœuds : les nœuds mobiles, les nœuds balises et les nœuds collecteurs [74].

### 3.2.1. Les nœuds mobiles

Typiquement ce sont les entités porteuses de ces nœuds que l'application cherche à localiser ou à tracer le parcours. Chaque nœud mobile (ou colporteur mobile) a la capacité d'être déplacé selon une trajectoire le plus souvent contrainte par le milieu confiné (dans une mine il est difficile de ne pas suivre la galerie). Cette trajectoire sera spécifiée et/ou modélisée pour permettre une évaluation de performances de ce concept de colportage.

A chaque contact, ces nœuds peuvent créer de la connaissance, la stocker et échanger celle accumulée durant leur parcours. Leur tâche est assimilable à celle des *data MULE* [50] où chaque nœud mule parcourt une trajectoire afin de récolter et de transmettre des informations au sein d'un réseau de capteurs sans fil.

### **3.2.2. Les nœuds fixes**

La localisation ou la reconstitution de parcours nécessite des repères géographiques. Ces nœuds fixes appelés également balises fixes sont placés à des endroits stratégiques du champ de captage. Ils ont deux rôles celui d'indiquer une position en se signalant périodiquement et celui de miroir : ils participent aussi au processus de colportage en jouant le rôle de points de dépôt du réseau en stockant et en transmettant les informations aux nœuds qui transitent à proximité.

### **3.2.3. Les collecteurs**

Un nœud collecteur (ou point de collecte) est généralement une station fixe faisant office de puits de données. Le rôle principal de cette entité est de collecter les données colportées par les nœuds mobiles. Ces informations collectées servent ensuite à alimenter des applications de localisation a posteriori dans l'exemple de l'application LoWCA. Les nœuds collecteurs ne sont pas contraints par leur capacité de stockage.

Une première représentation générique d'une galerie minière est donnée dans la figure 41 pour schématiser le rôle de chaque entité du réseau. La mine est décrite comme étant un polygone 2D où plusieurs nœuds balises (B1, B2 et B3) sont placés en des points stratégiques pour les applications de localisation tels qu'à des intersections ou des extrémités de couloirs. Le collecteur quant à lui, est placé juste après l'entrée de la mine qui sert aussi de sortie. Enfin, les nœuds mobiles représentent des mineurs en activité. Ces derniers entrent par la zone de départ avant de passer à proximité du collecteur. Ils se déplacent ensuite dans la mine durant toute leur période d'activité selon la tâche affectée (zone X ou Y) et ressortent de la mine en repassant par le collecteur à la fin de leur période d'activités.

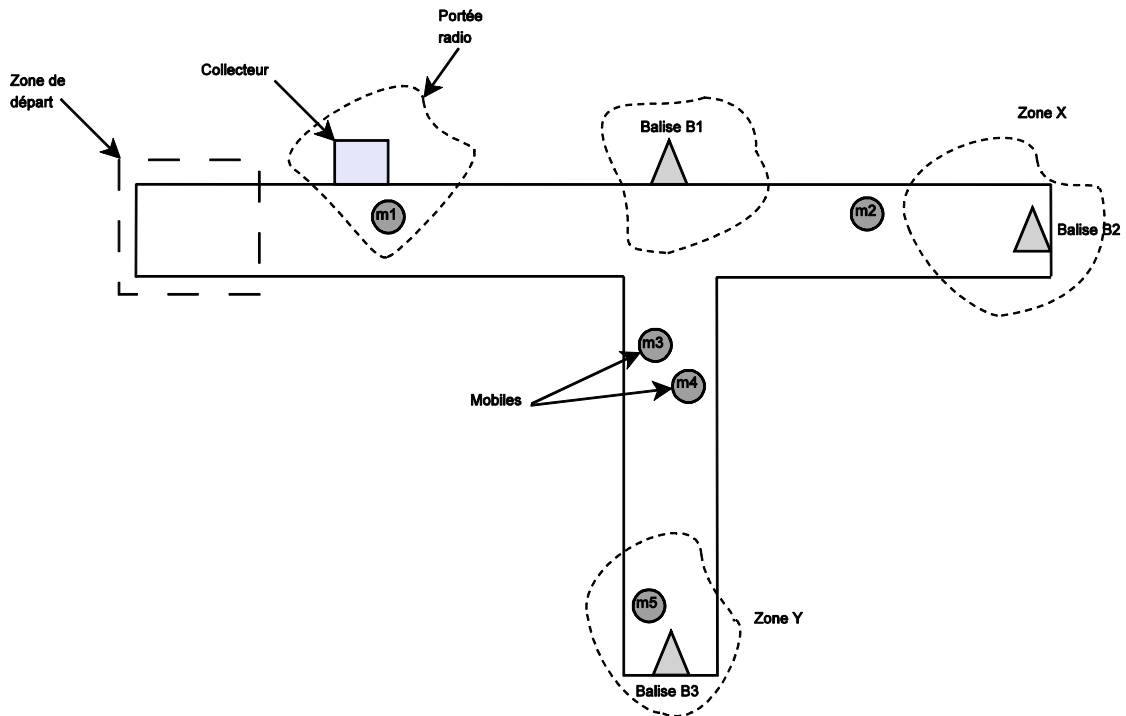


Figure 41 : Représentation générique d'une galerie minière.

### 3.3. Connaissance et information atomique

Le mécanisme de routage proposé dans cette thématique exploite la mobilité des nœuds en permettant à des stations mobiles d'entrer en contact avec d'autres stations fixes ou mobiles afin de collecter, transmettre et de colporter leur connaissance vers un point de collecte. Rappelons que la notion de connaissance citée ici fait référence à une collection d'événements de contact pouvant être manipulés. En effet, l'information atomique (événement de contact) représente l'enregistrement qu'une station est à portée d'une autre. Chaque nœud du réseau quelque soit son type diffuse périodiquement son identité pour signaler sa présence. Ce procédé donne la possibilité à un nœud de détecter la présence d'un ou plusieurs partenaires lorsque ce ou ces derniers sont à portée. On dit que les nœuds sont en contact. Lorsqu'une station détecte la présence d'une autre station à portée, une structure de donnée appelée événement de contact est créée à la suite de cette rencontre. Cet événement représente de ce fait l'information atomique du processus de colportage. Durant la période de contact, les nœuds peuvent échanger leur connaissance afin de contribuer au processus de colportage des informations requises par l'application. Le format générique d'un événement de contact et un exemple d'instanciation (les tailles en octets des différents champs) sont illustrés dans la figure 42.



| My ID    | Numéro de séquence | ID partenaire | Heure arrivée | Heure départ | Champ applicatif |
|----------|--------------------|---------------|---------------|--------------|------------------|
| 2 octets | 1 octet            | 2 octets      | 3 octets      | 3 octets     | 1 octet          |

**Figure 42 : Format générique d'un événement de contact.**

Le choix des tailles en octets des différents champs est basé sur la volonté de minimiser les ressources consommées (capacité mémoire, faible débit, ...etc.). Cette information atomique est néanmoins structurée en trois parties : l'identification du contact, sa durée et un champ applicatif dont l'usage va dépendre de la finalité de l'application. Le couple (IdA, IdB) des identifiants (Id) des deux entités A et B à portée réciproque n'est pas une façon d'identifier un contact car ces deux entités A et B peuvent se croiser plusieurs fois. Il a donc été décidé que l'entité qui crée le contact lui attribue un numéro de séquence, ce qui garantit l'unicité au problème du modulo du numéro de séquence près. La structure d'un événement de contact a pour notre application générique une taille fixe de 12 octets et sa composition détaillée est la suivante :

- My ID (2 octets) : cet identifiant est unique, il est renseigné par l'adresse courte du nœud générateur de l'événement de contact.
- Numéro de séquence (1 octet) : le numéro de séquence est généré par l'entité qui crée l'événement. L'unicité de cet événement est garantie par le couple (MyID, Numéro de séquence).
- ID partenaire (2 octets) : ce champ identifie le partenaire du nœud générateur.
- Heure arrivée (3 octets) : le début de contact est mémorisé dans le champ Heure arrivée.
- Heure départ (3 octets) : la fin d'un contact représente la rupture du lien radio entre les deux entités. Cette information est sauvegardée dans la partie Heure départ de la structure. Les deux informations d'arrivée et de départ sont enregistrées en dixièmes de seconde pour notre application générique.
- Champ applicatif (1 octet) : ce champ est réservé aux besoins de l'application ou pour une utilisation future.

Remarquons qu'un contact génère naturellement deux événements de contact duaux (A, B, N°SeqA) et (B, A N°SeqB).

### 3.3.1. Événement de contact complet

La fin d'un contact est causée par la perte du lien entre un nœud et son partenaire. A cet instant, chacune des deux stations enregistre l'heure à partir de laquelle elle ne reçoit plus aucune trame de ce partenaire. Ensuite, elle insère cette information dans le champ heure départ de l'événement de contact qui initialement, lors d'un début de contact, a une valeur égale à 0. Dans ce cas, on parle d'événement de contact fini ou complet lorsque ce champ est complété en fin de contact. Dans la figure 43, un exemple d'un contact entre deux entités est donné pour montrer la manière dont les événements de contacts finis sont complétés.

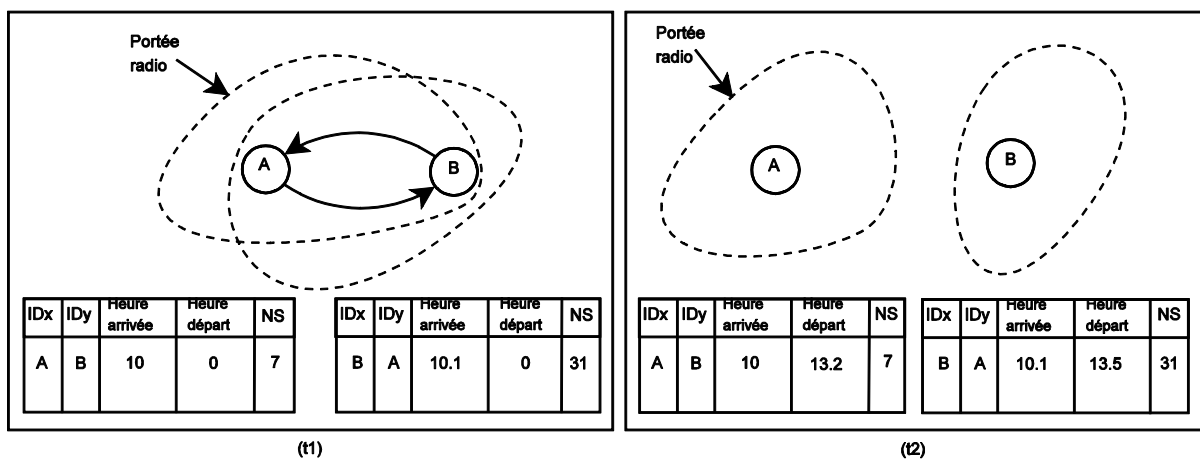


Figure 43 : Exemple de création d'événements de contact complets.

Dans la première partie de la figure 43 (t1), les deux nœuds sont en contact et créent un événement de contact pour enregistrer cette rencontre avec une heure de départ nulle. A cet instant, ces événements sont appelés événements incomplets ou en cours. On note qu'un décalage peut exister concernant les heures d'arrivée (10 pour le mobile A et 10.1 pour le mobile B), cela est dû :

- A la dérive des horloges de nœuds impliqués dans le contact.
- Au fait que la détection de la trame de signalement du partenaire A par le partenaire B et réciproquement ne sont pas des événements synchrones.

Lorsque le contact est rompu (figure 43, t2), les stations complètent alors leur événement de contact avec l'heure locale de la dernière trame reçue (13.2 pour A et 13.5 pour B).

### 3.3.2. Événement de contact incomplet

L'utilité des événements de contact incomplets appelés également événements de contact en cours dans un processus de colportage est d'indiquer, lors d'un échange quelconque entre deux ou plusieurs entités, qu'un contact est en cours et n'est pas encore fini. Cette information peut avoir un impact considérable pour des applications de localisation a posteriori. Pour mieux comprendre cet enjeu, prenons l'exemple de la figure 44 pour lequel un nœud passe à proximité d'un contact entre deux autres stations. Lors de son passage, le mobile B récupère une partie de la connaissance du mobile A et de la balise C1, de plus, il récupère aussi l'information que ces deux entités sont en contact. Si on suppose maintenant que le mobile A reste bloqué dans une mine et que seul le mobile B a permis que cet événement de contact (C1 avec A) soit rapporté au collecteur, cette donnée sera indispensable pour localiser le mobile A a posteriori.

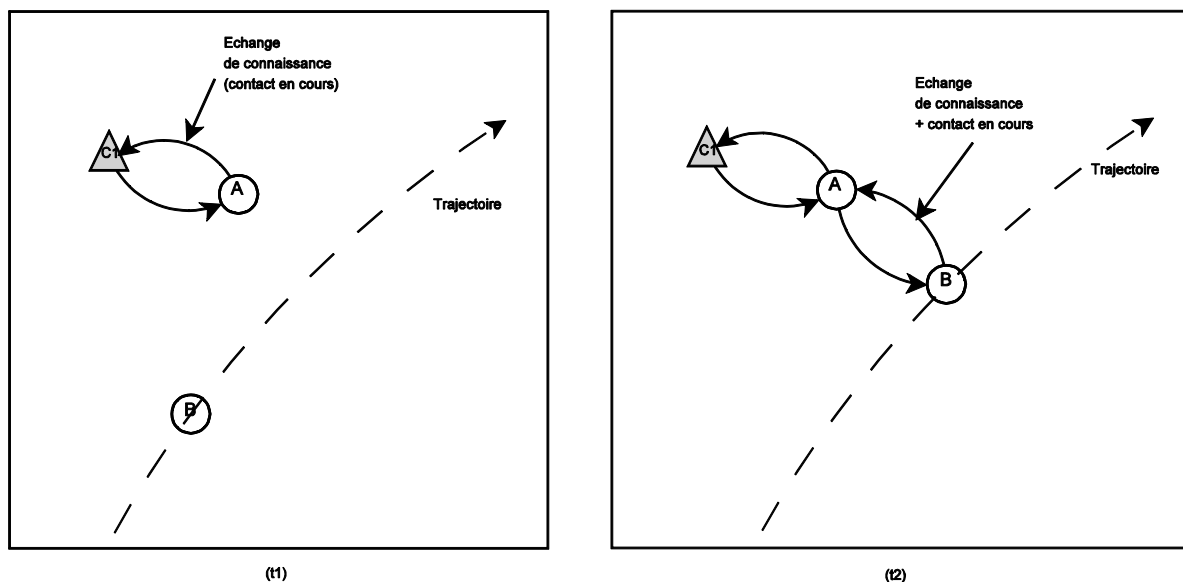


Figure 44 : Échange de contact incomplet.

## 3.4. Stratégies de diffusion

### 3.4.1. Mécanisme d'écho

Tel que mentionné précédemment, le modèle de réseau proposé dans cette thématique est fondé sur un échange de proche en proche de connaissance accumulée lors de rencontres entre des stations fixes ou mobiles. La propagation de ces événements lors d'un contact engendre un nombre important d'échanges lorsque la densité du réseau est importante. On parle ici de processus d'inondation, d'échange de type *1 vers n* (1 émetteur vers n récepteurs) sans aucun envoi d'acquittement pour valider la bonne réception des messages. Il est donc nécessaire de limiter les échanges inutiles afin d'éviter une surcharge du médium et causer un nombre important de collisions et de pertes de paquets

dans le réseau. Le risque que les messages transmis sur le médium soient perdus augmente considérablement étant donné l'asymétrie des communications et la qualité du lien. En effet, une perte importante d'événements de contact entraînera forcément une grande consommation énergétique, une diminution considérable des performances de routage voire une réduction de l'efficacité des applications visées.

Le choix fait ici de diffuser de l'information à son voisinage sans utiliser d'acquittement demande à être vigilant au niveau des risques de pertes de trames. La méthode d'accès employée (CSMA/CA non slotée) présente la particularité de bien résister à la charge car le surplus de trames proposé à la couche MAC est supprimé (« dropé »). Les effets de la surcharge du médium en un même lieu seront étudiés dans le chapitre 4 de ce mémoire. Pour éviter cette surcharge, nous avons choisi de diffuser ce qui doit l'être une seule fois par contact. L'expérimentation et la simulation qui seront développées dans le prochain chapitre ont mis en évidence une conséquence néfaste de l'asymétrie des liens que nous pouvons appeler : Le problème du Rendez-vous manqué.

#### **3.4.1.1. Le problème du « Rendez-vous manqué »**

Supposons qu'un nœud  $m_1$  par exemple se rapproche d'un second nœud  $m_2$  et détecte son signalement. Si  $m_1$  décide de transmettre sa connaissance à  $m_2$ , en raison de l'asymétrie des liens, il se peut que  $m_2$  ne reçoive pas (ou alors qu'une partie) les informations de  $m_1$ . Entendre un partenaire, ne signifie pas être entendu pas lui.

Pour que  $m_1$  soit certain que le partenaire  $m_2$ , qu'il entend, l'entende aussi, un échange entre  $m_2$  et  $m_1$  est nécessaire. Il s'agit de faire un écho à la diffusion pour indiquer que le lien est opérationnel dans les deux sens. Ce phénomène est illustré par la figure 45 dans le cas où deux entités se retrouvent en limite de portée l'une de l'autre et tentent de transmettre des messages. Lorsque le mobile  $m_1$  détecte la présence du mobile  $m_2$  (figure 45, partie t1) il se peut que le mobile  $m_2$  ne soit pas encore touché par le signalement de  $m_1$ . La diffusion de la connaissance qu'entreprendra  $m_1$  (figure 45, partie t2) sera improductive aussi longtemps que le lien ne sera pas efficace dans les deux sens.

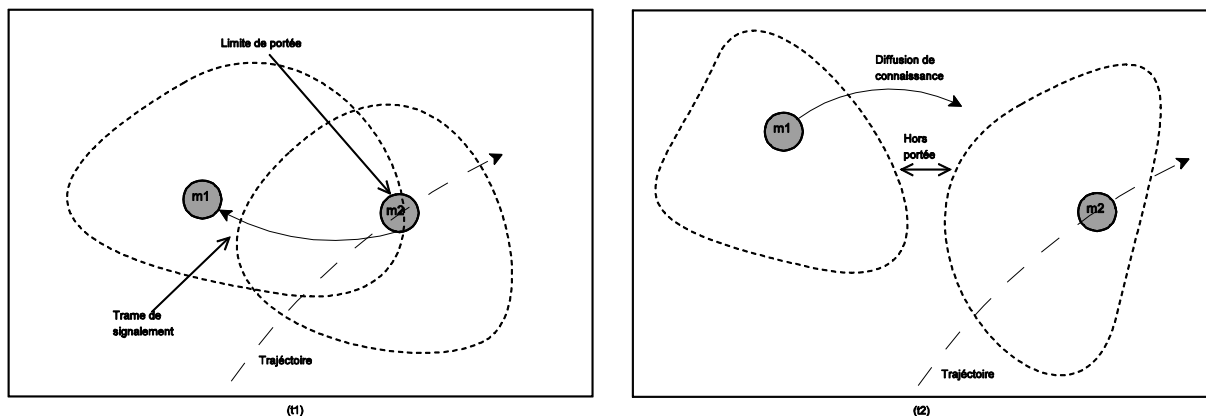


Figure 45 : Problème du rendez-vous manqué.

Afin de remédier à ces problèmes, nous proposons un mécanisme de régulation de flux appelé *écho*. Lors d'un contact entre deux ou plusieurs stations, chaque nœud intervenant dans ce contact, peut éventuellement autoriser les autres partenaires à diffuser ou non leur connaissance en insérant l'identité des nœuds autorisés à transmettre dans sa trame. Un exemple démonstratif est donné dans la figure 46 où le nœud collecteur C1 autorise seulement la station m1 à transmettre réduisant ainsi le nombre de transmissions simultanées.

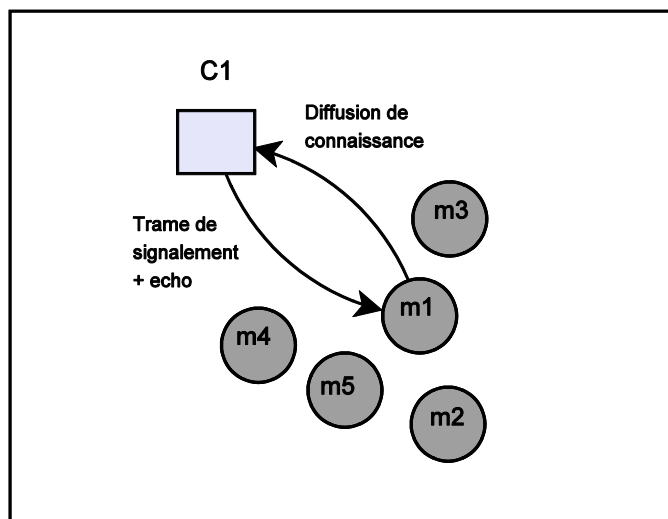


Figure 46 : Fonctionnement du système d'écho.

Nous retenons les principes suivants :

Un nœud diffuse une seule fois sa connaissance à un nouveau partenaire. Cette diffusion peut commencer seulement si le lien est opérationnel dans les deux sens. Un nœud entendant plusieurs partenaires peut décider à quel moment il les autorise à diffuser sur le médium (ceci étant cette autorisation peut être donnée par d'autres entités de ce rassemblement de nœuds). Pour ce faire un nœud indique les nœuds qu'il autorise à émettre. Pour tenir compte du fait que cette autorisation est donnée alors que les entités concernées sont le plus souvent en limite de portée, cette autorisation est confirmée N fois.

### 3.4.1.2. Piste d'implémentation

Nous introduisons deux structures de données indispensables au fonctionnement du mécanisme de l'écho. Il s'agit de deux tables utilisées par chaque nœud lors d'un contact pour gérer son voisinage. La première table appelée *table des nouveaux entrants* concerne les nouveaux arrivants dans la cellule du nœud avant que le contact ne soit confirmé et la seconde table appelée *table des voisins* est dédiée à la gestion des partenaires lorsque le contact est confirmé.

#### **Tables des nouveaux entrants**

La première structure de données utilisée par le mécanisme d'écho est la table des nouveaux entrants qui est consacrée principalement à la gestion des stations avant la confirmation d'un établissement de contact et dont la composition est représentée dans la figure 47.

| Id entrant | Nb écho |
|------------|---------|
| 1          | 3       |

**Figure 47 : Structure d'une table des nouveaux entrants.**

Les deux champs qui constituent cette table sont donc l'identité du nouveau partenaire et le nombre de fois que l'écho lui sera signifié. Ce champ représente le nombre de fois qu'une station avertit son partenaire qu'elle est à proximité et qu'elle est prête à recevoir de la connaissance de sa part.

L'exploitation de cette table est la suivante :

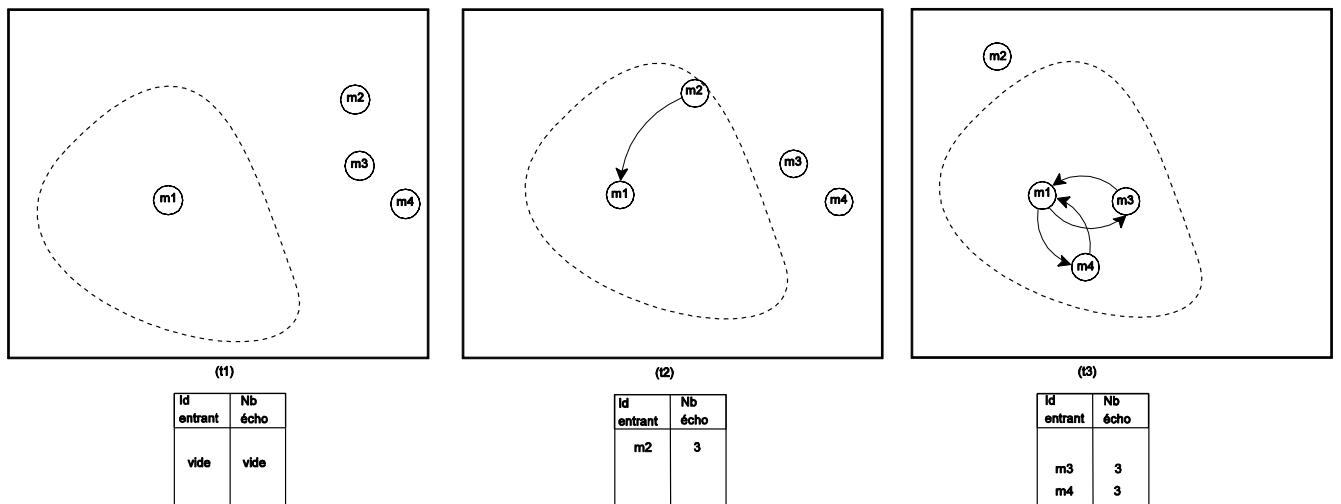
- A l'initialisation, la table des nouveaux entrants est vide.
- A la détection d'un nœud à proximité, ce nœud est déclaré automatiquement dans la table des nouveaux entrants.
- La taille maximale de cette table est déterminée par la constante *maxSizeNewsNeighbours* (par défaut égale à 8 enregistrements). Cette valeur signifie que chaque nœud ne peut gérer et autoriser en même temps que 8 stations à portée. Dans les cas où un nœud est à proximité de plus de 8 stations, les autres devront attendre leur tour pour communiquer.
- Ce partenaire est aussi déclaré dans la table des voisins.
- Ce partenaire reste dans cette table jusqu'à ce que la liaison soit établie dans les deux sens.

Supposons maintenant qu'un nœud  $m_1$  se rapproche d'un autre nœud  $m_2$  et détecte son signalement. Si  $m_1$  veut confirmer qu'il voit  $m_2$ , c'est-à-dire qu'il veut faire savoir que la diffusion peut commencer, il faut pour cela que le lien soit devenu symétrique. Tant que cela ne sera pas le cas cette confirmation sera perdue. Nous avons décidé de transmettre cette confirmation à  $m_2$  plusieurs fois en espérant qu'entretemps le lien sera devenu symétrique et stable.

- Ce nouvel entrant doit être averti qu'il vient d'être détecté par un partenaire pour qu'il puisse diffuser sa connaissance. Pour cela, nous disposons d'un nombre de tentatives maximal fixé par une variable *nbMaxEcho* pour l'avertir. Avertir un partenaire qu'on le voit revient à remplir un champ avec l'identité du nœud concerné. Lorsque ce dernier lit une trame entrante et trouve son identité, il sait que le partenaire qui l'a transmise est en train de le voir.
- Un élément de ce tableau est supprimé dans les trois cas suivants :
  - Si aucune trame n'est reçue de ce partenaire pendant une période d'observation.
  - Si le partenaire commence ou est déjà en train de diffuser sa connaissance.
  - Si le nombre d'écho utilisé pour l'avertir est dépassé ( $nbMaxEcho = 0$ ).

Un exemple de remplissage de cette table est donné dans la figure 48, où suite à la détection de 3 nouveaux partenaires ( $m_2, m_3, m_4$ ), le nœud  $m_1$  remplit sa table comme suit :

- Etape 1 : la table est initialisée à vide lorsqu'aucune entité n'est à portée.
- Etape 2 : l'identifiant du mobile  $m_2$  est inséré dans la table de  $m_1$  pour faire suite à la réception d'une première trame de  $m_2$  vers  $m_1$ . Le champ écho est aussi initialisé à sa valeur de départ, c'est-à-dire 3. Le mobile  $m_1$  tente d'avertir son voisin en insérant l'identité de  $m_2$  dans sa trame. La même opération est effectuée par le mobile  $m_2$  s'il reçoit une trame de  $m_1$ .
- Etape 3 : lorsque le mobile  $m_2$  sort de la zone de détection de  $m_1$ , ce dernier détecte qu'il ne reçoit aucune trame de ce partenaire pendant une période d'observation. En conséquence, aucune information n'est transmise. Par contre, quand les mobiles  $m_3$  et  $m_4$  entrent à leur tour en contact avec le mobile  $m_1$ , en spécifiant respectivement dans leurs trames qu'ils ont détecté sa présence, le nœud  $m_1$  déclenche automatiquement une procédure de diffusion de sa connaissance.



**Figure 48 : Scénario d’usage du système d’écho.**

On voit à travers cet exemple l’utilité du système de l’écho pour lequel chaque nœud doit attendre une confirmation d’un de ses nouveaux partenaires avant d’entamer une diffusion de sa connaissance. De plus, lorsqu’une station détecte la présence de plusieurs autres stations à proximité, elle peut décider du nombre de stations autorisées à transmettre. Ceci dans le but de réduire considérablement le nombre de transmissions simultanées et en conséquence le nombre de collisions. Dans le point suivant, nous allons expliquer le rôle de la table des voisins dans le système des échos en nous appuyant sur l’exemple de la figure 48 pour montrer le fonctionnement de cette structure.

### **Tables des voisins**

Lorsqu’une ou plusieurs stations entrent en contact avec un nœud, chacun de ses partenaires met à jour à son niveau la table des nouveaux entrants et la table des voisins. Cette structure de données est utilisée en premier lieu pour la gestion des partenaires lorsque le contact est confirmé par le mécanisme d’écho de part et d’autre. En second lieu, elle fournit des informations pour renseigner la structure d’un événement de contact. Les différents champs qui composent cette structure sont illustrés dans la figure 49.

| Id_Part | Num_Seq | Heure_arrivée | Heure_dernière_trame |
|---------|---------|---------------|----------------------|
| 1       | 2       | 22.0          | 27.2                 |

**Figure 49 : Structure d’une table des voisins.**

Considérons un nœud m1 et la table de ses voisins, les définitions des champs de cette structure sont les suivantes :

- Id\_Part : ce champ représente l’identifiant du nœud partenaire de m1 impliqué dans le contact.



- Num\_Seq : un numéro de séquence unique est attribué par m1 à chaque nouveau voisin. Ce numéro est incrémenté à chaque nouvelle entrée dans la table de m1.
- Heure\_arrivée : l'heure d'arrivée indique l'heure locale de m1 à la réception de la première trame de ce voisin. Cette information est ensuite réutilisée dans la structure d'un événement de contact pour indiquer le début de ce contact.
- Heure\_dernière\_trame : cette heure locale est mise à jour à chaque nouvelle réception d'une trame de ce partenaire. Si la différence entre l'heure de la dernière trame et l'heure courante dépasse un certain seuil, on dit qu'il y a rupture de contact et le partenaire est retiré de la table des voisins. L'heure de la dernière trame reçue devient l'heure de la fin de contact et sera ensuite insérée dans la structure de l'événement de contact.

Le fonctionnement de cette table est expliqué dans les points suivants :

- Initialement, la table des voisins est vide.
- Un nouvel entrant est inséré en même temps dans la table des nouveaux entrants et celle des voisins.
- La table des voisins est limitée en taille par le paramètre *maxSizeNeighbours* dont la valeur est fixée à 20 dans notre étude. Cette limite aura un impact important sur le choix de la connaissance à stocker en mémoire.
- Un élément de ce tableau disparaît quand on ne l'entend plus. Dans ce cas, un événement de contact est généré et sera stocké dans la table de connaissance locale.

Nous reprenons dans la figure 50 l'exemple du système d'écho de la figure 47 en ajoutant la table des voisins pour montrer comment cette table se remplit dans le temps.

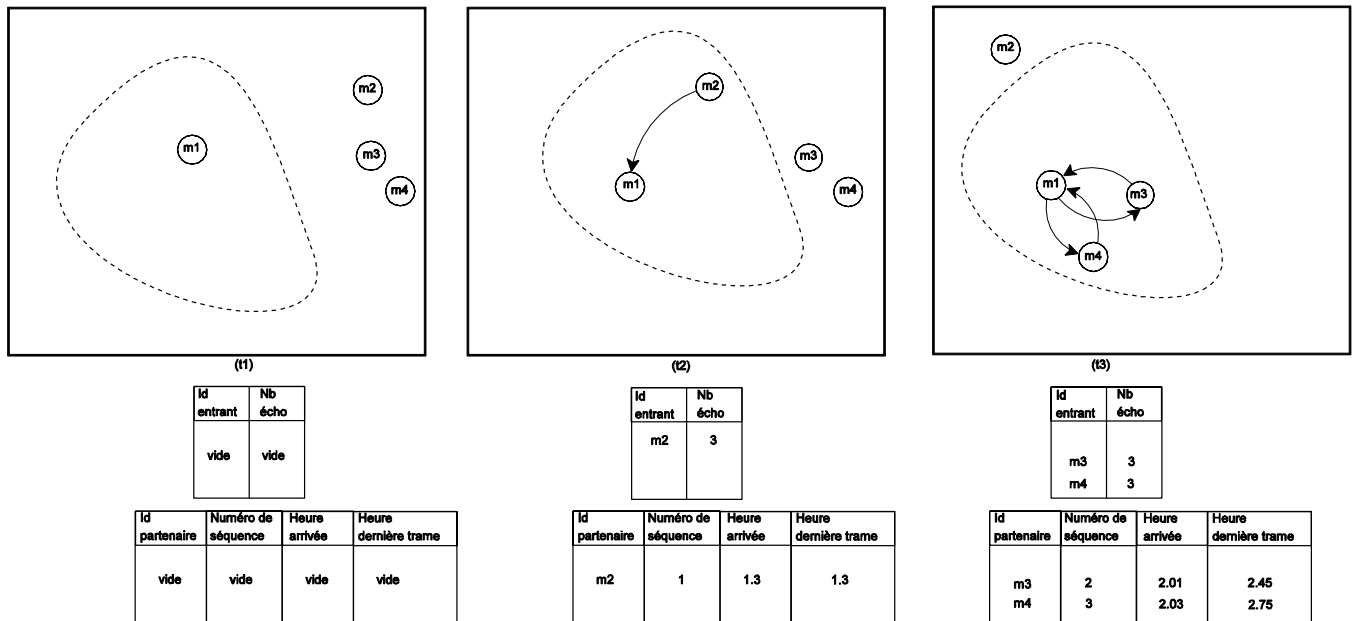


Figure 50 : Scénario d’usage de la table des voisins.

Dans cet exemple, le mobile m1 remplit sa table comme suit :

Nous supposons dans ce scénario que cette entité m1 n’a pas de voisin au départ.

- Etape 1 : la table est initialisée à vide lorsqu’aucune entité n’est à portée.
- Etape 2 : suite à l’insertion de m2 dans la table des nouveaux entrants, il est automatiquement inséré dans la table des voisins avec un numéro de séquence égale à 1 et l’heure courante locale est utilisée pour remplir simultanément les champs heure d’arrivée et de dernière trame reçue.
- Etape 3 : lorsque le mobile m2 n’est plus à portée de m1, le champ dernière trame reçue n’est plus modifié et le voisin est supprimé après un seuil temporel, un événement de contact est créé suite à cette rupture de contact. Les mobiles m3 et m4 sont ensuite insérés dans la table en mettant à jour l’heure des dernières trames reçues.

### Table de connaissance

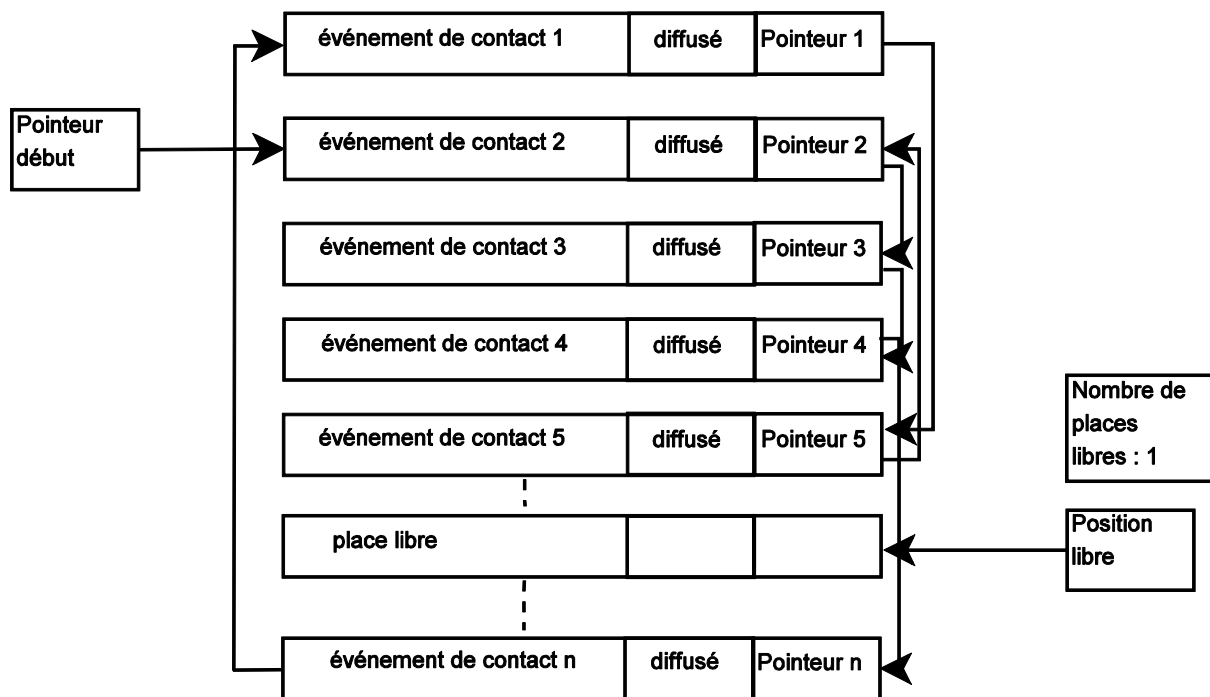
Chaque nœud du réseau a pour mission de colporter sa connaissance accumulée au fil du temps vers un ou plusieurs points de collecte pour alimenter par exemple des applications de localisation ou de traçabilité. Dans le mécanisme colportage, chaque station est dotée d’une structure de donnée pour stocker toutes ces informations. Il s’agit d’une table limitée en espace mémoire (une centaine de contact) où chaque ligne représente un événement de contact. Lorsqu’un contact entre deux entités est

déecté après l'expiration d'un délai d'observation (seuil temporel), cet événement est automatiquement inséré dans la table de connaissance. Toutes les informations qui concernent cet événement sont issues de la table de voisins (identifiants, numéro de séquence...etc.).

La table de connaissance est gérée sous forme de liste chaînée, c'est-à-dire que chaque élément pointe vers l'élément suivant pour faciliter le parcours de cette table et pour avoir un séquençement dans le temps des événements de contact.

Nous montrerons dans le prochain chapitre que la qualité de la détection d'un contact est tributaire de la quantité de trafic échangé dans la zone de contact. Pour optimiser le volume de l'information diffusée nous avons prévu de diffuser la connaissance qui doit l'être, dans son intégralité au dernier entrant. Si cette diffusion est déjà en cours lors de l'arrivée du dernier entrant par un jeu de pointeurs et de marqueurs (diffusés) nous allons nous assurer que cette diffusion sera complète sans qu'elle soit redondante pour lui (le dernier entrant).

Le champ *diffusé* indique si l'événement doit être diffusé ou non dans le contact courant. L'offset *pointeur début* est utilisé pour marquer le premier élément de cette structure. Enfin, le nombre de places libres et la position de la première sont gérés respectivement par les variables *nombre de places libres* et *position libre*. La figure 51 donne une représentation de cette table.



**Figure 51 : Structure d'une table de connaissance.**

Lorsqu'un contact a lieu entre deux entités communicantes, le nœud émetteur commence à garnir ces trames à partir du contenu de la table de connaissance en commençant par l'événement pointé par la

variable *pointeur début*. Ce même nœud est supposé s'arrêter de transmettre lorsque le pointeur aura fait tout le tour de la table. On pose l'hypothèse ici qu'une station ne communique qu'une seule fois sa connaissance lors d'un contact afin d'éviter de retransmettre la même chose à son partenaire et de saturer le trafic local. Evidemment, le format des trames proposé par la norme 802.15.4 [34] nous limite à un *Payload* (données utiles) de 116 octets, ce qui a un impact sur le nombre d'événements de contact à diffuser. Ce nombre est fixé à 8 événements par trame, nous justifierons ce nombre lorsque nous présenterons le format des trames diffusées dans le mécanisme de colportage. Un exemple d'échange de connaissance est donné plus loin dans la figure 53 juste après la présentation du concept d'état.

### 3.4.2. Notion d'état

Le comportement des nœuds dans le processus de colportage peut être décrit par un automate d'états fini. A chaque état correspond un comportement particulier du nœud, qu'il s'agisse de transmettre des trames de diffusion de connaissance ou des trames de signalement. Une trame de signalement est une trame ne contenant que l'identifiant du nœud émetteur. Le but escompté par cette notion d'état est de diminuer le nombre de transmissions lorsqu'une station n'a aucune entité à proximité ou que tous les événements stockés en mémoire devant être diffusés, l'ont été au moins une fois lors d'un contact. La figure 52 regroupe tous les états et transitions possibles de cette stratégie.

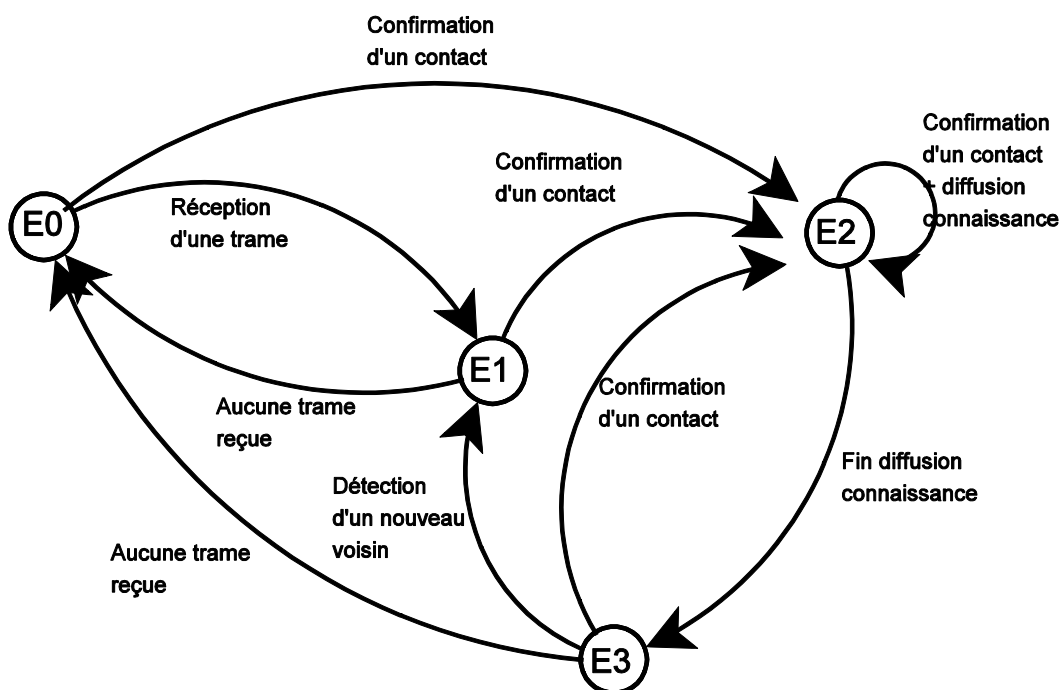


Figure 52 : Automate d'états pour la gestion des nœuds.

Considérons l'entité *m1*. Les spécifications de chaque état de *m1* sont données en détail dans les points suivants :

- Etat 0 [Isolé]: correspond à l'état initial où le nœud m1 ne fait que de se signaler, pour cela il émet périodiquement des trames de signalement. La station change d'état dans les cas suivants :
  - Si elle reçoit une trame ne contenant pas son identifiant, elle passe automatiquement à l'état E1.
  - Si la trame reçue confirme qu'un partenaire de m1 est en train de le voir : l'identité de m1 est indiquée dans la trame reçue. L'entité m1 passe à l'état E2.
  
- Etat 1 [Voisin entrant]: cet état confirme qu'un voisin est à portée mais que le contact n'est pas encore confirmé par les deux partenaires. Chaque nœud qui se trouve dans cet état suite à la réception d'une trame doit enclencher le mécanisme d'écho pour avertir son voisin. Si aucune réponse n'est reçue après cela, le nœud repasse à l'état initial E0. Sinon, dans le cas où le partenaire confirme le contact, les deux stations passent à l'état E2 pour entreprendre l'échange d'information.
  
- Etat 2 [Contact établi]: cet état confirme qu'un nœud est bien en contact avec une ou plusieurs stations. Si la base de connaissance contient des événements de contact, le nœud diffuse une partie ou la totalité de sa connaissance. Dans le cas où une station m1 n'a aucune donnée à transmettre, ou bien si m1 a fini de diffuser sa connaissance au dernier entrant impliqué dans le contact, cette entité passe à l'état E3.
  
- Etat 3 [Echange terminé] : cet état représente une situation de stabilité qui indique qu'un contact a lieu, mais que toutes les stations impliquées dans ce contact ont fini d'échanger leurs données et n'ont plus rien à communiquer. Les changements d'état s'effectuent comme suit :
  - Si un nouveau voisin a été détecté, mais qu'il n'a pas confirmé, le nœud passe à l'état E1.
  - Si une nouvelle station confirme le contact, le nœud repasse à l'état E2.
  - Si aucune trame n'est reçue après une fenêtre d'observation le contact est considéré comme rompu, l'état repasse à E0.

Les changements d'état définis au-dessus sont appliqués à tous types de nœuds (mobiles, balises et collecteur). Néanmoins, une exception est à signaler dans le cas du collecteur, qui ne diffuse pas sa connaissance lorsqu'il atteint l'état E2. En effet, durant tout le processus de colportage, le collecteur ne fait que se signaler et reste en attente des données colportées par les mobiles qui passent à proximité.

Un exemple est donné dans la figure 53 montrant les changements d'état pour un scénario impliquant deux nœuds mobiles. Initialement, les deux mobiles sont hors portée et leurs états est « isolé » i.e. à E0 (t1). Lorsque le mobile m1 détecte la présence de m2 en recevant sa trame de signalisation (t2), l'état de m1 passe à « Voisin entrant » i.e. à E1 et le mobile répond automatiquement avec une trame de données comportant l'identifiant de m2 pour l'avertir qu'il a reçu sa trame. Lorsque la station m2 reçoit cette trame, elle passe directement à l'état « Contact établi » i.e. à E2 puisque son identifiant est dans cette trame. M2 met à jour sa connaissance en récupérant les données reçues de m1. Ensuite, le mobile m2 effectue la même opération en transmettant à son tour l'identifiant de son nouveau partenaire ainsi que sa connaissance. M1 passe dans ce cas à l'état E2 et met à jour la connaissance reçue (t3). Dans la partie (t4), les deux mobiles passent à l'état « Echange terminé » i.e. E3 puisqu'ils n'ont plus rien à se dire et ne font qu'émettre des trames de signalement. Enfin, lorsque les mobiles se quittent dans (t5), les états passent à E0 mais les stations se signalent encore pour détecter d'éventuels nouveaux partenaires.

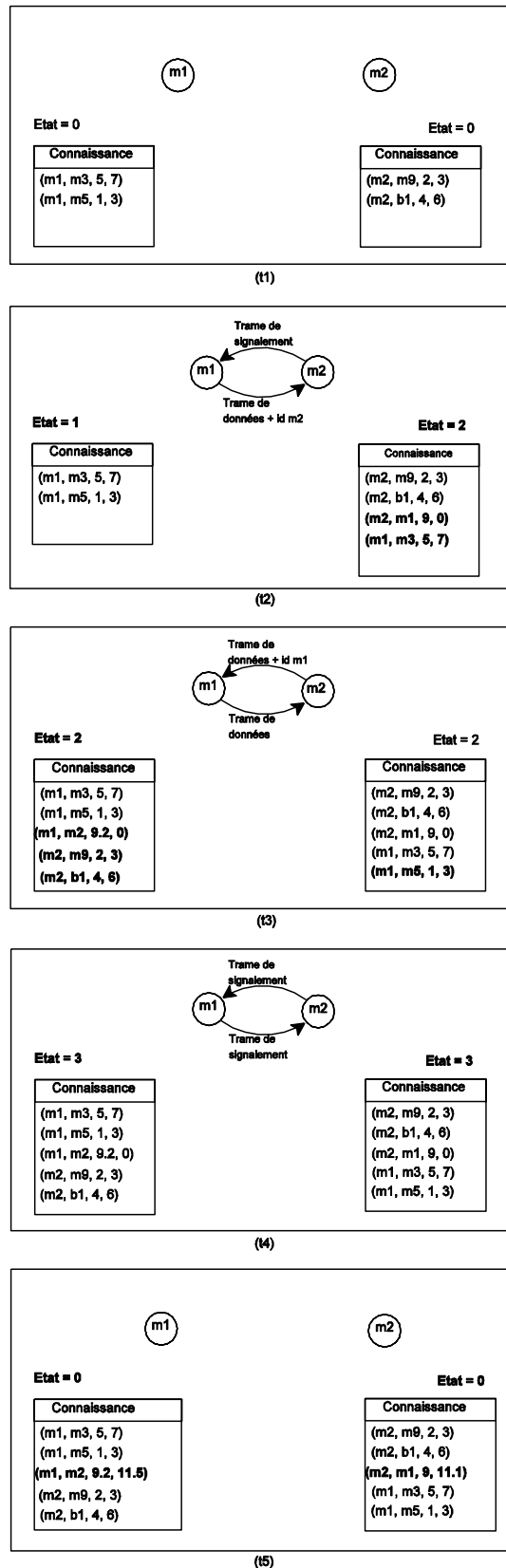


Figure 53 : Exemple d'évolution de l'état des mobiles et de leur connaissance.

### 3.4.3. Format d'une trame de diffusion

Après avoir défini le format général d'un événement de contact, les principales structures de données et le mécanisme de l'écho, nous introduisons à présent le format générique d'une trame de diffusion de connaissance. Le rôle de cette structure est de permettre à la fois de s'identifier, d'appliquer le mécanisme d'écho et de transmettre de la connaissance. Cette trame est donc composée de trois grandes parties. La première est consacrée à l'identifiant du nœud, la seconde gère le système écho et la troisième est réservée à l'envoi des événements de contact. La structure de cette trame est représentée dans la figure 54.

| Heure d'émission | Contenu champ écho : N1 | Contenu champ connaissance : N2 | Champ écho      | Champ connaissance |
|------------------|-------------------------|---------------------------------|-----------------|--------------------|
| 3 octets         | 1 quartet               | 1 quartet                       | $N1 * 2$ octets | $N2 * 12$ octets   |

**Figure 54 : format d'une trame de diffusion.**

La description détaillée des différents champs de la trame de diffusion est donnée dans les points suivants :

- Heure émission (3 octets) : le champ heure émission est indispensable pour le processus de synchronisation et la correction des horloges. Dans la partie synchronisation, nous décrirons plus en détail l'utilité de ce champ.
- Contenu champ écho N1 (1 quartet) : ce champ indique le nombre de nouveaux entrants impliqués dans le contact courant et autorisés à diffuser.
- Contenu champ connaissance N2 (1 quartet) : il indique le nombre d'événements de contact insérés dans le champ connaissance de la trame.
- Champ écho ( $N1 * 2$  octets) : il contient les identifiants des nouveaux entrants autorisés à émettre. Ce champ est garni à partir de la table des nouveaux entrants.
- Champ connaissance ( $N2 * 12$  octets) : à chaque confirmation d'un contact, les nœuds diffusent tout ou une partie de leur connaissance en remplissant ce champ à partir de la table de connaissance. Néanmoins, le format de trame proposée par la norme 802.15.4 [34] fixe la taille du *Payload* autorisé à 116 octets, huit événements de contact peuvent être donc insérés dans une trame au maximum.



Lorsqu'un nœud transmet une trame vide pour signaler seulement sa présence, les deux champs « contenu » sont renseignés par un 0 et les champs écho et connaissance ne sont pas renseignés, ce qui donne un *Payload* de 4 octets. Par contre, dans le cas où tous les champs sont renseignés et utilisés au maximum de la capacité de cette structure, le *Payload* total est de 116 octets.

### 3.4.4. Algorithme de transmission de trames (diagramme de transmission)

Après avoir défini le format d'une trame de diffusion, qu'elle soit courte quand elle ne sert qu'au signalement ou bien longue pour une trame de diffusion de connaissance. Nous allons à présent expliquer le fonctionnement de la diffusion de ces trames dans le processus de colportage. L'algorithme de transmission est schématisé dans la figure 55.

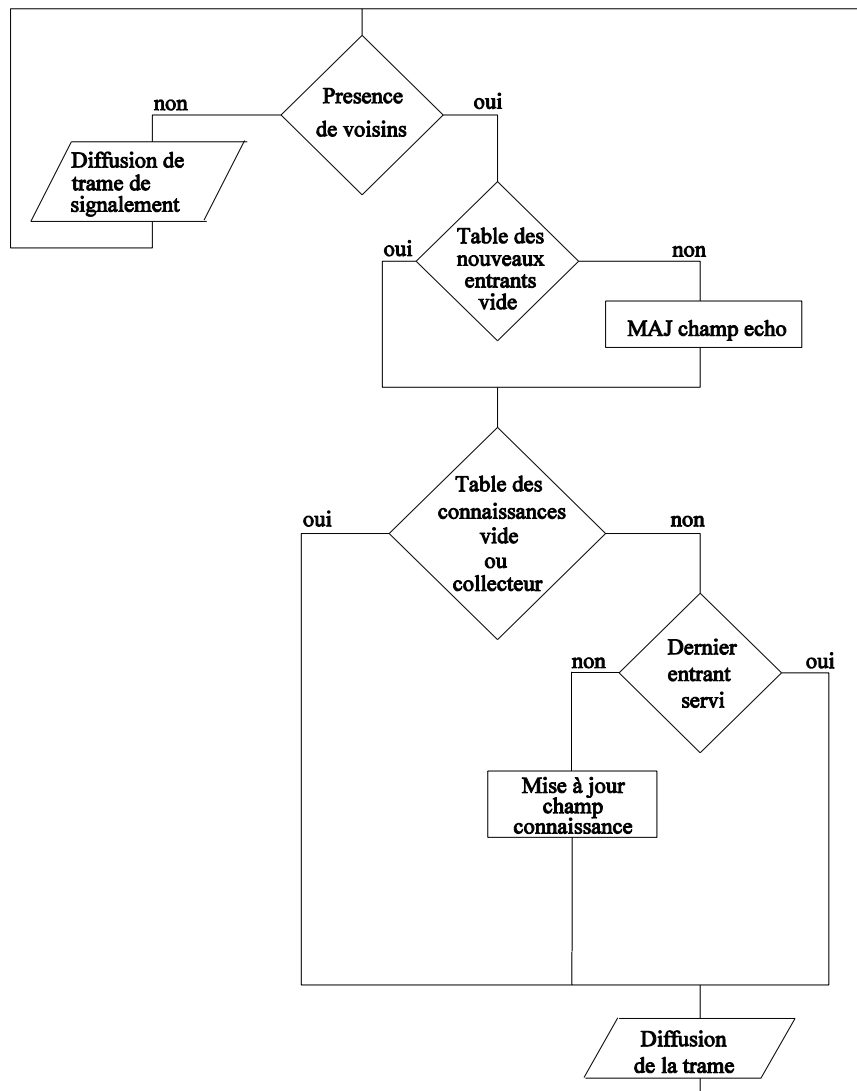


Figure 55 : Algorithme de transmission de trame

Les différentes étapes de l'algorithme de transmission sont expliquées dans les points suivants :

- Chaque nœud dans le réseau, lorsqu'il est isolé (n'est à portée d'aucune autre station), se signale périodiquement en diffusant des trames courtes pour signaler sa présence.
- Lorsqu'un nouveau partenaire est à proximité du nœud, ce dernier doit vérifier l'existence de ce partenaire dans sa table des nouveaux entrants :
  1. Si le partenaire est un nouveau voisin, son identifiant doit être inséré dans la partie écho de la trame pour l'informer qu'une station est à portée. Il est aussi possible de différer cette réponse pour asservir localement le flux.
  2. Si l'identifiant du voisin existe déjà dans la table des nouveaux entrants, l'algorithme passe à l'étape suivante.
- Lorsque le contact est établi entre deux entités du réseau (à l'aide du mécanisme d'écho), l'algorithme de transmission passe à la seconde étape qui consiste à diffuser la connaissance locale d'un nœud :
  1. Si la table de connaissance d'une station est vide ou si le nœud émetteur est le collecteur du réseau, l'algorithme se termine en ne diffusant que l'identité de la station avec éventuellement la partie écho pour autoriser les voisins à diffuser leur connaissance.
  2. Sinon, si la table de connaissance contient des événements de contact, l'algorithme doit vérifier si le dernier entrant a été servi ou non à l'aide de la variable booléenne *dernier entrant*. En effet, un nœud est supposé diffuser la totalité de sa connaissance qu'une seule fois lors d'un contact entre une ou plusieurs stations fixes ou mobiles :
    - a) Si le dernier entrant est servi, le nœud arrête d'alimenter sa trame avec des événements de contact en attendant un nouvel entrant.
    - b) Si le dernier entrant n'a pas été complètement servi, le nœud émetteur de la trame continue à remplir sa trame

avec la connaissance qui ne lui a pas été encore diffusée.

- Diffusion de la trame : la dernière étape de l'algorithme consiste à diffuser périodiquement cette trame qu'elle soit courte ou longue.

### 3.4.5. Algorithme de réception de trames (diagramme de réception)

L'algorithme de réception des trames du processus de colportage doit assurer deux tâches principales qui consistent en premier lieu à gérer la réception des nouveaux événements de contact lorsque la trame reçue n'est pas qu'une simple trame de signalement. Le second rôle de cet algorithme consiste à administrer le voisinage d'un nœud afin d'autoriser ou non un voisin à émettre en utilisant les tables des nouveaux entrants et des voisins. La figure 56 illustre les étapes de cet algorithme.

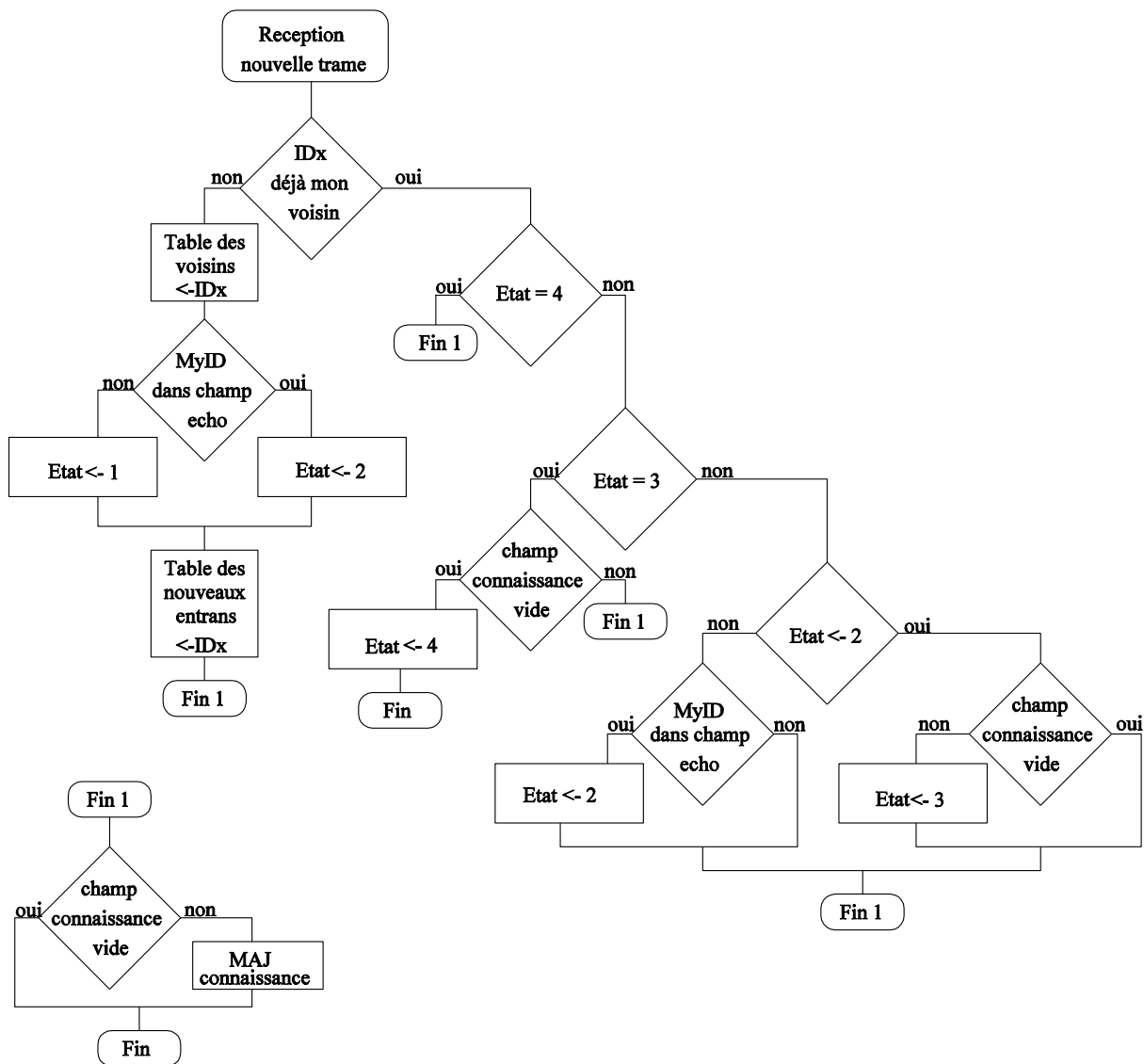


Figure 56 : Algorithme de réception de trame.

L'algorithme exécuté par un nœud qui reçoit une trame, c'est-à-dire un récepteur, est composé de deux parties : l'une (jusqu'à Fin1) consiste à gérer les échanges entre récepteur et l'émetteur de la trame, l'autre (de Fin1 à Fin) consiste à prendre en compte la connaissance éventuellement portée par la trame reçue pour mettre à jour la connaissance du récepteur.

Le fonctionnement de l'algorithme est le suivant :

- Un état est attribué à chaque voisin selon sa situation :
  - Etat = 1 : le contact entre les deux partenaires n'est pas confirmé.
  - Etat = 2 : le contact est confirmé en recevant une trame contenant l'identité du nœud récepteur.
  - Etat = 3 : la trame contenant l'identité du récepteur contient également de la connaissance.
  - Etat = 4 : le contact est en cours mais le nœud voisin ne transmet aucune connaissance (trame de signalement seulement).
- Réception d'une nouvelle trame : l'algorithme est exécuté par chaque récepteur à la réception d'une trame quelque soit son type.
- IDx déjà mon voisin ? : le premier test effectué par le récepteur consiste à vérifier si le partenaire qui a émis cette trame fait déjà partie de la liste des voisins :
  1. Si l'identifiant de l'émetteur n'existe pas dans la table des voisins, il sera inséré dans cette table. Ensuite, le nœud doit vérifier si son identifiant est dans le champ écho. Si le récepteur retrouve son identifiant dans la trame, alors son état passe à 2, sinon il passe à l'état 1. L'identifiant du partenaire est ensuite inséré dans la table des nouveaux entrants.
  2. Si l'identifiant du nœud voisin existe déjà dans la table des voisins, alors le récepteur doit vérifier son état actuel :
    - a. Etat = 4 ? Si l'état du nœud est égal à 4 alors l'algorithme passe à la phase « Fin 1 ». Sinon, il teste l'état 3 (la diffusion de la connaissance n'est pas finie).
    - b. Etat = 3 ? Dans le cas où le nœud se trouve dans l'état 3, il doit inspecter le champ connaissance de la trame reçue. Si le champ connaissance est vide, alors l'état revient à l'état 2 et l'algorithme

de réception se termine. Dans le cas où la partie connaissance n'est pas vide, l'algorithme passe automatiquement à la phase « Fin 1 ».

- c. Etat = 2 ? Si l'état du nœud est égal à 2 et le champ connaissance n'est pas vide, l'état passe à 3. Ensuite, l'algorithme de réception se termine par la phase « Fin 1 ».
- d. Etat = 1 ? Si la trame contient des événements de contact, l'état doit passer à l'état 2 sinon, l'état du nœud reste à 1 et l'algorithme se dirige vers la dernière phase « Fin 1 ».

- Phase Fin 1 : l'algorithme de réception se termine par cette dernière phase qui consiste à mettre à jour la connaissance du nœud récepteur dans le cas où le champ connaissance de la trame reçue n'est pas vide.

### **3.5. Mécanisme de routage**

Le système de colportage de connaissance introduit précédemment à travers ces différentes facettes (mécanisme d'écho, notion d'états, algorithmes de transmission et de réception...etc.) induit une forte redondance des données propagées dans le réseau. En effet, chaque nœud fixe ou mobile impliqué dans le processus de colportage crée des événements de contact à chaque rencontre. Ces événements de contact sont ensuite conservés en mémoire et propagés dans le réseau à chaque nouvelle rencontre avec comme objectif final d'acheminer toute cette connaissance vers un point de collecte qui servira à alimenter les applications ciblées. En adoptant cette approche de diffusion de la connaissance, chaque nœud déployé dans le réseau risque de recevoir inutilement plusieurs fois le même événement de contact. Pour pallier ce problème, le mécanisme de routage proposé dans cette thèse est fondé principalement sur le filtrage de la connaissance. Il propose une liste de filtres afin d'apporter un maximum de diversité ou la pertinence dans la connaissance véhiculée par ce réseau afin de contrôler la redondance des événements de contact rapportés au collecteur.

#### **3.5.1. Stratégies de filtrage**

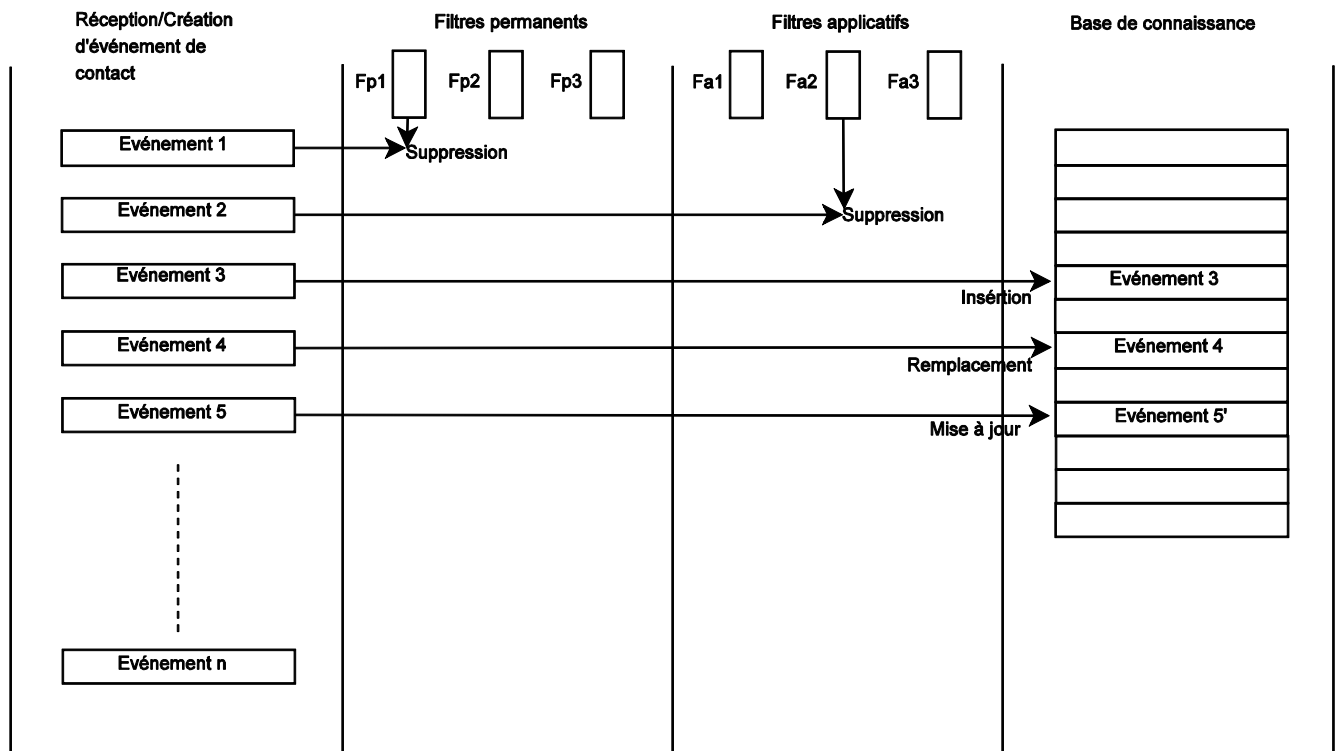
Pour lutter contre la saturation rapide de la faible capacité de stockage d'un nœud, nous proposons une politique de filtrage qui va permettre à un nœud de choisir parmi les informations qu'il reçoit celles qui souhaite conserver. Ceci consiste à supprimer un événement de contact selon des critères

caractérisant le filtre appliqué, ou à condenser l'information stockée : à partir de plusieurs événements de contact n'en faire qu'un seul par exemple.

Le fonctionnement du mécanisme de filtrage est le suivant :

- Chaque nœud applique un ou plusieurs filtres à la réception des événements de contact.
- Aucun filtre n'est appliqué pour la diffusion de la connaissance (nous avons fait ce choix car les événements de contact stockés en mémoire sont déjà filtrés).
- Les filtres sont appliqués de manière séquentielle dans un ordre précis.
- Un filtre peut soit supprimer un événement de contact en mémoire, modifier un champ (identifiant, heure arrivée, heure départ... etc.) ou remplacer un événement dans la base de connaissance.
- La même procédure est appliquée aux événements de contact créés localement.

Dans les points qui vont suivre, nous proposons une liste de tous les filtres impliqués dans le processus de colportage. Ces filtres sont classés en trois catégories. La première catégorie concerne les filtres permanents appliqués de manière systématique à chaque réception d'un événement de contact même si la mémoire du nœud n'est pas saturée. La seconde catégorie de filtres s'occupe de la gestion de la mémoire lors d'une saturation de l'espace de stockage. La troisième catégorie propose une liste de filtres applicatifs qui dépendent des applications finales visées. Un exemple est donné dans la figure 57 pour résumer le fonctionnement du filtrage des événements de contact.



**Figure 57 : Fonctionnement du filtrage des événements de contact [75].**

Lors de la réception ou de la création d'événements de contact, les premiers filtres appliqués systématiquement et dans l'ordre sont les filtres permanents, ils sont indépendants des applications visées. Leur rôle principal est de supprimer les événements représentant un même contact. En effet, plusieurs événements de contact peuvent représenter un même contact et il n'est pas nécessaire dans ce cas de tous les garder en mémoire. Dans la figure 57, l'événement 1 sera supprimé par un des filtres permanents, car il est considéré comme un doublon. Les filtres applicatifs (Fa) sont appliqués à la suite des permanents et dépendent de l'application visée. L'événement 2 de l'exemple n'est pas supprimé par les filtres permanents, mais le filtre applicatif Fa2 considère que cet événement n'est pas nécessaire pour l'application cible (localisation par exemple) et annule son insertion. Le troisième événement remplit toutes les conditions imposées par l'ensemble des filtres permanents et applicatifs, il est donc inséré dans la table de connaissance. Il en est de même pour l'événement 4 sauf qu'il remplace un événement dans la base.

Enfin, le dernier événement va servir à mettre à jour les champs d'un événement de contact existant. Dans les points qui vont suivre, nous allons présenter de manière exhaustive la liste de tous les filtres impliqués dans le système de colportage.

### 3.5.2. Liste des filtres

#### 3.5.2.1 Filtres permanents (anti-redondance)

La fonction principale des filtres permanents (Fp) est d'éliminer les doublons dans la base de connaissance du nœud. Trois filtres constituent la classe des filtres permanents : le filtre doublon, le filtre dual et le filtre concaténation. Dans les points qui vont suivre, nous prendrons comme exemple les deux événements de contact appelés dans ce qui suit CEa et CEb avec : CEa (Ida, Idb, NSa, T1, T2) et CEb (Idb, Ida, NSb, T'1, T'2) représentant le même contact entre deux nœuds A et B avec respectivement leurs numéros de séquence NSa et NSb ainsi que leurs durées [T1, T2] et [T'1, T'2].

##### *a. Le filtre doublon*

Le filtre doublon doit détecter si les événements de contact représentent le même contact avec la même pertinence. Rappelons qu'un événement de contact est identifié de manière unique par le premier champ de la structure de données événement de contact (cf Figure 54) c'est-à-dire l'identifiant du nœud et le numéro de séquence. Deux événements CEa et CEb sont donc considérés comme étant doublons s'ils possèdent les mêmes identifiants Ida, Idb et le même numéro de séquence NSa. L'algorithme 1 détaille l'implémentation du filtre doublon.

|  |
|--|
| <b>Algorithme 1</b> Algorithme du filtre doublon   |
| <b>Début</b>   |
| <b>Pré-requis</b> : CEa (Ida, Idb, NSa, T1, T2) = nouvel événement de contact créé ou reçu par le nœud appliquant l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud. |
| <b>Pour</b> chaque événement dans la table de connaissance <b>faire</b>  |
| <b>Si</b> table de connaissance [i].Ida = Ida <b>et</b> table de connaissance [i].Ida = Ida <b>et</b> table de connaissance [i].NS = NSa <b>Alors</b> Ne pas insérer le nouvel événement dans la table.              |
| <b>Fin Si.</b>   |
| <b>Fin Pour</b>  |
| <b>Fin</b>   |

##### *b. Le filtre dual*

A chaque contact confirmé, deux événements de contact sont créés désignant le même contact par les deux entités impliquées dans cette rencontre. Ces deux événements de contact CEa et CEb sont appelés duaux si leur période de contact, respectivement [T1, T2] et [T'1, T'2] sont équivalentes



(c'est-à-dire  $T1 \equiv T'1$  et  $T2 \equiv T'2$ ). Le choix de garder en mémoire soit le contact CEa ou CEb dépend du nœud récepteur Sk, si l'identifiant  $Ida = Sk$  alors le nœud Sk garde en mémoire le contact CEa. Dans le cas où l'identifiant  $Idb = Sk$  alors, le nœud Sk sauvegarde le contact CEb. Le fonctionnement en détail du filtre dual est expliqué dans l'algorithme 2.

**Algorithme 2** Algorithme du filtre dual

**Début**

**Pré-requis :** CEa ( $Ida, Idb, NSa, T1, T2$ ) = nouvel événement de contact créé ou reçu par le nœud appliquant l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud.

**Pour** chaque événement dans la table de connaissance **faire**

**Si** table de connaissance [i]. $Ida = Ida$  **et** table de connaissance [i]. $Idb = Idb$  **et** table de connaissance [i]. $T1 \equiv T1$  **et** table de connaissance [i]. $T2 \equiv T2$  **alors**

**Si** identifiant du nœud récepteur =  $Ida$  **alors** garder en mémoire cette version de l'événement CEa.

**Fin Si**

**Sinon Si** identifiant du nœud récepteur =  $Idb$  **alors** garder l'événement déjà stocké en mémoire.

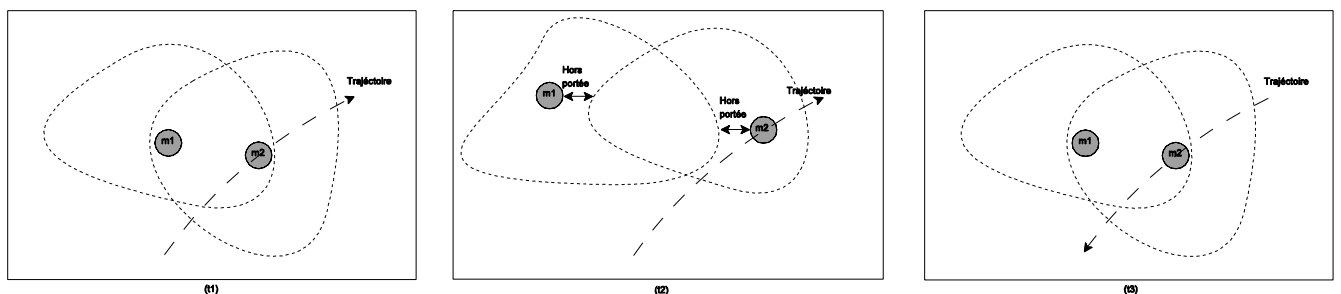
**Fin Si.**

**Fin Pour**

**Fin**

**c. Le filtre concaténation**

Deux contacts voire plusieurs peuvent se succéder avec un intervalle de temps très court, le contact radio est interrompu pendant un court instant puis rétabli. Cette situation peut être provoquée par exemple lorsqu'un nœud se retrouve à portée lors d'un aller-retour avec un second nœud comme illustré dans la figure 58.



**Figure 58 : Succession de contacts lors d'un aller-retour.**

L'expérimentation a aussi montré que les frontières d'une cellule n'étaient pas délimitées de façon stable et que l'approche de deux nœuds mobiles se traduisait très souvent par plusieurs contacts

rapprochés dans le temps. La conséquence de ce type de rupture est le nombre important d'événements de contact créés vainement qui représentent en fin de compte un même contact. Pour pallier ce problème de succession d'événements découpés par des intervalles de temps très courts, le filtre concaténation a été implémenté pour regrouper des événements de contact s'ils sont séparés par un temps inférieur à un certain seuil temporel. Le fonctionnement de ce filtre est donné dans la figure 59 où nous reprenons l'exemple précédant avec tous les événements créés et regroupés suite à l'application du filtre concaténation.

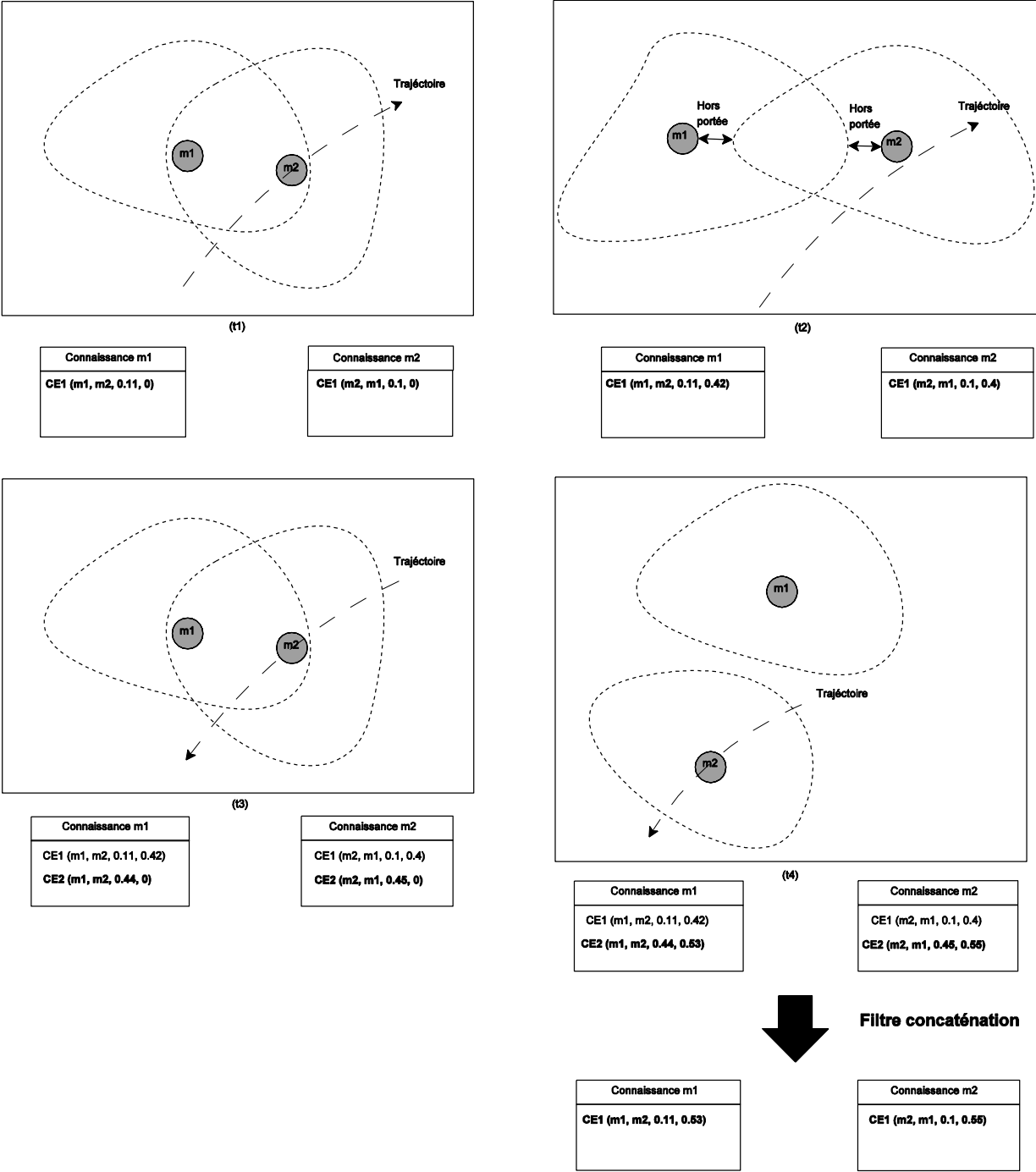


Figure 59 : Fonctionnement du filtre concaténation.

Suite à l'aller-retour effectué par le mobile m2, deux événements de contact sont créés par ce mobile : CE1 et CE2 (la même opération est effectuée en parallèle chez le mobile m1). A la fin du second contact (t4), le filtre concaténation va calculer la différence de temps entre la fin de contact de CE1 et le début de CE2. Cette différence étant inférieure au seuil fixé, les deux événements vont être par la suite fusionnés en un seul.

L'algorithme 3 suivant, explique le fonctionnement de ce filtre.

### **Algorithme 3** Algorithme du filtre concaténation

#### **Début**

**Pré-requis :** CEa (Ida, Idb, NSa, T1, T2) = nouvel événement de contact créé ou reçu par le nœud appliquant l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud.

**Pour** chaque événement dans la table de connaissance **faire**

**Si** table de connaissance [i].Ida = Ida **et** table de connaissance [i].Idb = Idb **Ou**

table de connaissance [i].Ida = Idb **Et** table de connaissance [i].Idb = Ida **Alors Si** Valeur absolue (T1 – table connaissance [i].T2) <= Seuil **Alors** choisir (min (T1, table connaissance[i].T1), max (T2, table connaissance[i].T2)).

**Fin Si**

**Sinon Si** Valeur absolue (T1 – table connaissance [i].T1) <= Seuil **Et** connaissance [i].T2 = 0 **Alors** choisir (min (T1, table connaissance[i].T1)).

**Fin Si**

**Sinon Si** Valeur absolue (T1 – table connaissance [i].T2) <= Seuil **Et** T2 = 0 **Alors** Connaissance[i].T2 = 0.

**Fin Si**

**Sinon Si** Valeur absolue (table connaissance [i].T1 – T2) <= Seuil **Et** table connaissance [i].T2 = 0 **Alors** T1 = connaissance [i].T1.

**Fin Si**

**Fin Pour**

**Fin**

### **3.5.2.2 Filtres de gestion de mémoire**

Les filtres qui vont suivre sont optionnels et appliqués dans les cas où la mémoire des nœuds du réseau est saturée. Chaque application peut appliquer dans ce cas un ou plusieurs filtres selon ses besoins.

### *a. Le filtre aléatoire*

Ce filtre est l'un des plus simples à appliquer et à mettre en place. Lorsque la mémoire est saturée, chaque nœud dans le réseau peut appliquer un filtre aléatoire avec une probabilité  $P$ . Ce paramètre aléatoire a été introduit pour ne pas filtrer systématiquement chaque fois qu'une place doit être libérée et privilégier tout le temps les contacts les plus récents au détriment des anciens. L'algorithme 4 détaille le fonctionnement de ce filtre.

#### **Algorithme 4** Algorithme du filtre aléatoire

##### **Début**

**Pré-requis :** CEa (Ida, Idb, NSa, T1, T2) = nouvel événement de contact créé ou reçu par le nœud appliquant l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud.

Paramètre du filtre :  $p$ , une valeur entre  $[0, 1]$

Soit  $P$  une valeur aléatoire tirée dans l'intervalle  $[0, 1]$

**Si  $P > p$  Alors faire une place au nouvel élément :** Trouver une position aléatoire dans la table de connaissance **Et** Insérer nouvel événement de contact.

**Fin Si.**

**Fin**

### *b. Le filtre ancienneté*

Le filtre ancienneté a pour rôle de supprimer les événements de contact les plus anciens. Ce filtre traite à la fois les événements complets et incomplets. Il commence d'abord par les événements finis (complets) en recherchant le premier événement fini et compare tous les autres événements au premier rencontré. Lorsque le plus ancien des événements de contact est trouvé, il sera remplacé par l'événement rentrant. Les mêmes étapes sont appliquées par la suite aux événements incomplets. Ce filtre est une solution de secours lorsque les autres filtres n'arrivent pas à libérer de la place mémoire. D'autres variantes existent pour ce filtre comme pour l'avantage balise et le filtre traçabilité. Nous expliquerons plus en détail dans ce qui suit le rôle de ces deux variantes. Les spécifications du filtre ancienneté sont développées dans l'algorithme 5.

#### **Algorithme 5** Algorithme du filtre ancienneté

##### **Début**

**Pré-requis :** CEa (Ida, Idb, NSa, T1, T2) = nouvel événement de contact créé ou reçu par le nœud qui

applique l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud.

// traitement des événements finis

// rechercher le premier événement fini

**Pour** chaque événement dans la table de connaissance **faire**

**Si** table connaissance [i].T2 > 0 **Alors** sauvegarder cet événement dans la variable ancien\_fini.

**Fin Si**

**Fin Pour**

// rechercher le plus ancien des événements fini et sauvegarder sa position

**Pour** chaque événement dans la table de connaissance **faire**

**Si** table connaissance [i].T2 < ancien\_fini.T2 **Et** table connaissance [i].T2 != 0 **Alors** ancien\_fini = table connaissance [i] **Et** position\_fini = i.

**Fin Si**

**Fin Pour**

// remplacer l'événement le plus ancien par l'événement rentrant

table connaissance [position\_fini] = événement rentrant

// traitement des événements non finis

// rechercher le premier événement non fini

**Pour** chaque événement dans la table de connaissance **faire**

**Si** table connaissance [i].T2 = 0 **Alors** sauvegarder cet événement dans la variable ancien\_non\_fini.

**Fin Si**

**Fin Pour**

// rechercher le plus ancien des événements non fini et sauvegarder sa position

**Pour** chaque événement dans la table de connaissance **faire**

**Si** table connaissance [i].T2 = 0 **Et** table connaissance [i].T1 <= ancien\_non\_fini.T2 **Alors** ancien\_non\_fini = table connaissance [i] **Et** position\_non\_fini = i.

**Fin Si**

**Fin Pour**

// remplacer l'événement le plus ancien par l'événement rentrant

table connaissance [position\_non\_fini] = événement rentrant

**Fin**

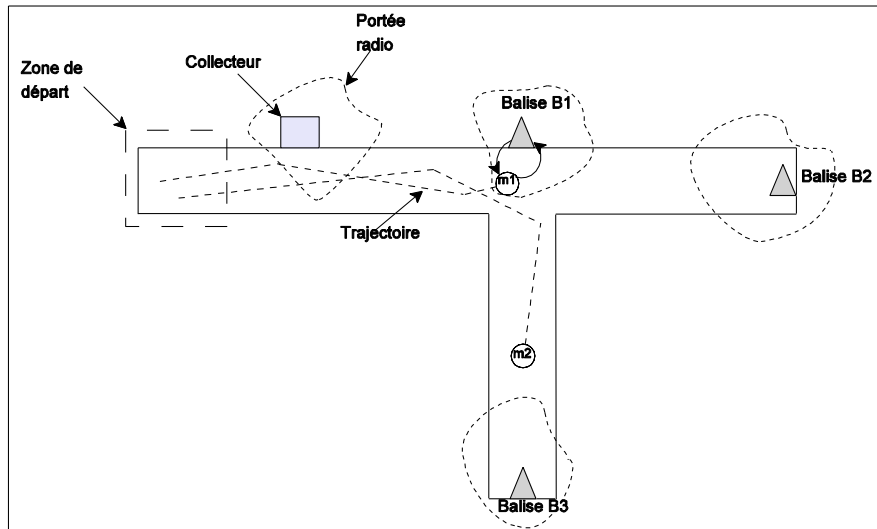
### 3.5.2.3 Filtres applicatifs

#### a. *Les filtres de localisation (application LoWCA)*

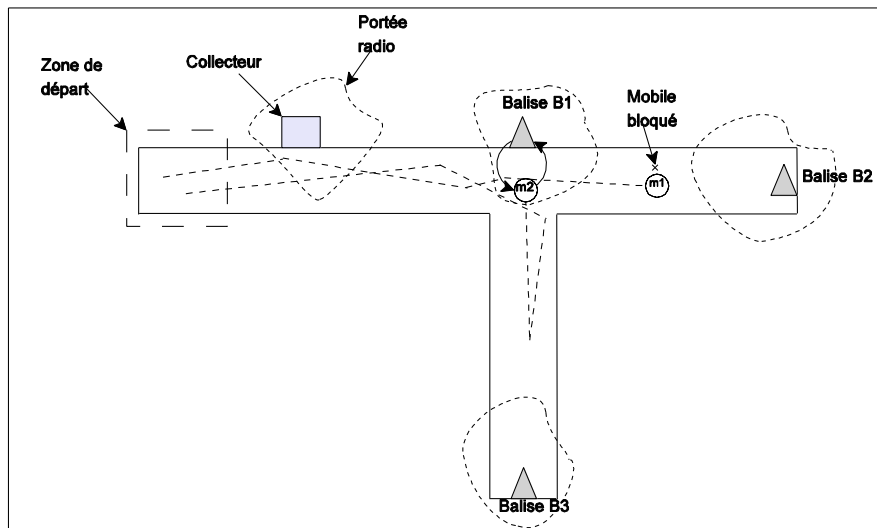
Les filtres impliqués dans le processus de localisation d'entités mobiles (application LoWCA) sont le filtre avantage balise avec ces deux versions : aléatoire et le plus ancien, ainsi que le filtre localisation.

##### a.1. Filtre avantage balise (aléatoire)

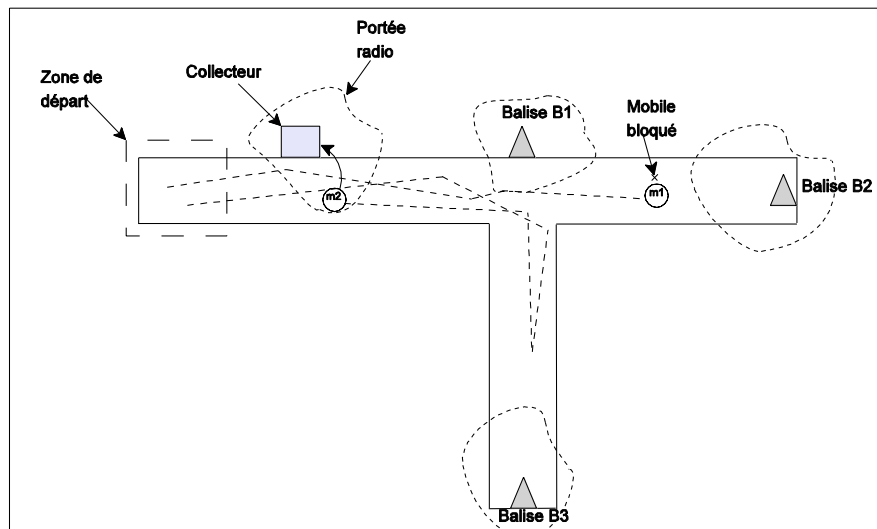
Les événements de contact impliquant des balises peuvent être cruciaux pour des applications de localisation a posteriori. Ces événements collectés au niveau du point de collecte donnent des informations de localisation pour une entité mobile donnée. En effet, les entités fixes du réseau (appelées balises), en plus de jouer le rôle de relais d'information pour les entités mobiles, sont déployées à des placements stratégiques pour des besoins de localisation. Ensuite, les événements de contact issus de rencontres impliquant des balises et des nœuds mobiles peuvent servir à alimenter les applications de localisation. Pour montrer l'utilité de cette solution, nous reprenons l'exemple de la galerie minière dans la figure 60 où une station mobile m1 reste bloquée dans une mine. Lors du passage du second mobile m2 à proximité de la balise b1 (t2), ce colporteur mobile va récupérer une connaissance très importante pour un processus de localisation : *l'évènement de contact qui indique la dernière fois que le nœud m1 a été vu à proximité de la balise b1*. Ensuite, le nœud m2 va colporter cette connaissance jusqu'au collecteur qui, lui, va alimenter l'application de localisation (t3). A travers cet exemple, on perçoit l'intérêt d'une telle solution de localisation lorsqu'une station mobile peut avoir une information sur une autre station sans jamais la croiser. Ce type de solution appelé localisation par zone ou par cellule est très satisfaisant dans des milieux confinés [19].



(t1)



(t2)



(t3)

Figure 60 : Exemple d'une collecte indirecte d'informations concernant des nœuds non rencontrés.

Le fonctionnement du filtre avantage balise est le suivant :

- Une priorité est donnée aux événements de contact de type balise-mobile.
- Le filtre est appliqué lorsqu'un nœud à la mémoire saturée, reçoit un événement de type balise-mobile.
- Le filtre cherche une place libre de type mobile-mobile aléatoirement :
  1. Si l'événement existe, il sera remplacé automatiquement par l'événement de type balise-mobile.
  2. Dans le cas où aucun événement de type mobile-mobile n'existe, le filtre aléatoire est appliqué aux événements de type balise-mobile avec une probabilité  $p$ .

L'algorithme du filtre avantage balise est le suivant :

|  |
|--|
| <b>Algorithme 6</b> Algorithme du filtre avantage balise   |
| <b>Début</b>   |
| <b>Pré-requis :</b> CEa (Ida, Idb, NSa, T1, T2) = nouvel événement de contact créé ou reçu par le nœud appliquant l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud. |
| // vérifier que l'événement reçu est de type balise-mobile   |
| <b>Si</b> CEa.ida est de type balise <b>Ou</b> CEa.idb est de type balise <b>Alors</b>   |
| Position i = position aléatoire de type mobile-mobile.   |
| <b>Si</b> i existe <b>Alors</b>  |
| Table de connaissance [i] = CEa.   |
| <b>Sinon</b>   |
| // appliquer un filtre aléatoire avec une probabilité $p$  |
| $p$ = paramètre du filtre compris entre [0, 1]   |
| <b>Si</b> P une valeur aléatoire est telle que $P > p$ <b>Alors</b> Trouver une position aléatoire dans la table de connaissance <b>Et</b> Insérer nouvel événement de contact.                                      |
| <b>Fin Si.</b>   |
| <b>Fin Si</b>  |
| <b>Fin Si</b>  |
| <b>Fin</b>   |



### a.2. Filtre avantage balise (plus ancien)

Ce filtre est une variante du précédent filtre avantage balise en prenant en compte les événements de contact les plus anciens. La différence avec la première version est que lorsque le filtre doit libérer de l'espace mémoire parmi les événements de type mobile-mobile, il doit chercher le plus ancien de ces événements. Si aucun événement de contact de type mobile-mobile n'est trouvé, le filtre aléatoire avec la probabilité  $P$  est appliqué aux événements de contact de type balise-mobile.

#### Algorithme 7 Algorithme du filtre avantage balise (plus ancien)

##### Début

**Pré-requis :** CEa (Ida, Idb, NSa, T1, T2) = nouvel événement de contact créé ou reçu par le nœud appliquant l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud.

// vérifier que l'événement reçu est de type balise-mobile

**Si** CEa.ida est de type balise **Ou** CEa.idb est de type balise **Alors**

Position  $i$  = Rechercher l'événement de type mobile-mobile le plus ancien.

**Si**  $i$  existe **Alors**

Table de connaissance [ $i$ ] = CEa.

**Sinon**

// appliquer un filtre aléatoire avec une probabilité  $p$

$p$  = paramètre du filtre compris entre [0, 1]

**Si**  $P$  une valeur aléatoire est telle que  $P > p$  **Alors** Trouver le plus ancien événement de contact dans la table de connaissance **Et** Insérer nouvel événement de contact à sa place.

**Fin Si.**

**Fin Si**

**Fin Si**

**Fin**

### b. Filtre localisation

Ce filtre est dédié particulièrement aux applications qui tentent de localiser des entités mobiles au sein d'un réseau mobile. Pour ce faire, le filtre localisation favorise à chaque insertion d'un événement dans la base de connaissance d'un nœud, la connaissance la plus récente. Son fonctionnement est le suivant :

- Pour chaque événement reçu, le filtre doit chercher dans la base de connaissance tous les événements concernant les mêmes partenaires.
- Pour chaque couple d'événements de contact impliquant les mêmes partenaires, garder le plus récent.

Un exemple du fonctionnement de ce filtre est représenté dans la figure 61. Lorsqu'un nœud reçoit un événement de contact, après avoir appliqué les filtres permanents, le filtre applicatif de localisation va parcourir toute la table de connaissance à la recherche des événements concernant les deux mêmes identités. Etant donné que ce nouvel événement est le plus récent parmi tous les autres déjà en mémoire, il sera inséré dans la base de connaissance.

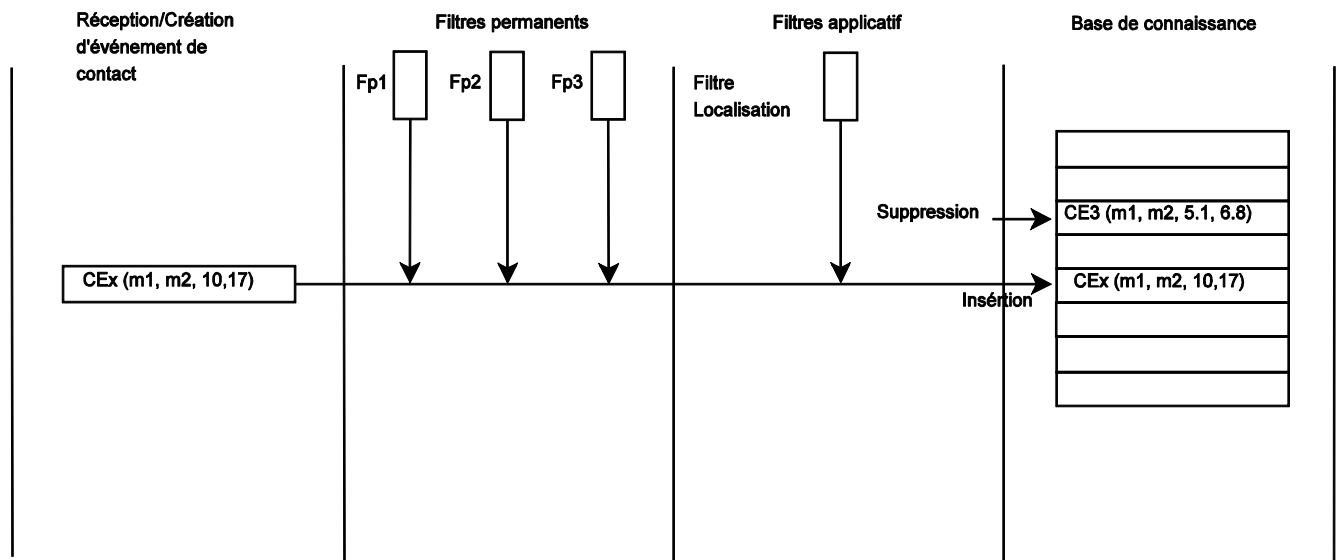


Figure 61 : Exemple de fonctionnement du filtre de localisation.

L'algorithme du filtre localisation est le suivant :

**Algorithme 8** Algorithme du filtre localisation

**Début**

**Pré-requis :** CEa (Ida, Idb, NSa, T1, T2) = nouvel événement de contact créé ou reçu par le nœud appliquant l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud.

// vérifier que l'événement reçu est de type balise-mobile

**Si** CEa.ida est de type balise **Ou** CEa.idb est de type balise **Alors**

// rechercher dans la table les événements concernant les mêmes partenaires

**Pour** chaque événement dans la table de connaissance **faire**

// vérifier les identifiants des partenaires

```

Si table de connaissance [i].Ida = Ida et table de connaissance [i].Idb = Idb Ou table de connaissance [i].Ida = Idb Et table de connaissance [i].Idb = Ida Alors
// traitement des événements finis
Si T2 > 0 Et table connaissance [i].T2 > 0 Alors
// comparaison des dates de fin et de début
Si T2 <= table connaissance [i].T1 Et Alors garder l'événement déjà en mémoire Sinon Si table connaissance [i].T2 <= T1 Alors supprimer événement en mémoire Et insérer nouvel événement de contact Fin Si
Fin Si
// traitement des événements non finis
Si T2 = 0 Et table connaissance [i].T2 = 0 Alors
// comparaison des dates de début seulement
Si table connaissance [i] <= T1 Alors supprimer événement en mémoire Et insérer nouvel événement de contact.
Fin Si
Fin Pour
Fin

```

### *c. Les filtres de traçabilité*

La dernière catégorie des filtres impliqués dans le processus de colportage de connaissance est le filtre de traçabilité avec ces deux variantes : aléatoire et le plus ancien. Ce filtre est semblable au filtre avantage balise décrit plus haut, car il affecte une priorité à une entité donnée. C'est le cas, par exemple, lors de la localisation d'une entité mobile (personnel ou matériel) au sein d'une galerie minière.

#### **b.1. Filtre traçabilité (aléatoire)**

La première version du filtre traçabilité a pour but de libérer de l'espace mémoire de manière aléatoire parmi les événements de contact pour lesquels l'entité concernée par le processus de traçabilité n'est pas impliquée. Son fonctionnement est résumé dans les points suivants :

- A la réception d'un événement de contact, l'algorithme vérifie si l'entité Idx concernée par le processus de traçabilité est impliquée dans le contact.
- Si le nœud Idx est impliqué dans le contact, l'algorithme va lire la liste d'événements de contact dans la base de connaissance n'impliquant pas l'entité Idx (x : est l'identité du nœud concerné par la traçabilité).

- L'algorithme va ensuite tirer une place au sort une place parmi cette liste et insérer le nouvel événement de contact reçu.
- Si tous les contacts sont impliqués (en relation avec Idx) il faut enchaîner par un autre filtre (ancien, aléatoire,...).

Un exemple filtrage aléatoire est donné dans la figure 62.

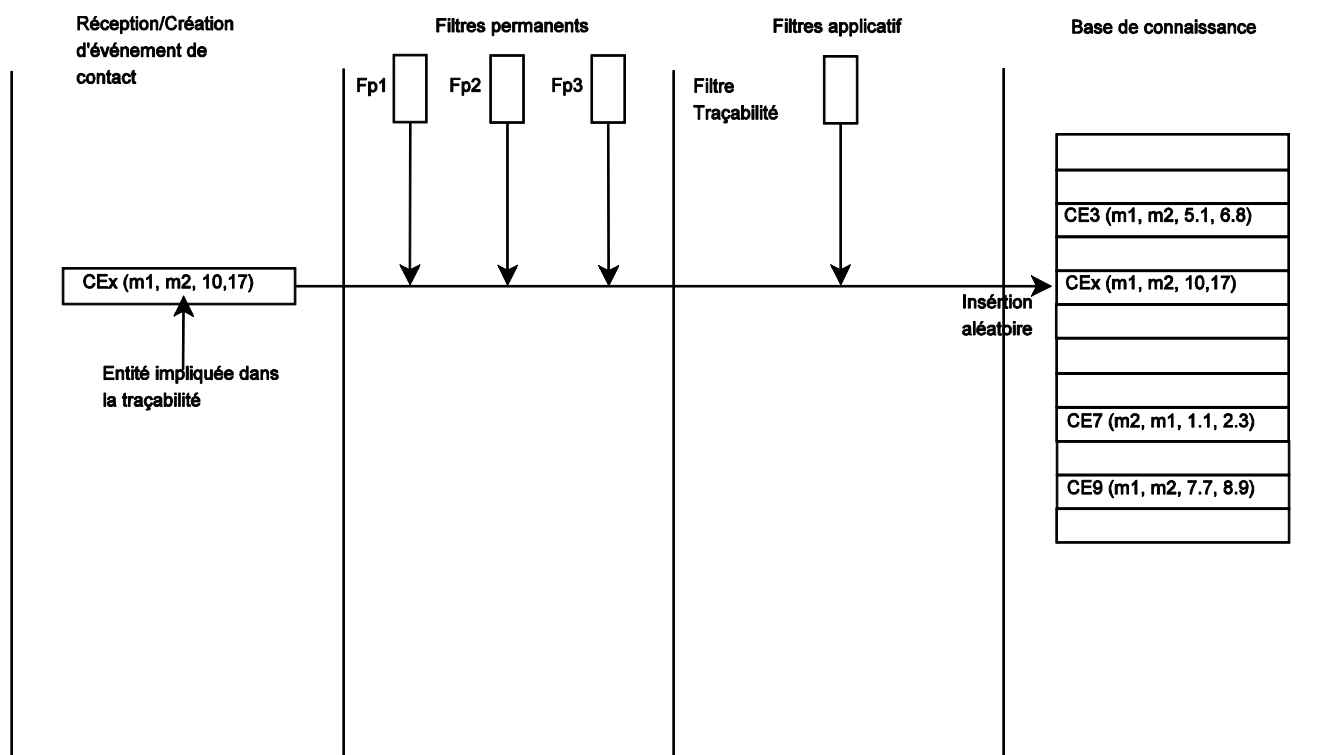


Figure 62 : Exemple de fonctionnement du filtre traçabilité aléatoire.

L'algorithme du filtre traçabilité aléatoire est le suivant :

**Algorithme 9** Algorithme du filtre traçabilité (aléatoire)

**Début**

**Pré-requis :**  $CEa( Ida, Idb, NSa, T1, T2)$  = nouvel événement de contact créé ou reçu par le nœud appliquant l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud.

Paramètre : Idx = identité de l'entité impliquée dans le contact.

// vérifier si l'entité concernée par le processus de traçabilité est impliquée dans le contact.

**Si**  $Ida = Idx$  **Ou**  $Idb = Idx$  **Alors**

// compter le nombre d'événements de contact n'impliquant pas l'entité Idx

**Pour** chaque événement dans la table de connaissance **faire**

**Si** table de connaissance [i].Ida != Idx **Et** table de connaissance [i].Idb != Idx **Alors** Nb Occurrence ++ .

**Fin Si**

**Fin Pour**

// choisir une place aléatoirement parmi les entités non concernées par la traçabilité

P = valeur aléatoire dans l'intervalle [1 : Nb Occurrence].

// insérer le nouvel événement de contact

Insérer nouvel événement de contact en position P de la liste des événements non impliqués

**Fin Si**

**Fin**

## b.2. Filtre traçabilité (le plus ancien)

Cette version du filtre traçabilité libère une place mémoire en cherchant le plus ancien des événements de contact qui ne concerne pas l'entité Idx impliquée dans la traçabilité. Le fonctionnement de ce filtre est le suivant :

- Le filtre doit d'abord vérifier si l'entité Idx est impliquée dans le nouveau contact reçu.
- Si oui, l'algorithme de filtrage va chercher le plus ancien des événements de contact stockés en mémoire qui ne contient pas Idx.
- Le contact reçu va être inséré à cette position.

L'algorithme du filtre traçabilité le plus ancien est donné ci-dessous :

### **Algorithme 10** Algorithme du filtre traçabilité (le plus ancien)

**Début**

**Pré-requis** : CEa (Ida, Idb, NSa, T1, T2) = nouvel événement de contact créé ou reçu par le nœud appliquant l'algorithme. table de connaissance [] = tableau contenant tous les événements de contact du nœud.

Paramètre Idx = identité de l'entité impliquée dans le contact.

// vérifier si l'entité concernée par le processus de traçabilité est impliquée dans le contact.

**Si** Ida = Idx **Ou** Idb = Idx **Alors**

// chercher le plus ancien des événements de contact non concerné par la traçabilité

Chercher la position P du plus ancien

```
// insérer le nouvel événement de contact
Insérer nouvel événement de contact en position P
Fin Si
Fin
```

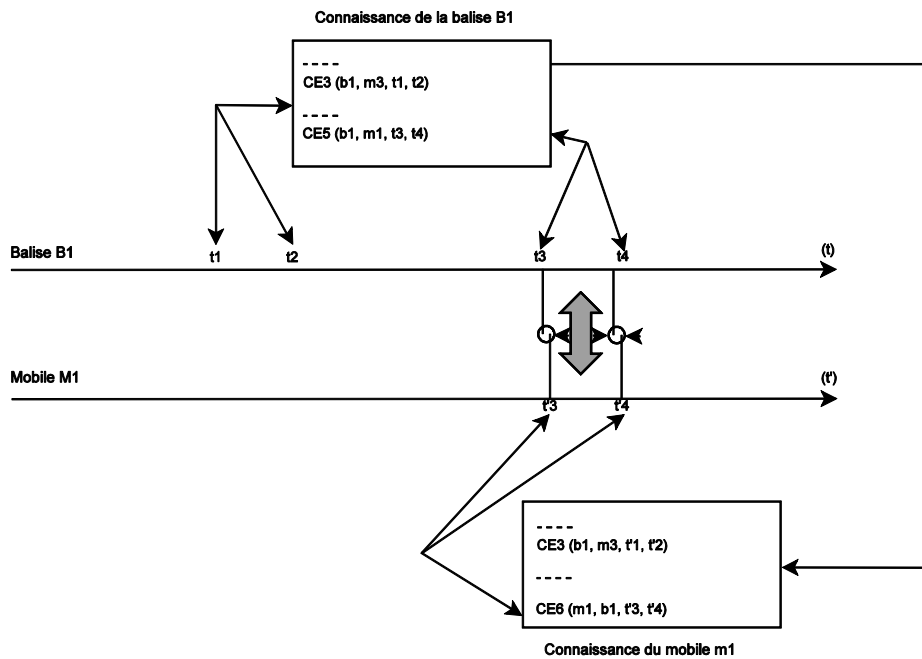
### 3.6. Synchronisation

Le mécanisme de synchronisation proposé dans ce travail s'inspire à la fois des méthodes de synchronisation de type point-à-point et de type maître-esclave [12] où chaque nœud diffuse son horloge locale dans le réseau dans le but de réaliser une synchronisation globale. Cette synchronisation est orchestrée par le nœud collecteur qui joue le rôle pour l'ensemble des autres nœuds. Le fonctionnement de notre mécanisme est résumé dans les deux points suivants :

- Au début de chaque activité d'un nœud fixe ou mobile, son horloge n'est pas synchronisée.
- C'est le nœud collecteur qui synchronise les nœuds mobiles lors de leur passage à proximité.
- C'est le premier mobile qui visite un nœud balise qui synchronise son horloge.
- Chaque nœud dans le réseau ensuite diffuse son horloge locale à chaque transmission d'une trame (de signalement ou de diffusion).
- Lors d'un échange de connaissance entre deux entités à portée, tous les événements de contact reçus sont systématiquement datés avec l'horloge du récepteur. Le nœud récepteur doit corriger les dates de début et de fin de ces événements en fonction du décalage des horloges observées entre les deux entités en contact.

L'idée n'est pas de synchroniser les horloges des différentes entités mais de faire en sorte que le contenu de la base de connaissance de chaque nœud soit cohérent avec son horloge locale.

Un exemple illustrant la synchronisation et la correction des dates portées par les événements lors d'un échange entre deux entités est donné dans la figure 63.



**Figure 63 : Datation des événements de contact lors d'un échange entre nœuds.**

Initialement, avant le début du contact, la balise B1 possède dans sa base de connaissance l'événement de contact CE3 qui représente l'enregistrement qu'a eu lieu entre cette station et un autre mobile M3 entre les instants  $t_1$  et  $t_2$  (heure de B1). Ensuite, entre les instants  $t_3$  et  $t_4$  (horloge locale de B1), les deux entités B1 et M1 entrent en contact pour échanger leur connaissance. La différence entre l'heure de l'émetteur portée par la trame et son heure de réception permet au récepteur d'estimer la différence entre son horloge et celle de l'émetteur. Dans l'exemple de la figure 69, lorsque la station M1 a estimé son décalage d'horloge avec son voisin en recevant l'évènement de contact de B1, elle doit corriger les dates de cet évènement reçu pour le dater avec son horloge locale ( $t'_3$  et  $t'_4$ ).

Cette approche de synchronisation dite faible (de type point à point) s'avère intéressante car le but est seulement de synchroniser les événements entre eux.

# Chapitre 4 : Evaluation de LoWCA

## 4.1. Introduction

Dans ce chapitre, nous proposons une démarche d'évaluation de notre modèle générique en grande partie par des simulations réalisées à partir de l'outil NS2, nous aborderons d'abord la détection de proximité suivie d'une partie consacrée aux effets de la charge d'une cellule sur la pertinence du mécanisme de détection. Ces deux premières étapes ont pour but d'évaluer les performances de la méthode d'accès CSMA/CA non slottée de 802.15.4 et ses effets sur la détection de proximité. Le mécanisme de routage sera ensuite évalué en étudiant les effets des filtres sur la qualité de la connaissance colportée jusqu'au point de collecte, cette qualité aura un impact sur la précision de l'application de localisation a posteriori. Dans le but de valider la faisabilité de notre mécanisme, nous terminerons ce chapitre par une partie où nous exposerons les résultats d'expérimentations obtenus par maquettage afin de les comparer à ceux obtenus par simulation.

## 4.2. Simulation

Le processus de simulation à l'aide d'outils informatiques, généralement des ordinateurs dotés d'une capacité de calcul et de stockage importante, permet de simuler des phénomènes réels parfois très complexes. A titre d'exemple, nous pouvons citer : la chute d'un corps sur un support mou, la résistance d'une plateforme pétrolière, la faiblesse d'un matériau sous une sollicitation vibratoire, le déplacement d'une population mobile ...etc). Plusieurs simulations ont été effectuées dans le but d'évaluer les performances de l'application cible LoWCA. Pour cela, nous avons exploité le simulateur NS2 [76] pour l'implémentation de l'environnement de simulation, du modèle de déplacement et des algorithmes nécessaires aux échanges et au routage de l'information.

NS2 (Network Simulator 2) est un logiciel libre de simulation par événements discrets très répandu dans le domaine de la recherche académique et dans l'industrie. Plusieurs modèles de protocoles dont UDP, TCP/IP, la couche MAC 802.11 et 802.15.4 sont mis à disposition. Il est aussi tout à fait possible d'ajouter d'autres protocoles au simulateur en respectant la classification en couches du modèle OSI (Open Systems Interconnection). Le simulateur est implémenté essentiellement à partir du langage C++ et se sert des scripts Tcl/Tk pour la configuration des simulations. Le logiciel NS2 est fourni avec une interface graphique (NAM : Network Animator) permettant de donner un aspect graphique, mais très basique, du fonctionnement du réseau.



## 4.2.1. Les modèles de propagation

Nous allons à présent décrire les modèles de propagation les plus recensés dans la littérature qui permettent une modélisation du signal radio que ce soit pour des réseaux de type WLAN (*Wireless Local Area Network*) ou WPAN (*Wireless Personal Area Network*). En effet, la nature du médium, la distance parcourue, la présence d'objets et l'atténuation du signal ont un impact considérable sur la qualité du signal et sur la puissance du signal reçu, ce qui implique parfois une incompréhensibilité des données reçues. Les modèles existants dans la littérature que nous allons citer sont souvent paramétrés selon les conditions des environnements étudiés.

### 4.2.1.1. Le modèle Free Space

Ce modèle est aussi appelé modèle de Friis [77], c'est l'un des modèles les plus simples. En effet, ce modèle considère que le médium et les antennes sont parfaits et qu'aucun obstacle n'existe entre l'émetteur du signal et le récepteur. Sur la figure 64, la portée du signal est représentée en cercle (ou une sphère en trois dimensions) et chaque entité réceptrice se trouvant à l'intérieur de ce cercle (ou cette sphère) recevra automatique le paquet émis par la source avec une puissance reçue qui ne dépendra que de la distance qui sépare l'émetteur du récepteur.

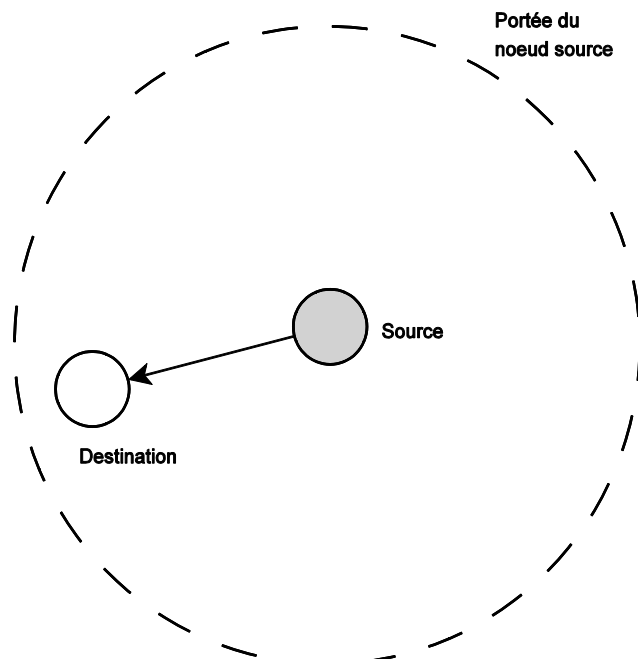


Figure 64 : Modèle Free Space.

La puissance du signal reçu par un nœud est donnée par la formule de *Friis* [78] suivante :

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2}$$

Le paramètre  $P_t$  représente la puissance du nœud émetteur,  $G_t$  et  $G_r$  représentent quant à eux les gains des antennes du nœud émetteur et récepteur. Le paramètre  $d$  représente la distance entre l'émetteur et le récepteur,  $\lambda$  la longueur d'onde du signal.

#### 4.2.1.2. Le modèle Two-Ray ground

Dans le premier modèle, la propagation du signal est supposée s'effectuer dans un environnement libre ce qui n'est pas réaliste dans un environnement réel où souvent les obstacles existent [79]. Dans ce modèle, le signal est supposé reçu selon deux chemins possibles : un premier chemin direct et un second dû à une réflexion sur le sol comme illustré dans la figure 65.

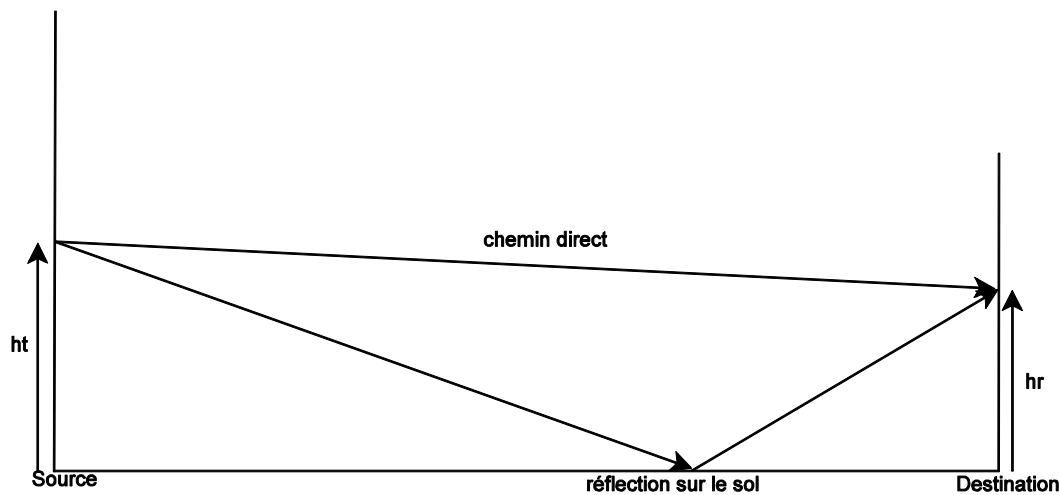


Figure 65 : Modèle Two-Ray ground.

La puissance reçue par un nœud peut être exprimée par la formule suivante :

$$P_r(d) = \frac{P_t G_t G_r h_t^2 h_r^2}{d^4}$$

Où les paramètres  $ht$  et  $hr$  représentent les hauteurs des antennes de transmission et de réception.

En comparaison avec le premier modèle Free Space, les résultats obtenus dans [79] avec le modèle *Two-ray ground* sont plus précis lorsqu'il s'agit de grandes distances, car il tient compte des variations

de la puissance reçue causées par la combinaison constructive et destructive des deux signaux considérés. Quand les distances sont petites, le modèle Free Space offre des résultats plus proches de la réalité car le trajet direct est dominant.

#### 4.2.1.3. Le modèle Shadowing

Ce modèle tente de se rapprocher des conditions réelles de propagation du signal en remettant en cause le fait que la portée à la fois dans les modèles *Free Space* et *Two-ray ground* soit un cercle parfait ou une sphère. Pour ce faire, le modèle *Shadowing* [79] ajoute une nouvelle composante aléatoire pour caractériser les effets de l'environnement sur la propagation du signal. Pour une même distance parcourue la qualité et la puissance du signal reçu dépendent notamment des obstacles rencontrés, cette composante aléatoire modélise cette dispersion.

#### 4.2.1.4. Le modèle ITU-R P1238-4

Ce modèle a été développé et proposé par l'organisme de normalisation international du domaine de télécommunication ITU (*International Telecommunication Union*) [80]. Des expérimentations ont été réalisées [81] afin de paramétrer ce modèle et comparer ses résultats aux autres modèles de propagation précédemment cités. Les résultats obtenus prouvent que le modèle ITU convenablement paramétré pour un environnement donné permet d'obtenir des résultats mieux adaptés à des milieux confinés que ceux donnés par Free Space, Two-ray ground et Shadowing.

La puissance reçue est estimée par la formule suivante :

$$P_{r_{mw}} = \exp\left(\frac{P_{t_{dBm}} - L}{10} \text{Log}_{10}\right) / 1000, \text{ (en mW)}$$

$$L = 20\text{Log}_{10}(f) + 10 * N\text{Log}_{10}(d) + pfn - 28 + \text{random},$$

Où  $P_{r_{mw}}$  dans la formule est la puissance de réception exprimée en *mW*,  $P_{t_{dBm}}$  est la puissance de transmission en *dBm*.  $L$  représente l'atténuation (*Path Loss*),  $f$  est la fréquence en *MHz*, le paramètre  $N$  est un coefficient d'atténuation,  $d$  représente la distance en mètre,  $pfn$  est le facteur d'atténuation utile quand émetteur et récepteur ne sont pas sur le même étage et *random* est une variable aléatoire suivant une loi normale.

Rappelons que notre domaine de recherche vise à proposer un modèle générique pour des applications en milieux confinés, par exemple, lorsqu'il s'agit d'applications de localisation ou de traçabilité. Il est

évident que notre choix se dirige vers le modèle ITU qui est le plus générique et le plus approprié pour représenter les conditions de propagation dans ses milieux d'intérieurs.

### 4.3. Evaluation de LoWCA

#### 4.3.1. Evaluation du comportement d'une cellule

Dans cette première partie où nous exposerons les premiers résultats de simulation, nous allons évaluer les performances de la méthode d'accès au médium CSMA/CA, et plus précisément de la version non slottée de la norme 802.15.4 [10]. Les premiers résultats exposés vont concerner la détection de proximité dans une cellule et plus précisément l'impact de la densité des nœuds mobiles dans une cellule sur la méthode d'accès au médium. L'objectif est de détecter un voisin ce qui va déclencher la création d'un événement de contact. La simulation va nous permettre d'évaluer les effets du médium et de la couche MAC sur le processus de création d'un événement de contact.

##### 4.3.1.1. Topologie et conditions de simulations

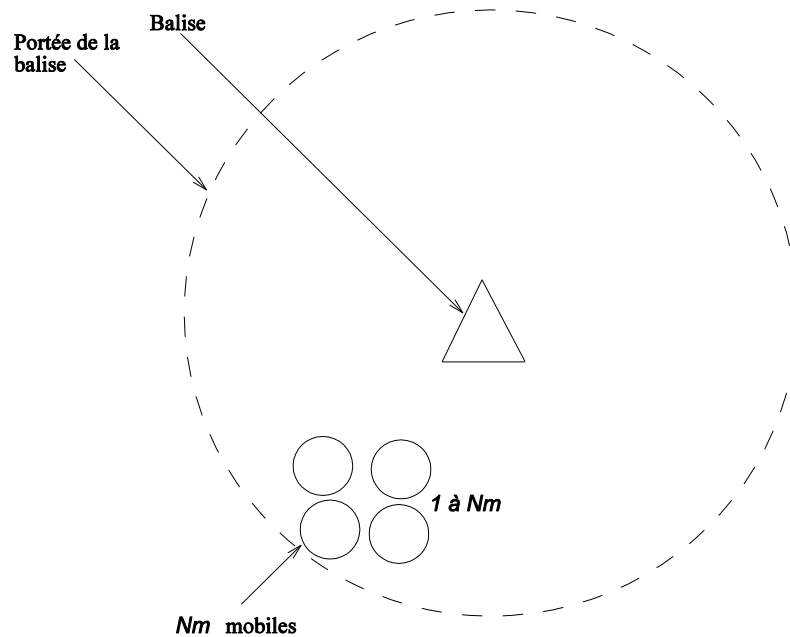
Le tableau 1 recense tous les paramètres de simulation pris en considération pour cette première évaluation.

|   |                                       |
|---|---------------------------------------|
| Simulateur  | NS2 [76] version 2.31                 |
| Protocole d'accès au médium                           | IEEE 802.15.4 CSMA/CA non slotté [34] |
| Type de trame de signalement                          | Trames diffusées                      |
| Débit   | 250 Kbps                              |
| Longueur des trames                                   | 15 octets                             |
| Nombre de mobiles en zone de couverture de la cellule | $N_m$ avec $N_m \in [1 \dots 200]$    |
| Fréquence de signalement                              | $1/T$ avec $T \in [0,1s \dots 2s]$    |
| Charge offerte à l'intérieur d'une cellule            | $N_m/T$ trames de signalement         |
| Durée de simulation                                   | 10 secondes                           |
| Modèle de propagation                                 | Free-Space                            |

**Tableau 1 : Paramètres de simulation pour la détection de proximité**

Tout au long de la simulation, tous les mobiles se signalent avec une fréquence égale à  $1/T$ . Chaque mobile utilise son horloge locale, il est tenu compte d'une possible durée d'horloge maximum de 30 ppm. L'activité de transmission des stations mobiles doit être non corrélée et quasi périodique. Pour ce faire, chaque nœud débute son activité de transmission de manière asynchrone en respectant un délai aléatoire choisi conformément à une loi uniforme dans un intervalle de temps  $[0, T [$ . Comme illustré dans la figure 66, le rôle de la balise dans la simulation est d'observer pour chaque mobile la

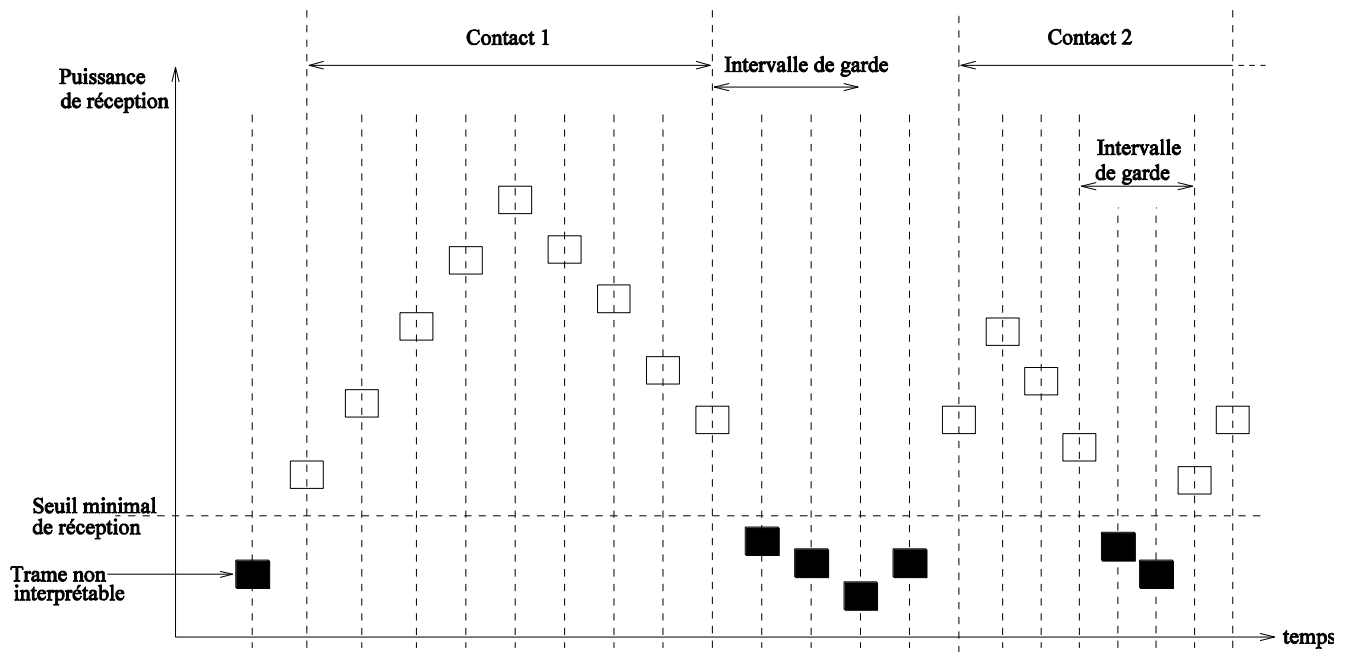
distribution des trames de signalement qu'elle reçoit pour étudier l'incidence de la charge (le nombre de mobiles dans la cellule) et de l'intervalle de garde sur le taux de fausses détections d'événements de contact. Le modèle de propagation Free-Space a été choisi pour ces premières simulations afin d'écartier les phénomènes aléatoire du modèle ITU qui sera repris dans la partie évaluation du mécanisme de colportage.



**Figure 66 : Configuration pour la détection de proximité.**

#### 4.3.1.2. Détection de proximité

La figure 67 rappelle le principe de détection de proximité qui est initiée par la bonne réception d'une trame qui déclenche le début d'un événement de contact [82]. L'intervalle de garde sur cette figure représente le temps durant lequel les trames peuvent être perdues sans provoquer la fin d'un événement de contact. Dans l'exemple donné dans la figure 67, l'intervalle de garde est fixé à trois périodes d'activité de signalement. En conséquence, la fin d'un contact dans cet exemple est déclenchée par la perte de trois trames consécutives (le mot perte signifie ici que ces trames ont été très probablement brouillées ou reçues avec une énergie inférieure à seuil minimal de réception qui permet leur bonne compréhension). Notons aussi que le second événement de contact n'est pas altéré par la perte de deux trames consécutives étant donné que l'intervalle de garde que nous avons choisi est suffisamment grand pour ne pas réagir à ce cas de figure.



**Figure 67 : Durée d'un événement de contact.**

Une raison expliquant aussi la perte de trames diffusées est la collision dont le risque est très lié à la surcharge du canal radio. On parle de collision lorsqu'un récepteur reçoit plusieurs trames simultanément émises par plusieurs émetteurs et que le signal issu de la superposition de ces trames ne permet de décoder aucune d'entre-elles. Si le récepteur est en mesure d'en comprendre une malgré tout, on parle alors de capture. Dans la figure 68, on peut constater que le premier événement de contact est terminé, car trois trames consécutives ne sont pas reçues. Un tel cas peut se produire lorsque trois trames ou plus ne sont pas reçues en raison de collisions ou si les trames ont été rejetées après plusieurs tentatives de transmissions échouées (ce qui est fréquemment observé avec la méthode CSMA/CA de la norme IEEE 802.15.4 lorsque la charge offerte augmente significativement).

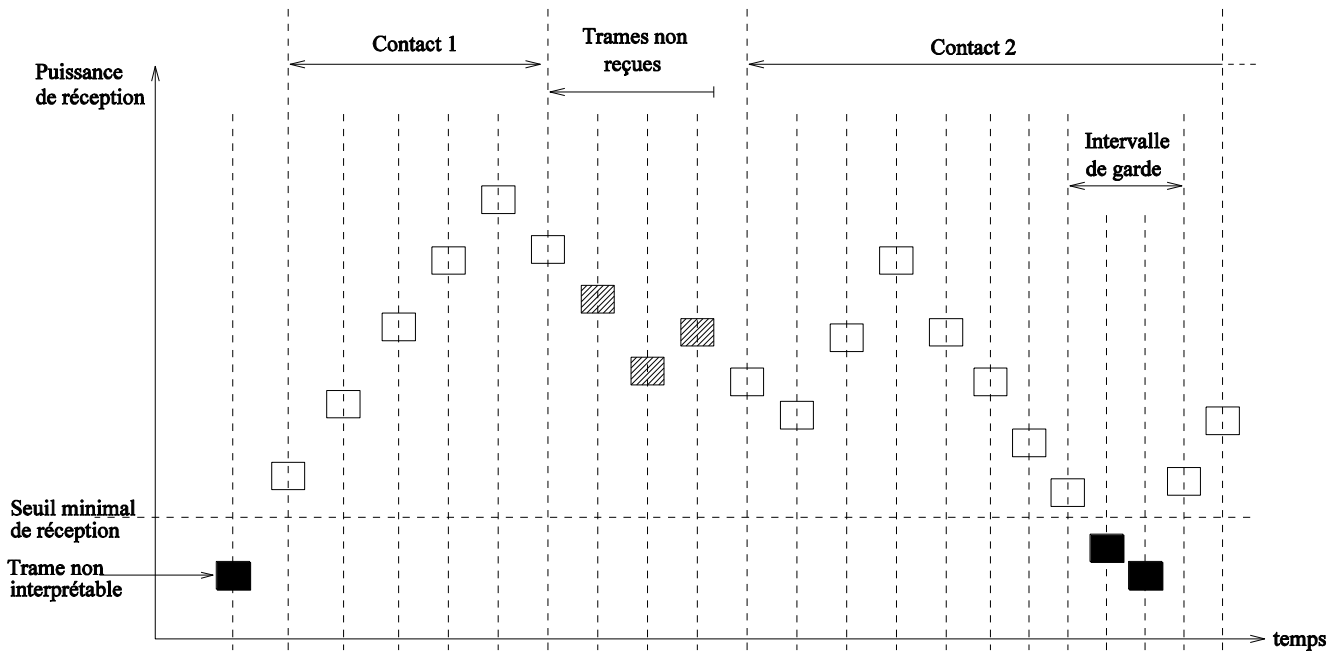


Figure 68 : Fausse détection de rupture de contact.

*a. Effets de la densité des mobiles dans une cellule*

La couverture cellulaire prise en considération est désignée par cellule de la balise dans ce qui suit. La première étude concerne la distribution des délais inter-trames pour l'ensemble des mobiles. La figure 69 donne la représentation d'une distribution de délais avec 5 mobiles.

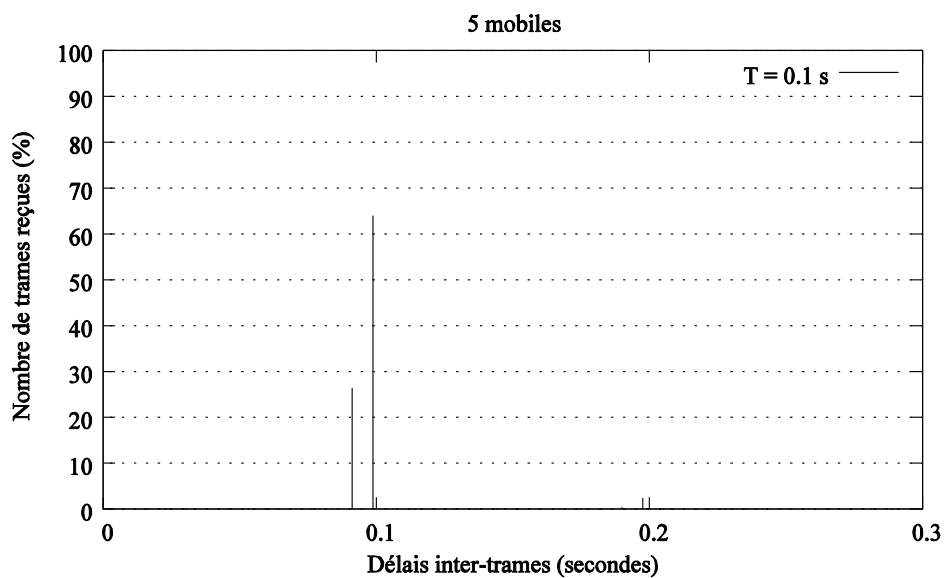


Figure 69 : Effets de la densité des mobiles avec 5 mobiles.

Lorsque le nombre de mobiles  $N_m$  est faible (5 mobiles), les délais inter-trames sont centrés autour d'une valeur de  $T = 0.1$  qui correspond à la fréquence de signalement des nœuds.

Lorsque le nombre de mobiles  $N_m$  augmente, deux phénomènes apparaissent : Le nombre de trames de signalement perdues augmente et la distribution montre un pic secondaire dû aux collisions des trames qui correspond à un délai d'inter-trame de  $2T$  lorsqu'une trame est perdue, et de  $3T$  lorsque deux trames successives sont perdues et ainsi de suite comme illustré dans la figure 70 pour 10 mobiles et la figure 71 pour 100 mobiles.

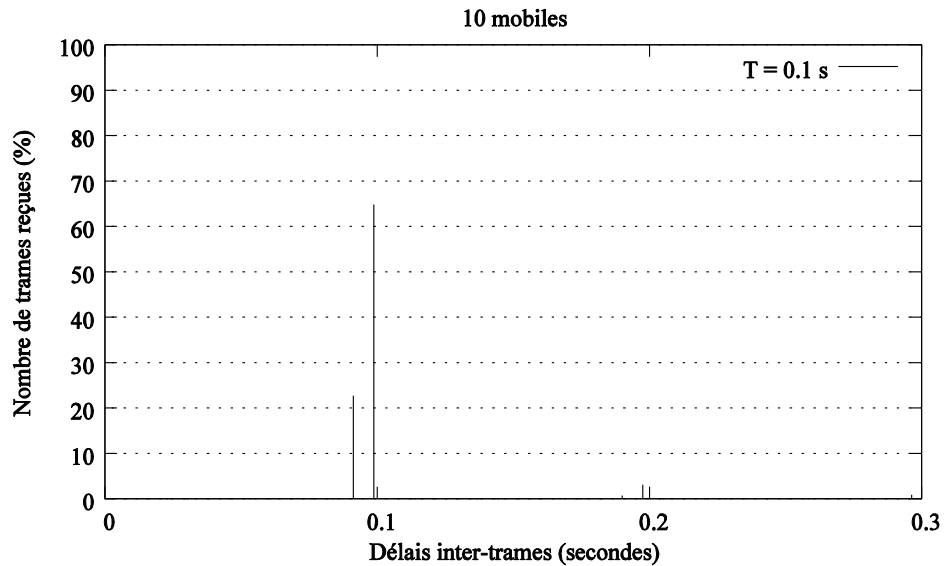


Figure 70 : Effets de la densité des mobiles avec 10 mobiles.

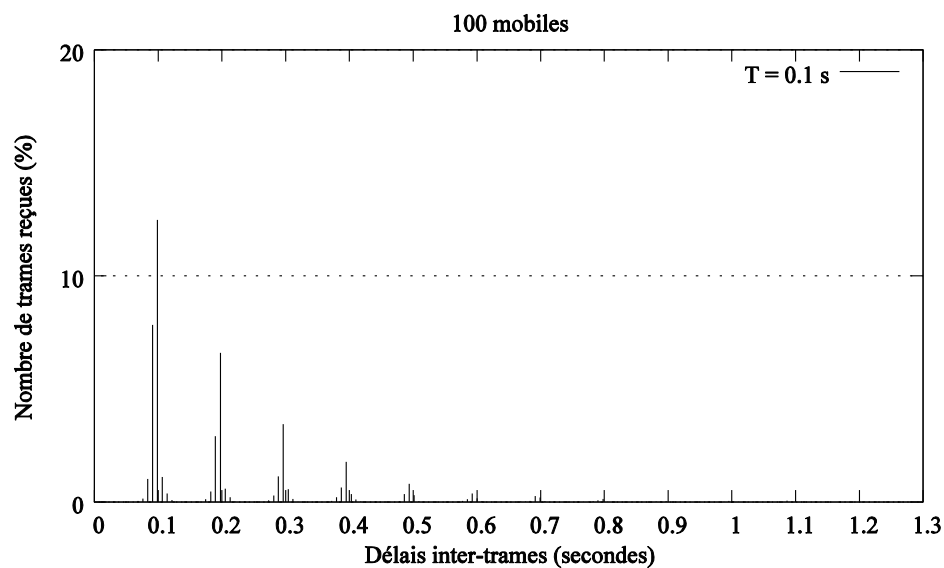


Figure 71 : Effets de la densité des mobiles avec 100 mobiles.

Dans le cas où  $N_m$  atteint une valeur de 200 stations (figure 72), on obtient une saturation de la cellule de la balise et les pics secondaires deviennent nombreux. Ceci signifie que lorsque les pertes de trames successives deviennent nombreuses, les pics ne sont plus interprétables et le nombre de trames perdus interrompt le processus d'identification des événements de contact.



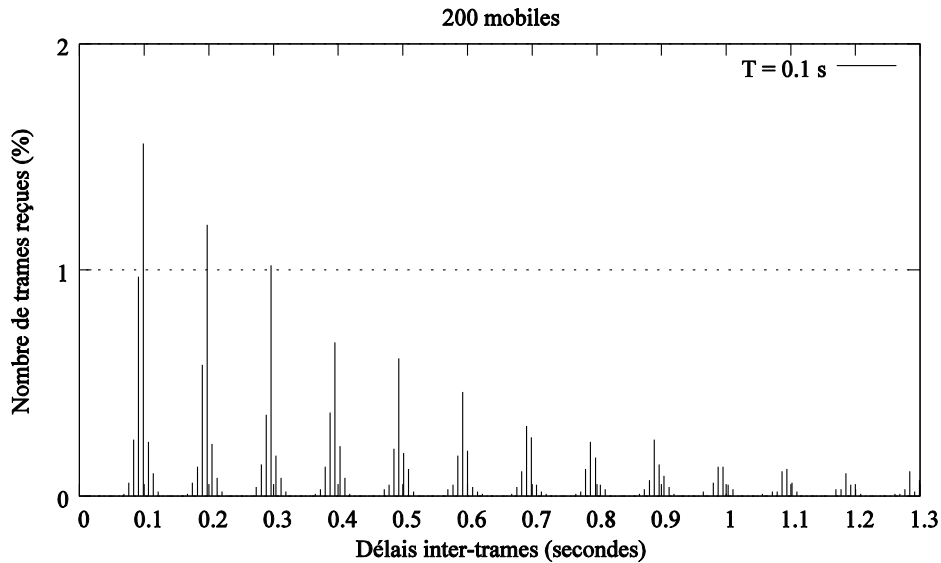


Figure 72 : Effets de la densité des mobiles avec 200 mobiles.

Prenons dans ce qui suit  $n_i$  comme étant la surface du pic  $i$  (somme des trames), si la surface du pic  $2T$  correspond à un nombre  $n_2$  de trames détectées avec un délai inter-trame de  $2T$ , ce qui induit que  $n_2$  trames ont été perdues. Le nombre de trames reçues  $n_2$  doit être associé à  $2*n_2$  trames transmises. Si  $n_3$  est le nombre de trames reçues correspondant à un délai d'inter-trame de  $3T$ , la surface du pic doit être associée dans ce cas à un nombre de trames transmises égal à  $3*n_3$  et ainsi de suite. Nous pouvons donc déduire :

$$\text{Nombre de trames estimées transmises} = \left( \sum_{i=1}^k i * n_i \right) + Nm$$

Où  $k$  : nombre de pics et  $Nm$  : le nombre de mobiles dans la cellule de la balise.

Le variable  $Nm$  représentant le nombre de stations mobiles est un facteur de correction introduit dans la formule pour tenir compte du fait que nous avons affaire à une distribution d'intervalles.

Cette formule est vérifiée lorsque la charge offerte est dans des conditions de surcharge dans la zone de couverture de la balise : le nombre de trames effectivement transmises est très proche du nombre de trames transmises estimées.

### ***b. Effets du nombre des mobiles pour une charge offerte donnée***

Nous allons évaluer dans ce qui suit l'effet du nombre de mobiles pour une charge offerte donnée. Cette dernière peut être obtenue comme suit :

- $Nm$  (100) stations mobiles avec une période d'activité de signalement égale à  $T$  (0.1 s).
- $2Nm$  (200) mobiles avec une période d'activité de signalement égale à  $2T$  (0.2 s).

Les résultats obtenus pour 100 mobiles et 200 mobiles sont donnés sous la forme logarithmique respectivement dans les figures 73 et 74.

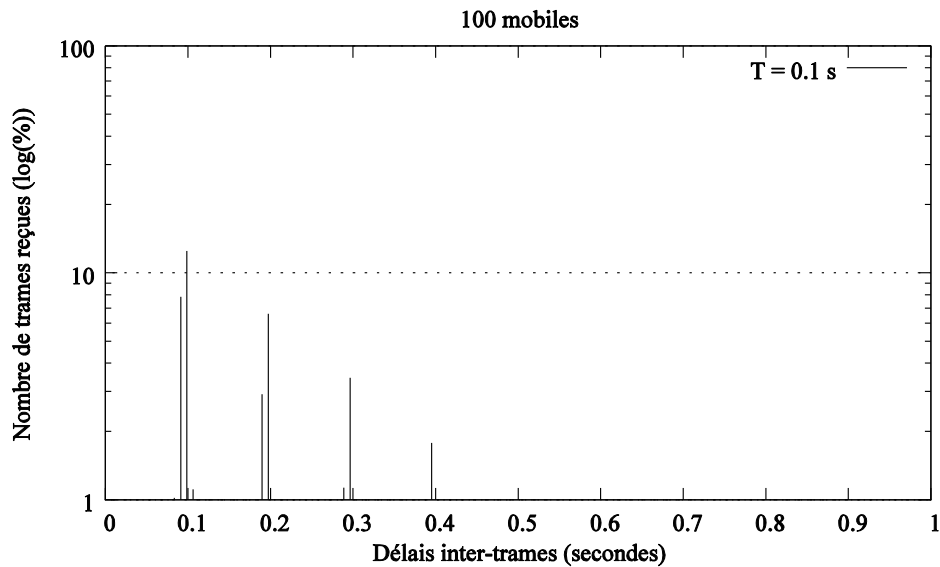


Figure 73 : Effets de 100 mobiles.

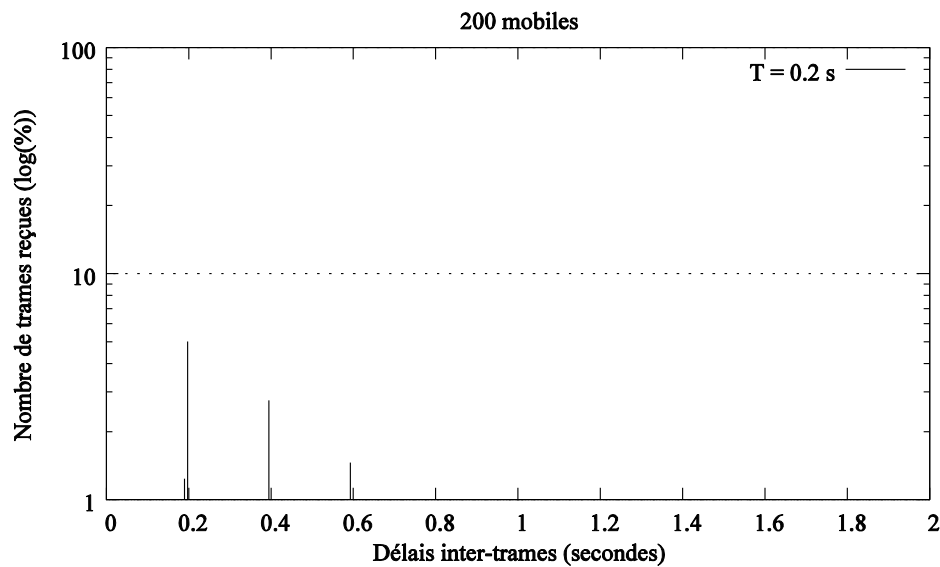


Figure 74 : Effets de 200 mobiles.

Nous pouvons noter que les formes des distributions des intervalles inter-trames sont assez proches. Néanmoins, on obtient une surface des pics beaucoup plus importante lorsque le nombre de mobiles impliqués dans la simulation est égal à 100 nœuds. En comparant les distributions pour un même trafic donné (charge offerte), produites par différents nombres de stations mobiles, nous constatons une saturation du réseau beaucoup plus rapide lorsque le nombre de stations est élevé [83].

### c. Effets de la charge offerte sur les fausses détections de rupture de contact

Pour les résultats donnés auparavant, la perte de trames résulte uniquement de la méthode d'accès au médium car tous les mobiles restent à portée de la balise durant toute la simulation. Même pour un petit nombre de mobiles, certaines trames sont perdues en raison des collisions. Ce qui explique l'apparition d'un second pic dans les figures précédentes (figures 69 à 74). C'est pour cette raison que l'intervalle de garde doit être calibré.

Dans nos simulations, bien que les entités sont toujours à portée radio l'une de l'autre, la couche physique et particulièrement la couche MAC vont générer des pertes de trames de signalement. Le choix de la valeur de l'intervalle de garde influe donc directement sur le nombre de fausses ruptures de contact. La figure 75 donne le pourcentage de fausses détections provoquées par la réception d'une trame en fonction de la longueur de l'intervalle de garde (donné en nombre de périodes de signalisation). La période de signalisation  $T$  est égale à 0.1 s et le nombre de stations mobiles varie de 5 à 100 mobiles.

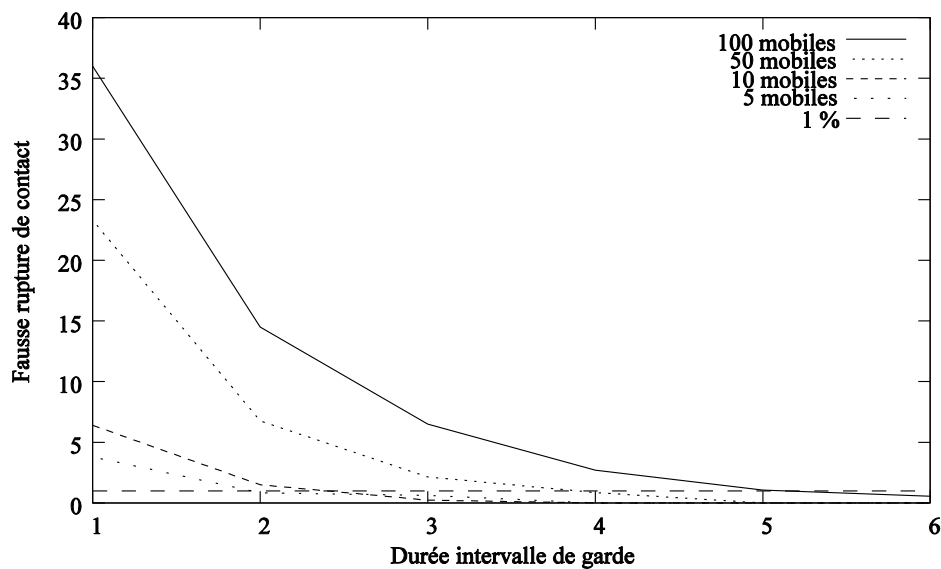


Figure 75 : Pourcentage des fausses détections de rupture de contact.

Cette courbe montre que le taux de fausses détections (qui multiplie donc le nombre d'événements de contact) peut être de plusieurs dizaines de pourcents quand la charge offerte (le trafic) à la cellule est trop grande. Nous pouvons néanmoins essayer de fixer une valeur de compromis pour l'intervalle de garde de façon à ce que le taux de fausses détections soit négligeable et que la fin d'un contact puisse être détectée rapidement. Si la charge offerte reste acceptable et si la longueur de l'intervalle de garde peut être ajustée à  $2T$  ou  $3T$ , alors le taux de fausses détections de rupture de contact est inférieur à 1% (pour  $N_m = 10$  mobiles et  $T = 0.1$  s).

### 4.3.1.3. Capacité d'une cellule

Dans cette section, nous allons étudier le comportement de la méthode CSMA/CA non slottée de 802.15.4 dans une cellule progressivement chargée par des trames diffusées. La même topologie illustrée dans la figure 66 est réutilisée; une balise définit une cellule de communication où  $N_m$  mobiles sont introduits progressivement, chacun d'eux offrant une charge de 10 trames par seconde. Le tableau 2 résume les paramètres de simulation pris en compte pour l'évaluation de la capacité d'une cellule.

|  |   |
|--|---|
| Simulateur   | NS2 [76] version 2.31   |
| Protocole d'accès au médium                          | IEEE 802.15.4 CSMA/CA non slotté [34]   |
| Type de trame de signalement                         | Trames diffusées  |
| Débit  | 250 Kbps  |
| Longueur des trames                                  | 15 octets   |
| Nombre de mobiles en zone de couverture d'une balise | $N_m$ avec $N_m \in [1 \dots 200]$  |
| Fréquence de signalement                             | $1/T$ avec $T \in [0,1s \dots 2s]$  |
| Charge offerte à l'intérieur d'une cellule           | $N_m/T$ trames de signalement   |
| Durée de simulation                                  | 10 secondes   |
| Dérive de l'horloge                                  | $T_i$ , avec $T_i = T [1 + \epsilon_i]$ et $\epsilon_i \in [-x, +x]$ ppm ( $x = 30$ $\mu$ sec dans les résultats qui vont suivre) |
| Modèle de propagation                                | Free-Space  |
| Nombre de répétition                                 | Moyenne sur 20 répétitions  |

**Tableau 2 : Paramètres de simulation pour l'évaluation de la capacité d'une cellule.**

Soit  $P$  la probabilité de manquer une trame et  $N_t$  le nombre de trames perdues consécutivement. Si par exemple,  $N_t$  est égal à 3 trois trames perdues consécutivement, la probabilité de fausse création d'événement de contact est de :  $P^{N_t} (1 - P^{n+1} / 1 - p)$  (pour prendre en considération les pertes de  $N_t, N_t+1, \dots, n$  trames consécutives). Cette formule sera approximée par  $P^{N_t}$  dans ce qui suit et va être utilisée afin de définir le nombre de mobiles pouvant se déplacer simultanément dans une cellule tout en gardant le risque de fausse création de contact sous un seuil donné.

Si on suppose à présent que la périodicité de diffusion des paquets est de 100 ms et que la durée pour qu'un mobile traverse toute la cellule d'une balise est de 100 secondes, durant ce temps, 1000 trames donc doivent être reçues.

Supposons maintenant que nous tolérons au plus qu'une seule fausse création d'événement de contact par nœud pendant qu'il traverse la zone de détection de la balise. Ceci peut être approximée par  $1000 * P^{N_t} \leq 1$ , c'est-à-dire  $P \leq 10\%$  pour  $N_t = 3$  (voir figure 75). Cela nous permet d'identifier une contrainte quant à la charge maximale supportée par une cellule.

### **a. Débit**

L'évaluation du débit est basée sur le calcul du trafic moyen reçu par chaque mobile. Ce débit est comparé ensuite à la charge offerte au sein de la cellule, c'est-à-dire :

*Charge de la cellule* =  $(N_m + 1) * 10$  trames/s, où  $N_m$  est le nombre de mobiles dans la cellule.

Ces simulations ont été réalisées pour différentes longueurs de trames : courtes (4 octets) correspondant à une trame de signalement, moyennes (60 octets) et longues (116 octets) correspondant à la trame de diffusion des événements de contact.

Nous avons aussi pris une valeur de  $P = 10\%$  dans le but d'identifier la capacité de la cellule. Notre objectif, dans ce qui va suivre, est d'avoir des événements de contacts pertinents au lieu d'avoir de faux contacts créés par erreur en raison de la surcharge de la cellule. En plus des trois modes de trafic homogènes (diffusion de trames courtes, moyennes et longues), un quatrième mode de transmission utilisant deux longueurs différentes a été analysé dont le fonctionnement est le suivant : une trame longue est diffusée suivie de deux trames courtes. Ce mode de transmission est appelé *transmission hybride*.

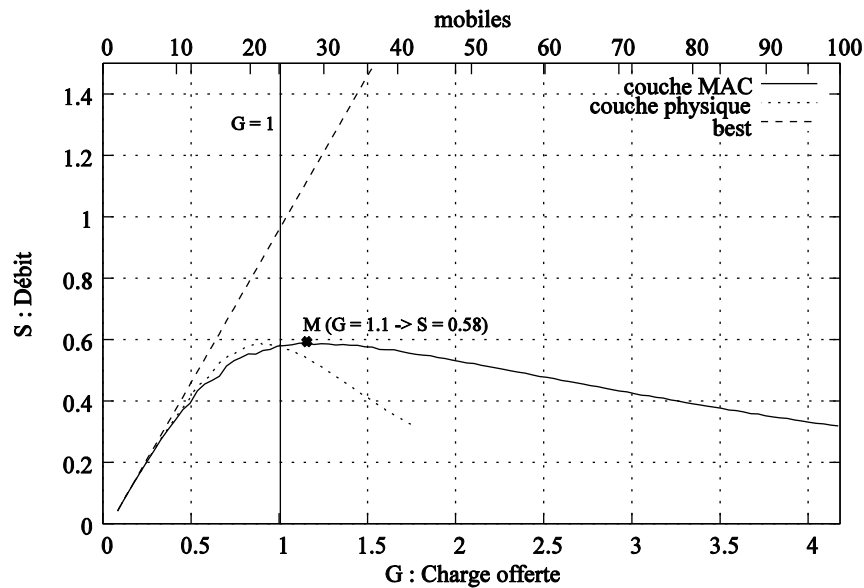
#### **Trafic homogène [84]**

L'étude du débit pour le trafic homogène a été réalisée pour les trois types de transmission définis auparavant. Les résultats obtenus sont donnés dans les figures 76, 77 et 78. Chaque figure est composée de trois courbes :

- Le trafic moyen reçu par un nœud (S) en fonction de la charge soumise (G) à la couche MAC (désigné par 'couche MAC'). Nous pouvons constater que la saturation du médium (145 kbps, 28 mobiles, point M) dans la figure 80 (trames longues) est atteinte pour une charge offerte supérieure à la capacité maximale du médium ( $G = 1$  pour 250 kbps). Ce paradoxe s'explique par le fait que la couche MAC absorbe une partie du trafic sans véritablement le soumettre au médium. Cela est dû au rôle de la couche MAC : un certain nombre de trames sont « dropées » par la couche MAC après plusieurs tentatives

consécutives d'envoi qui ont échouées.

- Le trafic moyen reçu par un nœud ( $S$ ) en fonction de la charge soumise ( $G$ ) à la couche physique (figure 76). Dans ce cas, la saturation du médium est obtenue avec une charge offerte plus petite que la capacité maximale ( $G \equiv 0.8$ ).
- La droite asymptotique représente le débit maximum théorique qui peut être obtenu ( $S = G$ ) appelée 'best'. Nous l'utiliserons par la suite pour calculer le nombre de collisions sur le médium.



**Figure 76 : Trafic moyen en fonction de la charge soumise : cas des trames longues.**

La même représentation pour des trames courtes est illustrée dans la figure 77. Notons que la saturation du médium est atteinte avec un débit de 62 kbps pour 96 mobiles ( $G = 0.5$ ).

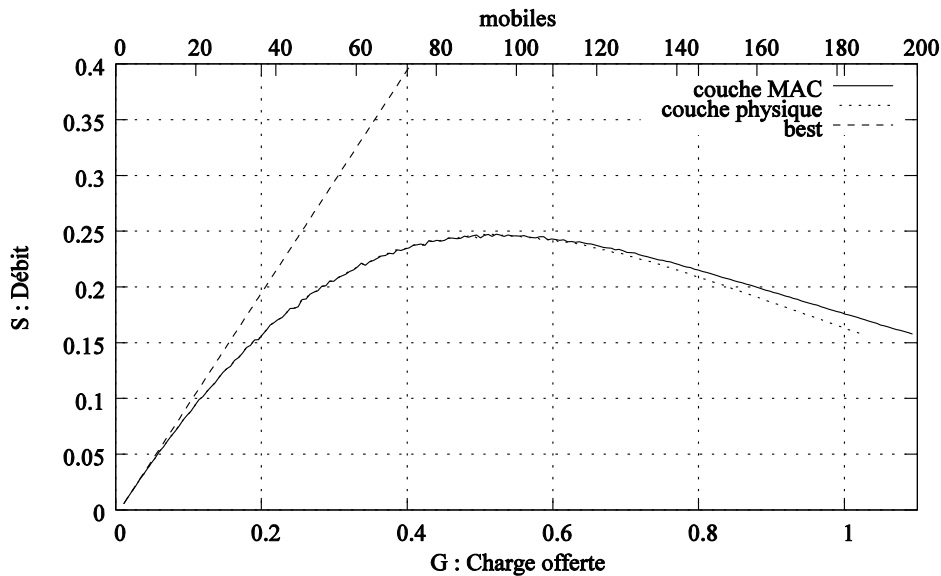


Figure 77 : Trafic moyen en fonction de la charge soumise : cas des trames courtes.

Enfin, la même approche a été utilisée pour une transmission avec des trames moyennes (60 octets) dans la figure 78. Le débit maximal est obtenu avec 125 kbps pour 42 stations mobiles ( $G = 0.9$ ).

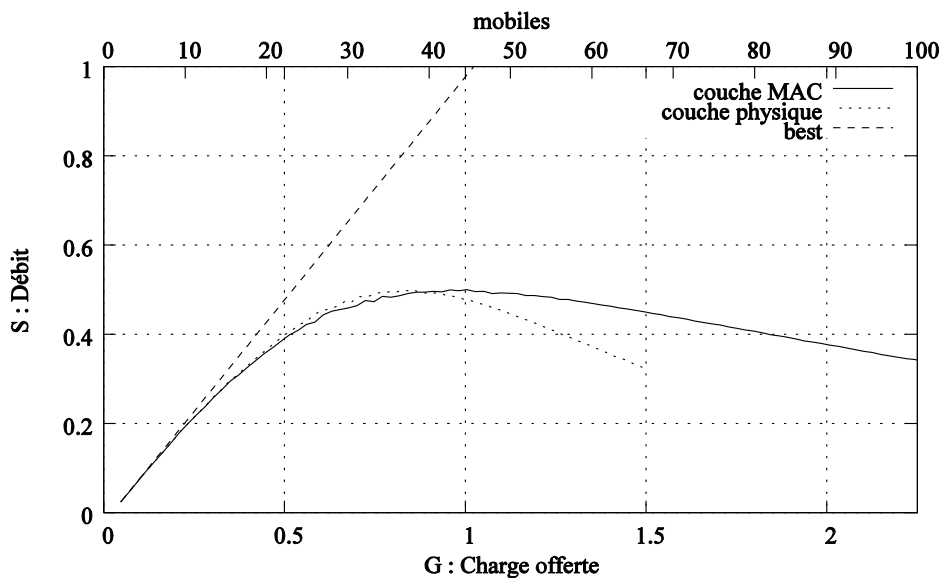


Figure 78 : Trafic moyen en fonction de la charge soumise : cas des trames moyennes.

### Trafic hétérogène

Nous représentons maintenant le débit des transmissions hétérogènes en utilisant le mode hybride qui bascule entre trames longues et courtes (une trame longue pour deux trames courtes). Ce type de transmission tente de reproduire le cas où plusieurs mobiles sont à portée, mais seulement une partie d'entre eux échange des événements de contact à l'aide de trames longues. La figure 79 montre l'impact d'une transmission hétérogène. Un débit maximal de 135 kbps est obtenu lorsque 29 mobiles

échantent des événements de contact dans des trames longues tandis que 58 nœuds mobiles transmettent des trames courtes.

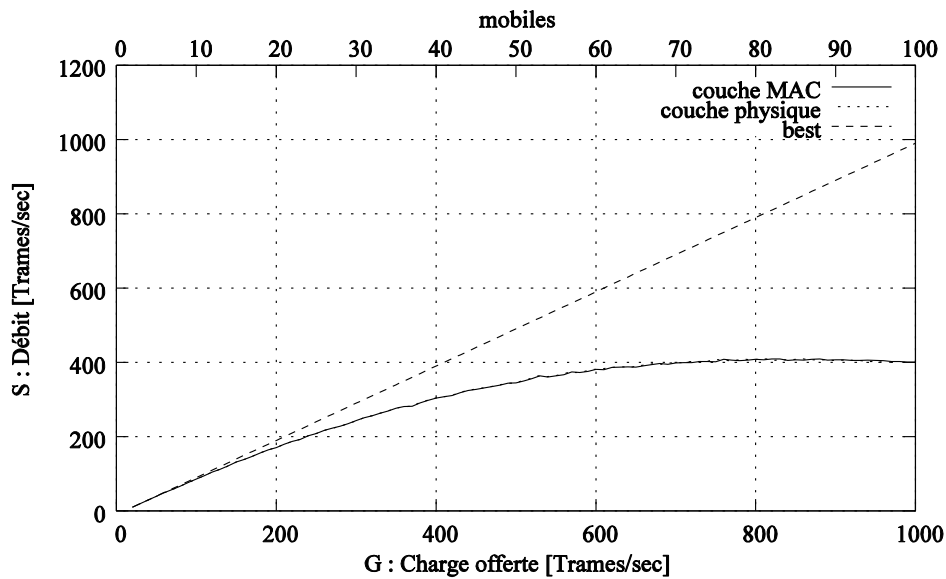


Figure 79 : Trafic moyen en fonction de la charge soumise : cas d'une transmission hybride.

### b. Etude de la cause de pertes de trames

Dans ce qui suit, nous allons montrer que nous pouvons déduire à partir des simulations précédentes (débit vs charge offerte au niveau de la couche physique et de la couche MAC) le nombre de drops au niveau de la couche MAC et le nombre de collisions des trames diffusées.

#### Nombre de drops

Afin de calculer le nombre de drops, reprenons les résultats présentés dans la figure 80 qui illustre le débit obtenu pour des trames longues ( $S = f(G)$ ). De chaque point A de la courbe 'couche MAC', une charge offerte  $X_a$  (sur l'axe des x) et un débit  $S_p$  (sur l'axe des y) peuvent être déduits. Ce débit particulier sera utilisé pour définir trois autres points comme le montre la figure 84. Soit B ( $X_b, Y_p$ ) le point correspondant de la courbe 'couche physique'. Il est également possible de définir un point C ( $X_b, X_b$ ) et un point D ( $X_b, Y_p$ ) de la droite 'best', qui correspond à  $S = G$ . Ces points seront utilisés dans ce qui va suivre pour évaluer la cause des pertes de trames.



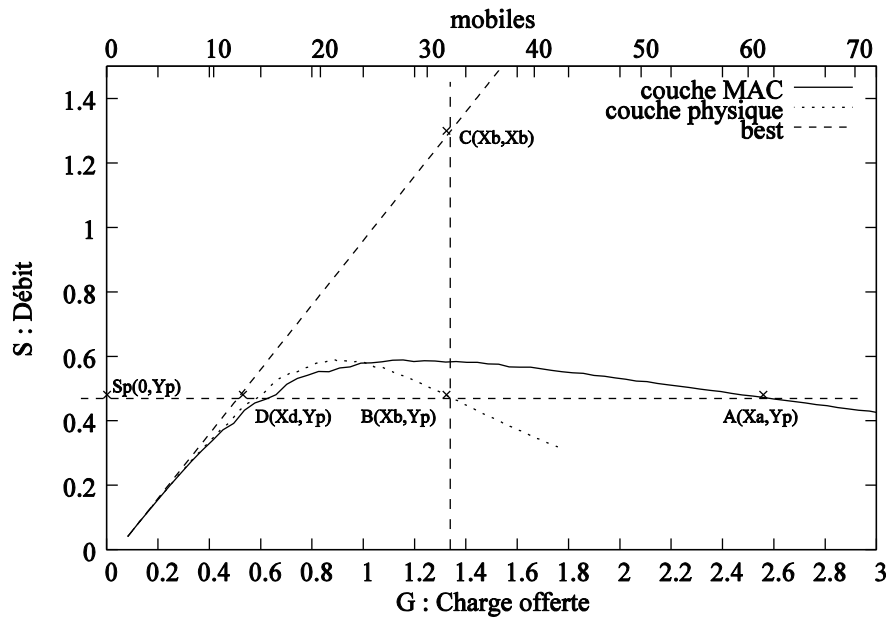


Figure 80 : Trafic moyen pour des trames longues.

Pour chaque point A, il est possible d'associer un point B. La différence entre ces deux points sur l'axe des x donne le nombre de trames perdues qui ont été mises en attente au niveau de la couche MAC mais qui n'ont finalement pas été émises. Soit  $F_1$  la fonction donnant le nombre de drops :

$$F_1(X_a) = X_a - X_b. (1)$$

La figure 81 donne le nombre de drops au niveau de la couche MAC pour les quatre types de transmissions. Notons que les résultats obtenus dépendent de la longueur des trames diffusées, et que le plus grand nombre de drops est obtenu avec une transmission de trames longues. Ce nombre par contre, est très faible lorsqu'il s'agit d'une transmission hybride.

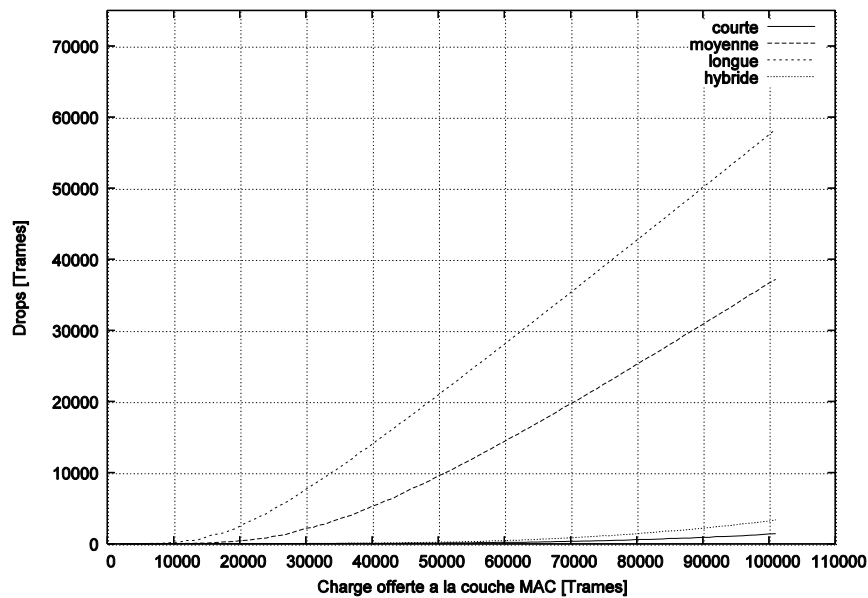


Figure 81 : Influence de la longueur des trames sur le nombre de drops.

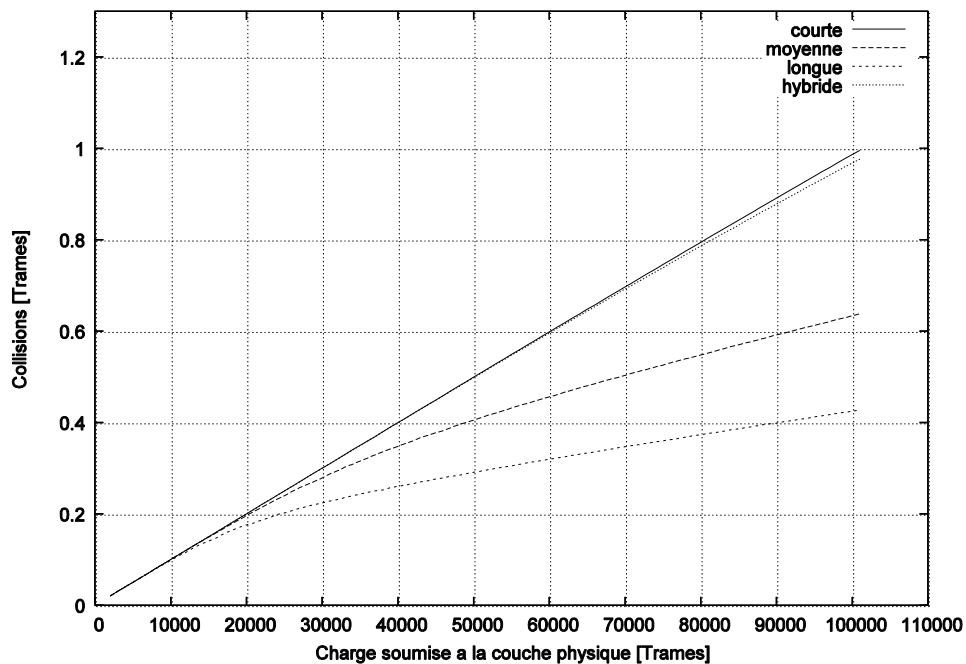
## Nombre de collisions

Dans nos simulations, nous avons utilisé un modèle de propagation de type Free-space, et tous les mobiles dans le scénario sont à portée les uns des autres. Nous supposons que les trames gérées et diffusées par la couche physique sont soit reçues ou bien affectées par des collisions.

Reprenons les points définis précédemment :  $B(X_b, Y_p)$  représente une charge offerte au médium qui donne  $S_p$  comme débit. La courbe notée 'best' donne le débit dans le cas où toutes les trames transmises sont correctement reçues. Le nombre de collision est obtenu par la différence entre ce qui est soumis au médium et ce qui est reçu (c'est à dire, le segment [DB]).

$$F_2(X_b) = X_b - X_d \quad (2)$$

La figure 82 illustre le nombre de collisions obtenu pour des trames courtes, moyennes, longues et pour une transmission hybride. Notons que le nombre de collisions augmente quasi proportionnellement avec le nombre de nœuds dans le réseau pour les quatre types de transmissions.



**Figure 82 : Nombre de collision en fonction de la charge offerte soumise à la couche physique.**

Pour avoir une meilleure vision des effets combinés des collisions et des drops, il est intéressant d'étudier le rapport entre les trames perdues et la charge offerte à la couche MAC. Ce taux peut être calculé par la formule :

$$R = (N_p x - N_r x - t/T) / N_p x \quad (3)$$

Où :

- $N_x$  : est le nombre de trames transmises sur le médium.
- $N_r$  : est le nombre moyen de trames reçues.
- $T$  : est la période de signalisation.
- $t$  : est le temps passé dans la zone de couverture de la balise.

Le facteur correctif ( $t/T$ ) dans (3) est utilisé pour prendre en compte le fait qu'un nœud n'est pas en mesure d'écouter ce qu'il est en train de transmettre.  $R$  est représenté dans la figure 83.

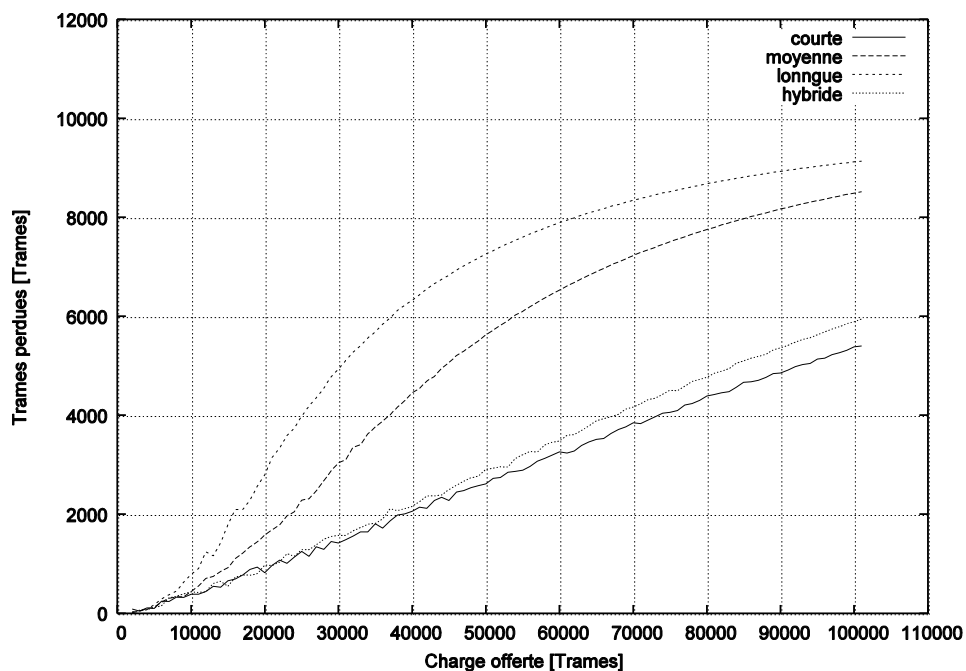


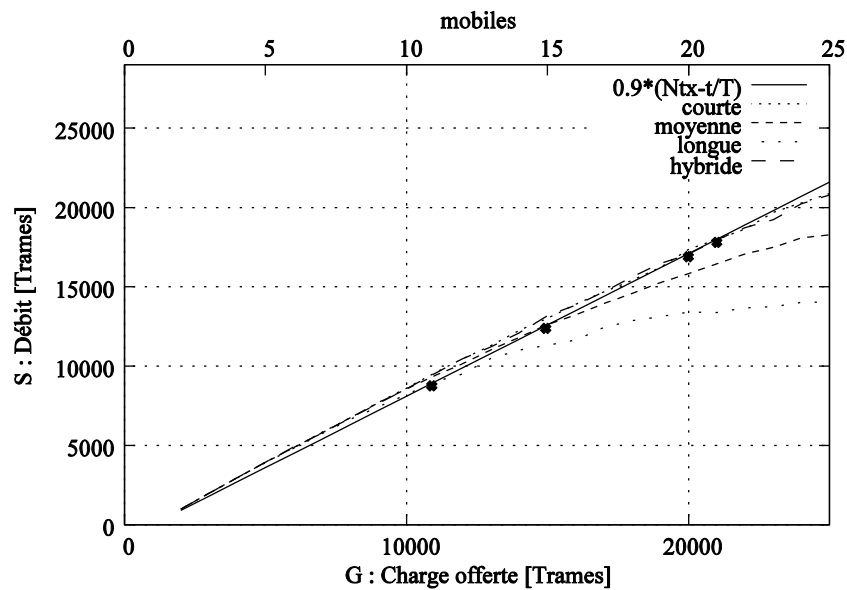
Figure 83 : Nombre de trames perdues en fonction de la charge offerte.

### c. Nombre maximum de mobiles dans une cellule

Nous nous intéressons maintenant au nombre maximum  $N_m$  de mobiles qu'une cellule peut accepter avant la création de faux événements de contact. Notre hypothèse est que nous tolérons au plus une création d'un faux événement de contact lors des 100 secondes de simulation (pour une fréquence de signalement de 10 trames par seconde nous avons vu que cela nous conduisait à une probabilité de perte de  $P \leq 10\%$ ). Pour identifier la valeur de  $N_m$  pour une longueur de trame donnée, nous considérons l'intersection entre la courbe donnant le débit et la ligne droite  $ET(x)$  représentant ce que nous exceptons : plus de 90% de trames sont reçues.

$ET(x) = 0.9 (N_x - t/T)$  (4) où  $ET$  représente le trafic minimum attendu.

La figure 84 donne l'évolution de ce ratio en fonction des trames diffusées. Nous pouvons nous apercevoir que le seuil de 10% est rapidement atteint.



**Figure 84 : Evaluation de la charge d'une cellule.**

La figure 84 illustre comment la valeur de  $N_m$  a été calculée selon les différentes tailles de trames. Nous déduisons de cette figure les valeurs données dans le tableau 3.

|                      |            |
|----------------------|------------|
| Trames longues       | 11 mobiles |
| Trames moyennes      | 15 mobiles |
| Trames courtes       | 21 mobiles |
| Transmission hybride | 20 mobiles |

**Tableau 3 : Nombre maximum de mobiles dans une cellule.**

Dans cette première partie de simulation, nous nous sommes focalisés sur le comportement de la méthode d'accès CSMA/CA de la norme 802.15.4 dans son mode sans beacon en surchargeant une cellule de référence progressivement. Les résultats obtenus par le simulateur NS2 concernent plusieurs caractéristiques de ce comportement : le débit, les drops au niveau de la couche MAC, les collisions, le taux de perte...etc. Le but principal de cette étude est de limiter la création de faux événements de contact produits principalement par les pertes de trames successives. Les résultats de simulations nous ont aussi permis de déterminer le nombre maximal de stations mobiles qu'une cellule peut supporter pour un nombre de faux événements de contact acceptable.

Après avoir étudié les limites de la détection de proximité dans une cellule donnée, nous allons maintenant aborder comment l'information (les événements de contact) circule dans ce réseau tolérant au temps pour atteindre un point de collecte. Pour ce faire nous allons considérer les effets du

mécanisme consistant à avantager ou à purger du réseau certains des événements de contact faisant partie de la connaissance des nœuds.

### 4.3.2. Evaluation du mécanisme de routage

Cette section du chapitre décrit deux représentations graphiques de deux sites qui seront réutilisées dans la suite de ce chapitre afin d'évaluer le mécanisme de colportage lors d'une phase de maquettage et/ou de simulation.

#### 4.3.2.1. Topologie et conditions de simulation

##### Représentation d'une galerie minière

Dans la première partie de ce chapitre où nous avons introduit les différentes entités du réseau impliquées dans le processus de colportage, nous avons donné une première représentation simpliste d'une mine. A partir de cette première approche, nous proposons un schéma plus détaillé basé sur des cartographies de mine réelles. Evidemment, pour des raisons de manque de place, ce schéma ne reprend qu'une partie d'une mine. Cette représentation est donnée dans la figure 85.

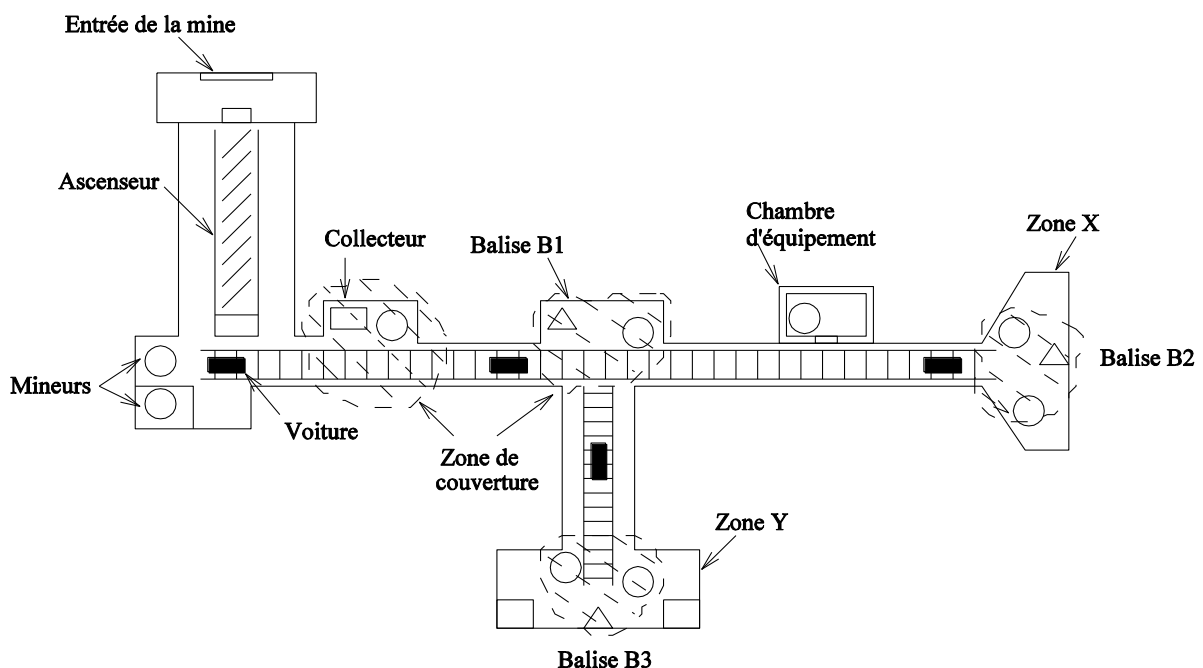


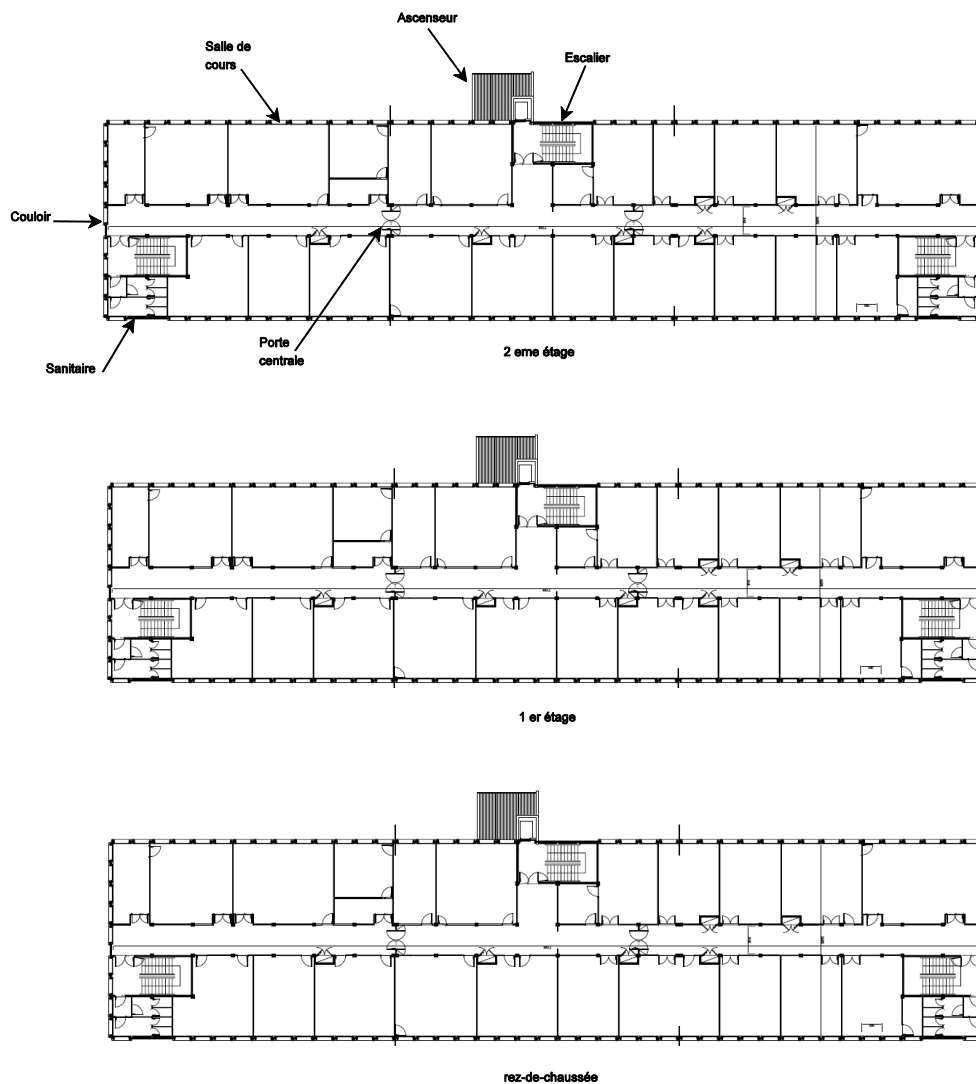
Figure 85 : Représentation réaliste d'une partie d'une mine.

On retrouve sur cette figure les mêmes acteurs définis précédemment, c'est-à-dire : le collecteur qui est placé à l'entrée de la mine qui sert aussi de sortie, les balises B1, B2 et B3 qui sont disposées dans différents points de la galerie et les deux zones d'activités (Zone X et Zone Y) qui sont localisées aux extrémités de la galerie. Enfin, les entités mobiles représentant le personnel de la mine suivent au

cours de la simulation des déplacements définis selon différents scénarii, certaines vont passer devant le collecteur lors de leur sortie de la mine.

### ***Représentation des couloirs d'un IUT***

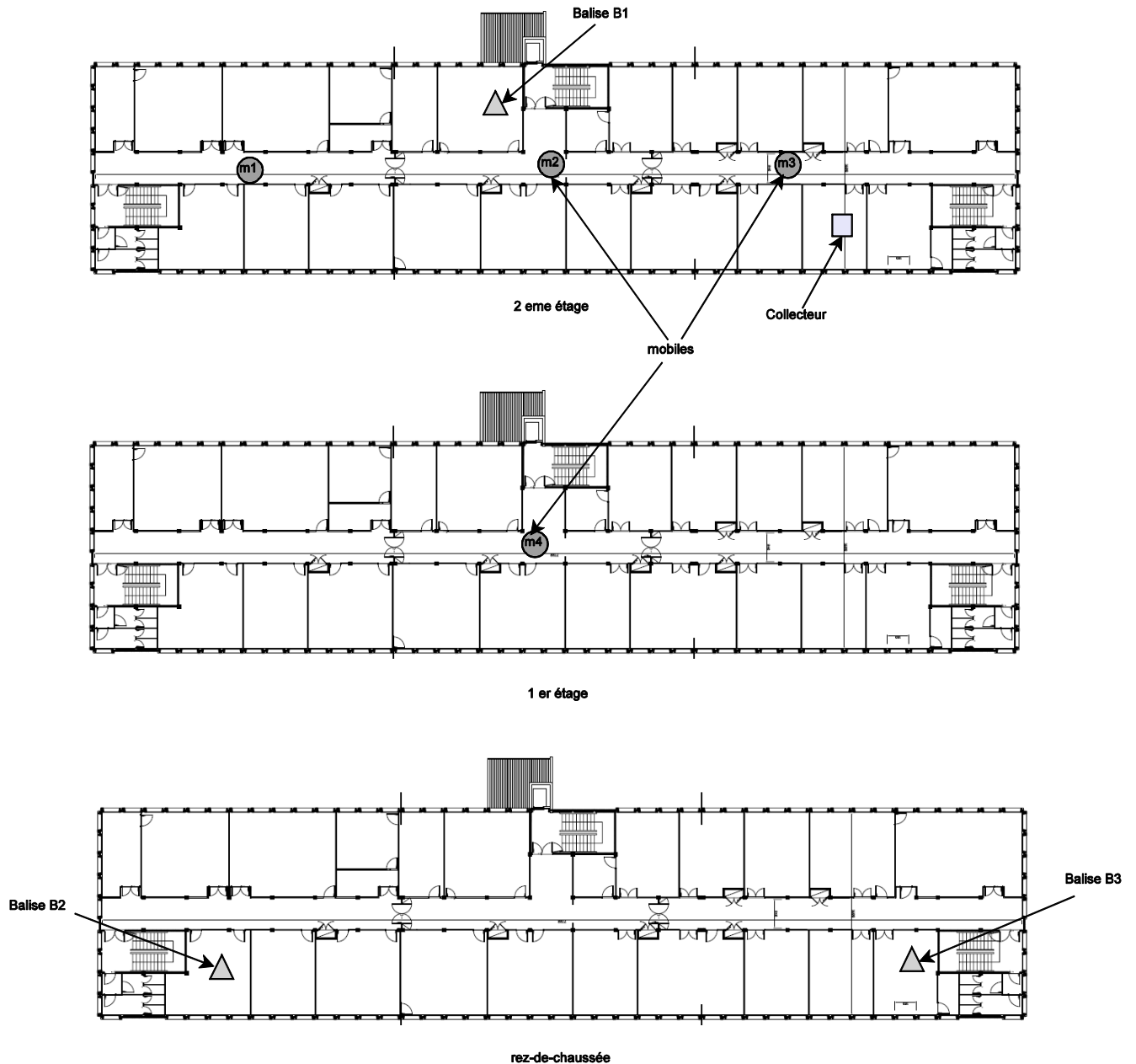
Dans la partie expérimentation de ce mémoire, tous les tests ont eu lieu sur trois étages d'un bâtiment de l'IUT d'Aubière (Institut Universitaire de Technologie de Clermont-Ferrand). Nous avons choisi ce type d'infrastructure d'intérieur afin d'approcher simplement les conditions de propagation des milieux confinés comme celles relatives à une galerie de mine par exemple [85]. La représentation donnée dans la figure 86 schématise le plan exact du couloir du rez-de-chaussée, du premier et du second étage de ce bâtiment.



**Figure 86 : Plan des trois étages du bâtiment INFO-R&T de l'IUT de Clermont-Ferrand.**

Un couloir est constitué de plusieurs salles de cours et salles de professeurs ainsi que deux salles sanitaires situées aux extrémités du couloir. Deux grandes portes séparent le couloir en trois parties et

trois escaliers sont disponibles à chaque étage, un à chaque extrémité et un troisième au milieu du couloir. Enfin, un ascenseur est placé au milieu du bâtiment pour donner accès aux autres étages. La figure 87 reprend la même représentation des étages en indiquant les emplacements des différents nœuds du réseau impliqués dans le colportage.



**Figure 87 : Déploiement des nœuds dans le couloir d'un bâtiment.**

Le premier nœud déployé dans le réseau est le collecteur qui est placé tout à droite de la figure au second étage dans une salle des professeurs. Le second nœud (Balise B1) est positionné au même étage, juste au milieu du couloir à côté de l'ascenseur. La seconde balise B2 se place à gauche du couloir du rez-de-chaussée et la troisième balise se trouve à droite de ce même étage. Enfin, les nœuds

mobiles employés dans le réseau sont portés par des étudiants en plein mouvement entre les différentes salles de cours.

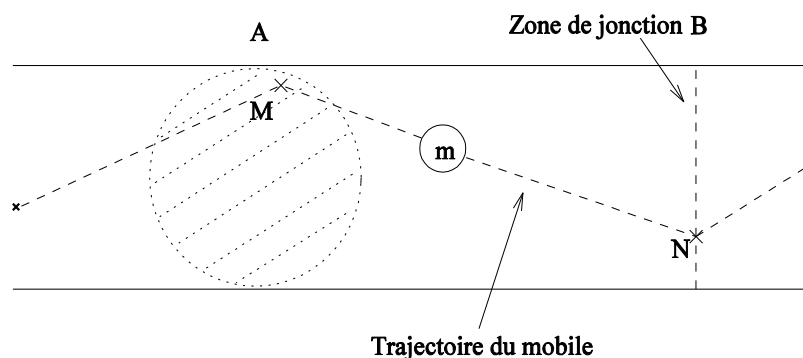
Cette topologie va être réutilisée par la suite dans la partie expérimentation dans le but de valider la faisabilité de notre approche de routage en comparant les résultats obtenus sur simulateur NS2 et ceux obtenus quand les nœuds capteurs sont des cartes TelosB [86].

### ***Modélisation de la mobilité***

Les performances des réseaux DTN en matière de débit et de délais dépendent fortement du facteur de mobilité des nœuds. Etant donné l'importance cruciale de ce paramètre, nous proposons dans cette partie de ce chapitre un modèle de mobilité pour des applications industrielles dans des milieux d'intérieurs ou confinés tels qu'une mine.

La mine présentée dans la première partie de ce chapitre (figure 85) est décrite comme étant un polygone en 2D où les sommets et les différents segments représentent les parois ou les murs de la galerie. Chaque nœud mobile est associé à un mineur en activité et suit un scénario qui dépend du type de la tâche qui lui est attribuée. Un tel nœud peut donc décider de rejoindre les zones de travail (Zone X ou Y, Figure 85) de façon aléatoire au début de son activité.

Tout au long de leur trajet, les mobiles se déplacent sur des segments qui sont définis comme étant la distance entre deux jonctions successives. Une jonction peut être soit un point d'un segment (orthogonal aux parois) ou de la surface d'un cercle (inscrit dans la galerie). Dans la figure 88, le pont M est choisi aléatoirement parmi les points du cercle A, tandis que le point N est choisi aléatoirement sur le segment B.

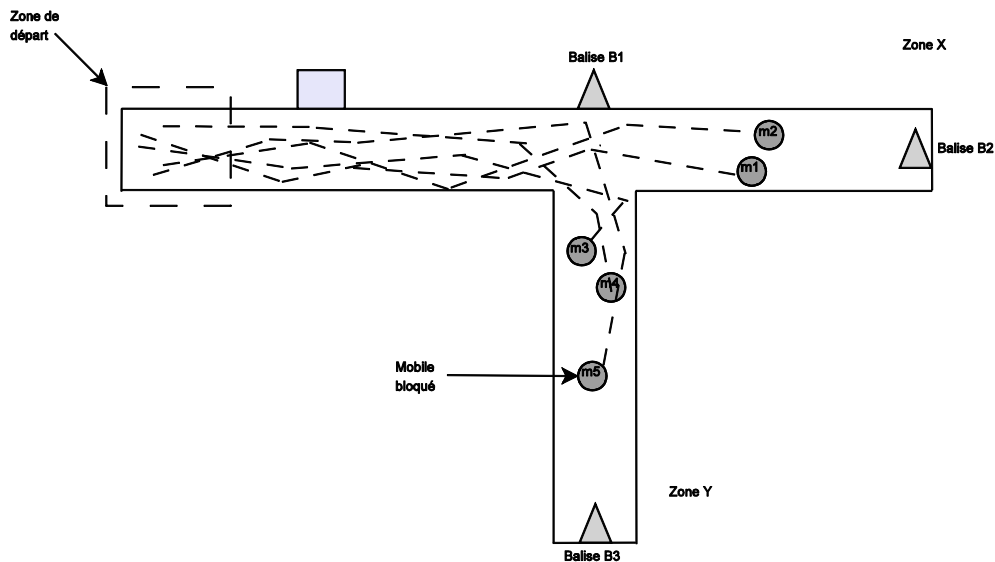


**Figure 88 : Trajectoire d'un mobile entre deux jonctions.**

Un autre paramètre pris en compte dans la mobilité est la vitesse de déplacement qui est choisie aléatoirement entre [0 et  $V_{max}$ ] et affectée à chaque segment parcouru. De plus, au franchissement de



chaque jonction les nœuds mobiles choisissent aussi un temps d'attente aléatoire choisi dans  $[0, T_{max}]$  et représentant une pause ou une petite tâche d'un ouvrier. A l'expiration de cette pause à la jonction, chaque nœud décide du prochain point destination à atteindre (à la jonction suivante). Pour avoir un aperçu de ce modèle de déplacement, la figure 89 reprend le schéma de la galerie minière en ajoutant la trajectoire de quatre mobiles effectuant des tâches en zone X et Y ainsi qu'un cinquième mobile bloqué dans la mine.



**Figure 89 : Représentation d'une galerie minière avec trajectoires de nœuds mobiles.**

Cette approche de modélisation basée sur les travaux de [54] et [55] permet d'obtenir des déplacements plus au moins aléatoires en introduisant plusieurs paramètres tels que la vitesse, la direction et les temps d'attente et en contrôlant les valeurs de  $V_{max}$ ,  $T_{max}$  et la position et la taille des jonctions que nous venons d'introduire. Cette façon de contraindre ces facteurs aléatoires, permet à la mobilité d'être plus ou moins prévisible étant donné qu'en respectant les trajectoires des tâches à accomplir en zone d'activité, les mobiles utilisent des chemins semblables ce qui peut être enrichissant pour la collecte de données.

Cette section est consacrée à l'évaluation de notre mécanisme de routage dédié à l'application de localisation LoWCA. Selon le type d'applications visées, il est nécessaire de définir quel type d'information (type d'événements de contact) il est nécessaire d'échanger lors d'un contact pour éviter une surcharge du système. En effet, chaque nœud dans le réseau maintient une connaissance locale d'événements de contact créés ou reçus de ses partenaires avec qui il est ou a été en contact. Une fois que la mobilité devient effective, notre système risque d'avoir à faire face à une forme d'explosion combinatoire. Comme dans la plupart des protocoles de diffusion [43], un tel mécanisme implique forcément des doublons qui peuvent provoquer une surcharge de la capacité mémoire des nœuds voire du médium lorsque plusieurs stations se retrouvent à portée. Il est donc nécessaire de faire un choix

sur l'information (événements de contact) à conserver localement. Pour pallier ces risques de saturation de mémoire et du médium, des règles sont appliquées par les nœuds lorsqu'ils reçoivent des copies d'événements de contact : des filtres sont appliqués dans ce cas pour décider quels événements doivent être conservés en mémoire pour être diffusés ultérieurement à d'autres partenaires. Le but de ces règles de filtrage est de collecter une grande diversité d'événements de contact et non pas un nombre important de répliqués de ses événements. Dans le chapitre précédent, nous avons dressé une liste de filtres impliqués dans le processus de colportage. Nous avons choisi d'étudier en premier les filtres permanents : doublons (DOF), dual (DUF) et concaténation (COF). Ensuite, pour les besoins de l'application ciblée LoWCA, nous aborderons l'évaluation du filtre pour la localisation qui avantage les rencontres « balise-mobile » appelé TNAF. Ce filtre sera comparé aux premiers filtres permanents ainsi qu'au filtre aléatoire (PRF : Filtre aléatoire avec probabilité  $P$ ) qui supprime les événements en mémoire de manière aléatoire selon une probabilité  $P$ . Un bref rappel de spécifications de ces filtres est résumé dans le tableau 4.

|   |   |
|---|---|
| Filtre doublons (DOF)                       | Si $CEa$ et $CEb$ doivent être stockés en mémoire d'un nœud $NI$ , avec le même couple d'identifiants $(Idx, Idy)$ et le même numéro de série $NS$ , ils sont considérés comme doublons et seul un est gardé en mémoire.  |
| Filtre dual (DUF)                           | Si $CEa$ et $CEb$ doivent être stockés en mémoire du nœud $NI$ avec le même couple d'identifiants $(Idx, Idy)$ et $([T1, T2] \equiv [T'1, T'2])$ alors garder en mémoire le $CEa$ si $Idx = NI$ et $CEb$ si $Idy = NI$ . Sinon, celui garder celui qui est déjà en mémoire. |
| Filtre concaténation (COF)                  | Si $CEa$ et $CEb$ doivent être stockés en mémoire du nœud $NI$ avec le même couple d'identifiants $(Idx, Idy)$ et $(T2 \equiv T'1)$ alors garder le contact $CEa$ : $Idx, Idy, NSa, T1, T'2$ .  |
| Filtre aléatoire avec probabilité $P$ (PRF) | Libérer une place mémoire aléatoirement avec une probabilité $P$ .  |
| Filtre avantage balise (TNAF)               | Supprimer une place de la mémoire du nœud $NI$ qui ne concerne pas une balise. Si aucun événement de ce type n'est trouvé en mémoire, appliquer alors le filtre aléatoire.  |

**Tableau 4 : Liste des filtres avec  $CEa : Idx, Idy, NSa, T1, T2$  et  $CEb : Idy, Idx, NSb, T'1, T'2$ .**

Nous examinerons à travers des simulations et expérimentations les effets de la mise en œuvre de ses filtres sur la pertinence et la précision des informations diffusées par les mobiles et collectées au niveau du collecteur.

#### 4.3.2.2. Effets des filtres

Les résultats présentés dans cette section illustrent l'effet des filtres sur la pertinence de l'information colportée par les stations mobiles jusqu'au point de collecte comparés au nombre d'événements de contact créés dans le processus de simulation par les nœuds fixes et mobiles (notée dans la suite par *connaissance absolue*) [70]. Nous utilisons aussi le concept de *connaissance utile* notée  $G$ , qui consiste à calculer la quantité d'informations que les nœuds mobiles soumettent au médium. Dans chaque figure qui montre l'effet des filtres, le nombre de contact *absolu* produit par le processus de simulation a été ajouté. Ce nombre augmente avec le nombre de nœuds mobiles dans le réseau et dépend des déplacements des nœuds mobiles. La courbe en tiret indique le nombre de contacts utiles diffusés par les mobiles tout au long de leur chemin et la courbe en pointillée représente le nombre de contacts collectés au niveau du collecteur.

La métrique adoptée dans cette partie est basée sur le comptage des événements de contact déduits. Pour déduire un contact, il suffit d'avoir au moins une de ses quatre versions qui sont les suivantes :

- $CE1 : N_x, N_y, NS1, T1, 0$  ou son dual  $CE2 : N_y, N_x, NS2, T'1, 0$ .
- $CE3 : N_x, N_y, NS3, T1, T2$  ou son dual  $CE4 : N_y, N_x, NS4, T'1, T'2$ .

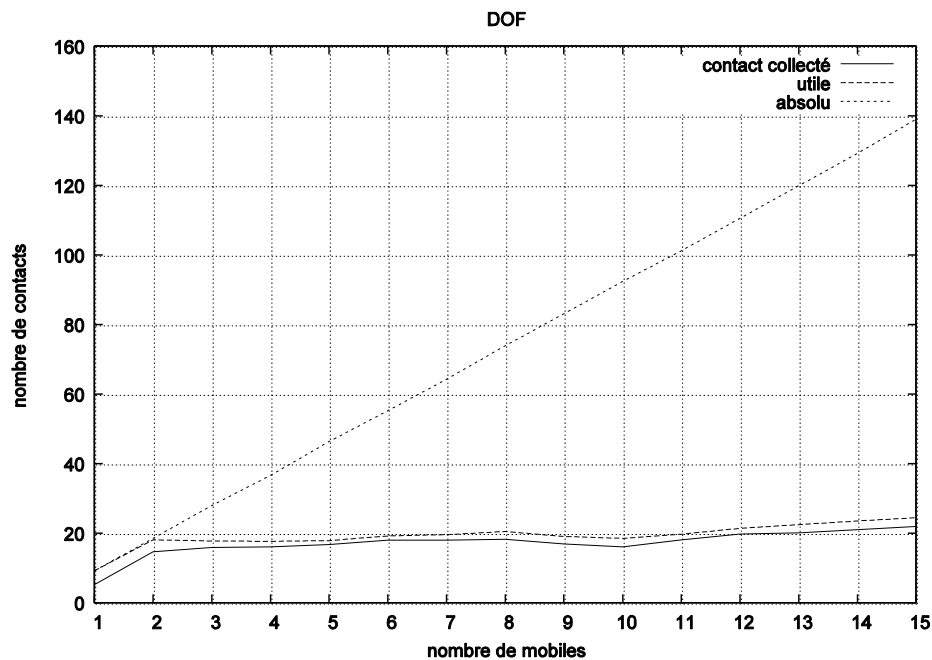
Ces résultats dépendent fortement du choix du filtre appliqué selon l'application visée, la capacité mémoire des nœuds, et aussi des scénarii de déplacement. Le tableau 6 résume les paramètres de simulation fixés pour l'évaluation du mécanisme de filtrage.

|                              |  |
|------------------------------|--|
| Simulateur                   | NS2 [76] version 2.31  |
| Protocole d'accès au médium  | IEEE 802.15.4 CSMA/CA non slotté [34]  |
| Type de trame de signalement | Trames diffusées   |
| Débit                        | 250 Kbps   |
| Nombre de mobiles            | $N_m$ avec $N_m \in [1 \dots 15]$  |
| Fréquence de signalement     | $1/T$ avec $T \in [0,1s \dots 2s]$   |
| Durée de simulation          | 150 secondes   |
| Dérive de l'horloge          | $T_i$ , avec $T_i = T [1 + \varepsilon_i]$ et $\varepsilon_i \in [-x, +x]$ ppm ( $x =$ |

|                               |  |
|-------------------------------|--|
|                               | 30 $\mu$ sec dans les résultats qui vont suivre) |
| Modèle de propagation         | ITU  |
| Nombre de répétitions         | 25   |
| Déplacement des nœuds mobiles | Aléatoire entre les zones d'activité X et Y      |

**Tableau 5 : Paramètres de simulation pour l'évaluation du mécanisme de routage.**

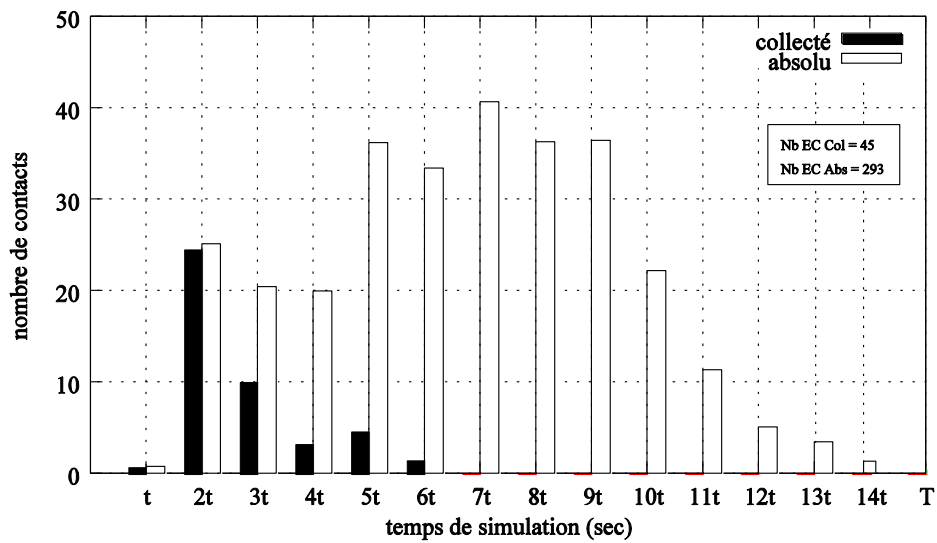
Concernant la figure 90 pour le filtre des doublons (DOF), notons que la différence entre le nombre de contacts colportés et la connaissance utile est très faible, car dans ce cas, les nœuds n'appliquent aucune autre politique de filtrage mise à part la suppression des événements doublons. D'autre part, lorsque ces deux courbes sont comparées au nombre de contact absolu dans la simulation, le nombre de contacts pouvant être déduits est très élevé ce qui implique une nécessité de filtrer l'information.



**Figure 90 : Nombre de contacts pour le filtre permanent.**

On constate notamment que les filtres doublons (DOF), duals (DUF) et concaténation (COF) présentent les mêmes résultats. Cette similarité s'explique par la procédure de comptage des contacts qui regroupe plusieurs événements de contact relatifs à un même contact et ignore les événements de contact doublons. Pour justifier ces résultats, la figure 91 montre la répartition dans le temps des contacts reçus avec les filtres dual (DUF) et concaténation (COF).

### DUF et COF Capacité mémoire d'un noeud 25 places



**Figure 91 : Distribution des contacts pour les filtres permanents.**

On constate sur cette figure que les événements de contact colportés jusqu'au point de collecte sont ceux créés au début de la simulation représentant ainsi les rencontres les plus anciennes. Ce phénomène est dû à la saturation de la mémoire des nœuds qui intervient tôt étant donné le rôle des filtres permanents qui ne filtrent que les doublons.

Les résultats obtenus lorsque le filtre aléatoire avec probabilité  $P$  (PRF) est appliqué sont montrés dans la figure 92. On note à travers cette figure que le filtre PRF augmente de plus de 100% le nombre de contact reçus par le collecteur. Cela est dû au mécanisme de filtrage utilisé lorsque la mémoire des nœuds est saturée. On constate aussi un enrichissement de l'information utile propagée dans le réseau qui se rapproche de l'information absolue créée dans la simulation, ce qui confirme l'amélioration des performances.

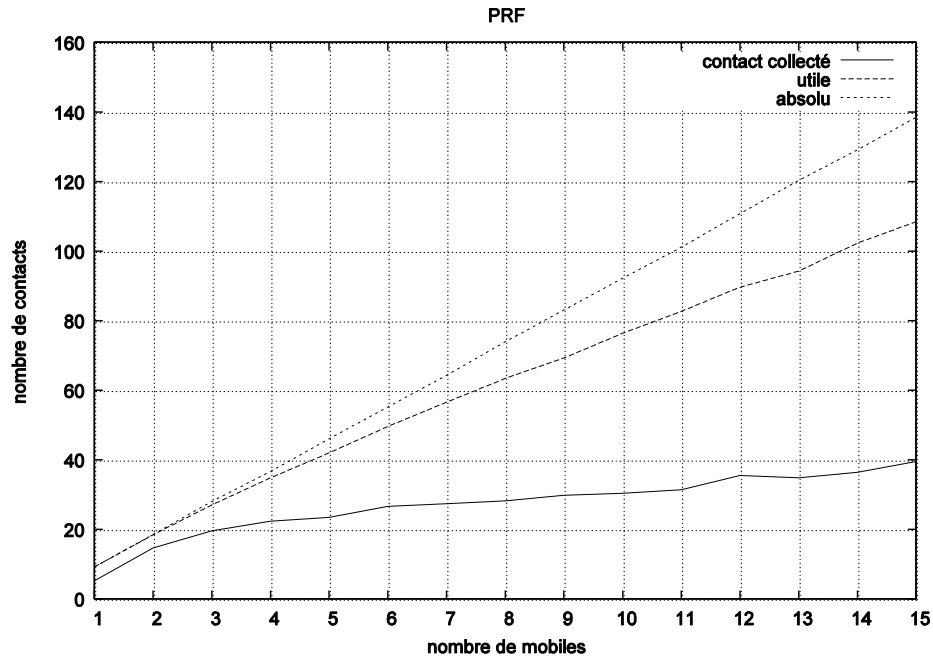


Figure 92 : Nombre de contact pour le filtre aléatoire.

La distribution des contacts est montrée dans la figure 93. On constate à travers cette figure que la distribution des contacts est bien répartie sur toute la durée de la simulation contrairement à la première distribution. L'extrait des contacts propagés représente non plus seulement le début du scénario simulé mais est un échantillonnage des événements de contact produit tout au long du scénario.

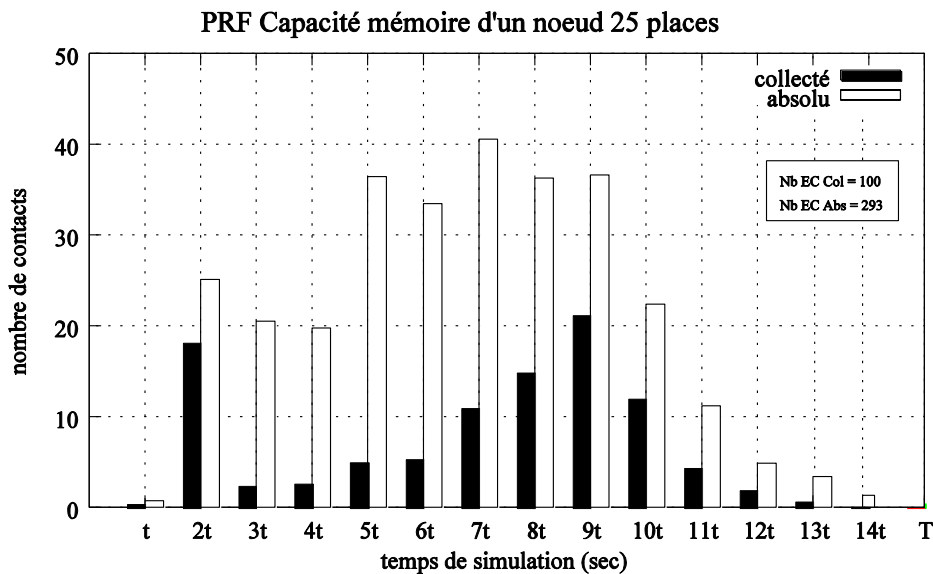


Figure 93 : Distribution des contacts pour le filtre aléatoire.

Enfin, les meilleurs résultats sont obtenus lorsque le filtre donnant une priorité aux contacts impliquant des nœuds balises est mis en œuvre, comme le montre la figure 94. Le nombre de contacts collectés augmente de plus de 270% comparé aux filtres DOF, DUF, COF et de plus de 80% comparé au filtre

aléatoire PRF. La répartition des contacts au cours du temps est illustrée dans la figure 95. On constate clairement que c'est la plus uniforme des distributions obtenues au cours des simulations.

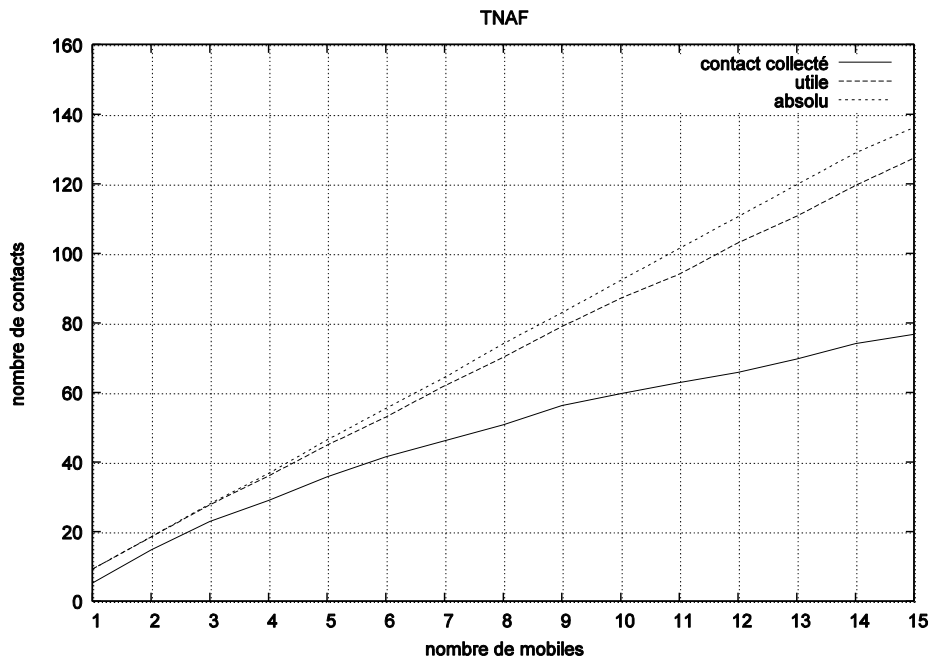


Figure 94 : Nombre de contact pour le filtre avantage balise.

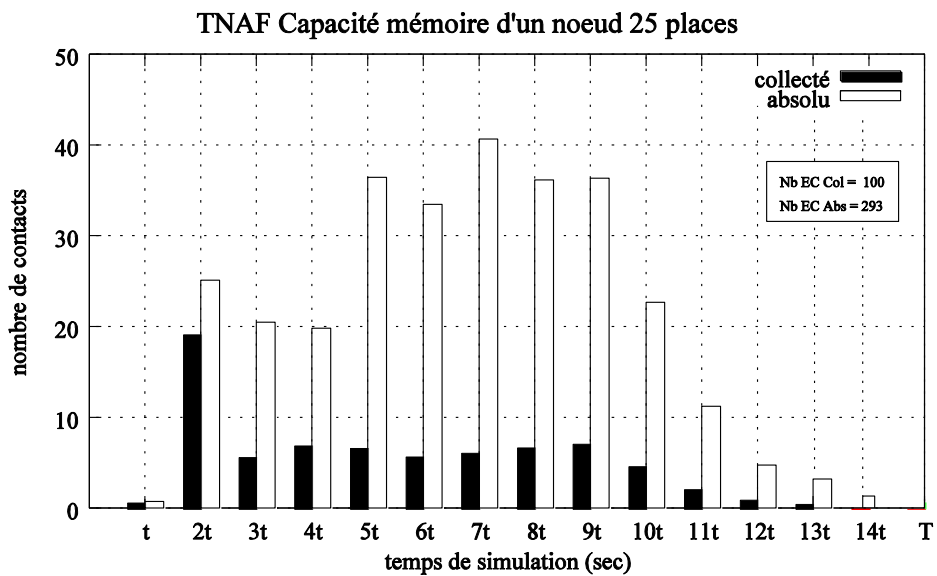


Figure 95 : Distribution des contacts pour le filtre avantage balise.

Ces premiers résultats obtenus en comparant les différents filtres confirment la nécessité d'appliquer des stratégies de filtrage lors de la saturation de la mémoire et montrent également des variations de performances selon le type de filtre appliqué.

### 4.3.2.3. Effets de la taille mémoire

Nous allons à présent exposer les résultats de simulations obtenus en faisant varier la capacité mémoire des nœuds dans le réseau [87]. La même métrique de calcul du nombre de contacts déduits, absolus et utiles est reprise pour représenter l'effet de la taille de la mémoire. La taille mémoire est supposée être la même pour tous les types de nœud, elle est exprimée en nombre de places : une place servant à stocker un événement de contact.

Pour cela nous proposons la métrique suivante : On suppose que la distribution de tous les contacts absolus induits par l'activité des mobiles dans la simulation ainsi que les contacts collectés au point de collecte est une distribution de type gaussienne. Afin d'évaluer la diversité de cette collection de contacts, nous considérons le ratio de l'écart-type de ces deux gaussiennes. Nous rappelons que la fonction gaussienne est définie par :

$G(t) = (1 / \sigma \sqrt{2\pi}) e^{- (t - \mu)^2 / 2 \sigma^2}$  où le paramètre  $\sigma^2$  représente la variance et le paramètre  $\mu$  est la moyenne.

Pour évaluer la diversité des contacts, nous nous concentrons uniquement sur la variance  $\sigma^2$  qui peut être assimilée à la largeur à mi-hauteur de la courbe de Gauss. La diversité est donc définie par la fonction :

$R = \sigma_1 / \sigma_2$  ( $\sigma_1$  : pour les contacts collectés et  $\sigma_2$  pour les contacts absolus). La valeur de R montre une bonne diversité lorsque sa valeur se rapproche le plus de 1.

La figure 96 montre le nombre de contacts colportés par les mobiles jusqu'au point de collecte. Le concept de la connaissance utile (notée  $G$ ) est aussi mis en valeur dans cette figure, il s'agit pour cela de compter la quantité d'informations que les nœuds mobiles soumettent au médium. Le nombre de contacts n'augmente pas considérablement en fonction de la taille mémoire et cela à cause du filtre DOF qui ne supprime que les événements de contact doublons. Nous pouvons justifier ces résultats en montrant la distribution des contacts par rapport à la durée de la simulation. Les figures 97 et 98 montrent la répartition des contacts pour une capacité mémoire de 15 et 100 places.



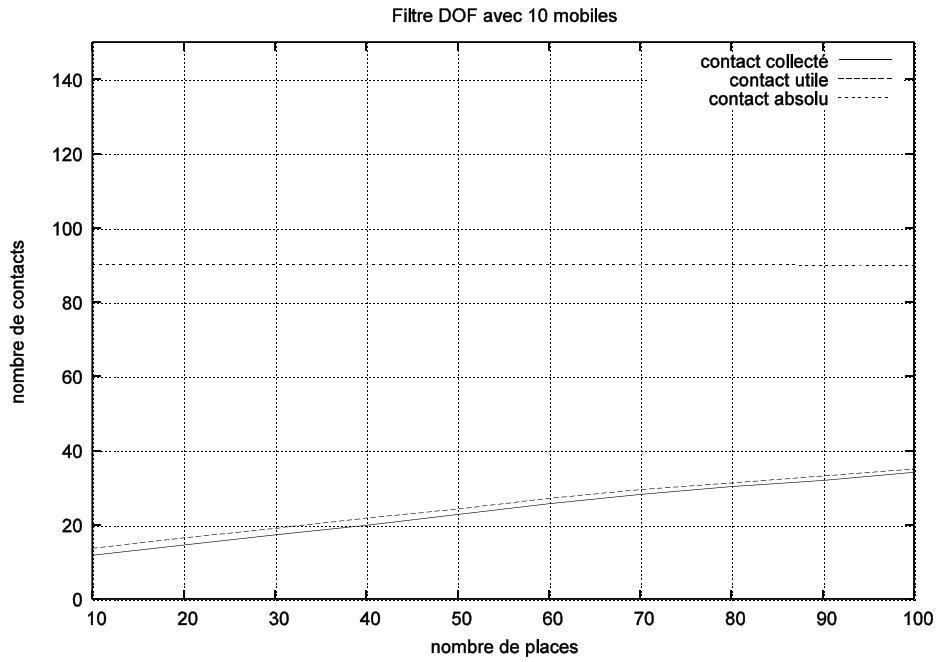


Figure 96 : Effet de la taille mémoire pour le filtre permanent.

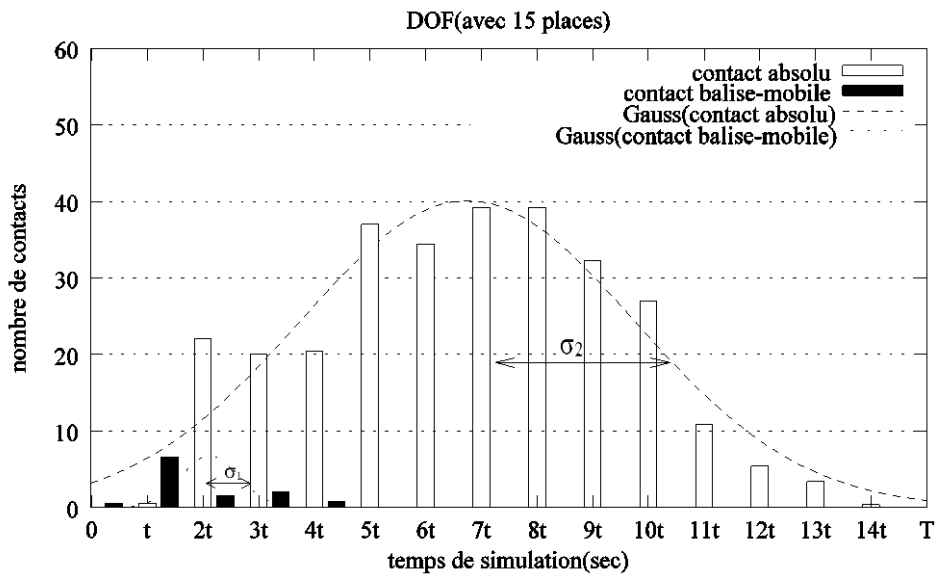
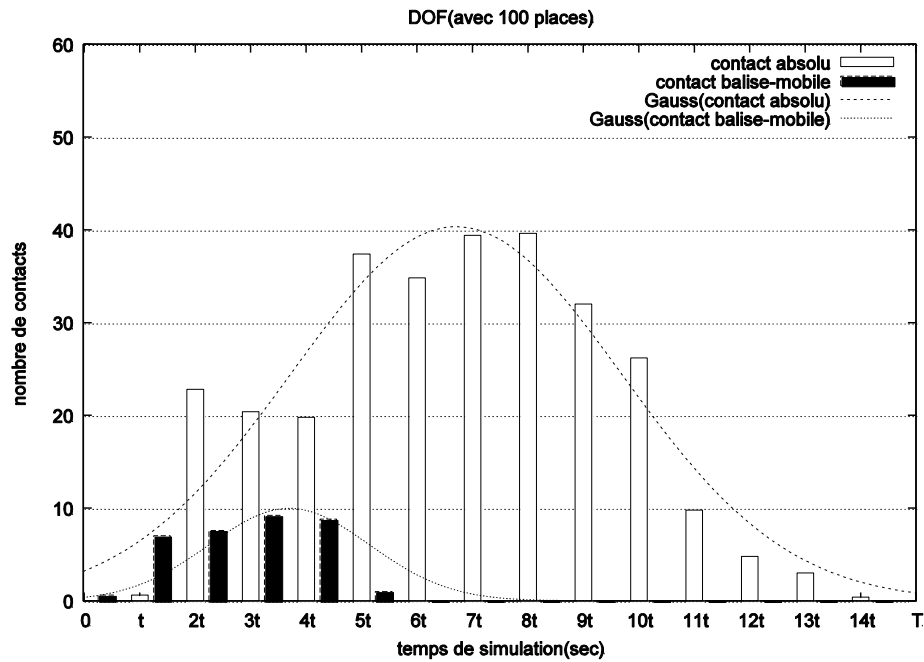


Figure 97 : Distribution des contacts pour le filtre permanent et pour une capacité de 15 places.



**Figure 98 : Distribution des contacts pour le filtre permanent et pour une capacité de 100 places.**

Les deux figures montrent que les contacts collectés font partie des contacts créés au début de la simulation. Le ratio entre la variance  $\sigma_1$  (contacts absolus) et  $\sigma_2$  (contacts balise mobile) est égal à 33% lorsque 15 places mémoire sont utilisées et 50% pour 100 places. Nous concluons que le filtre DOF n'est pas suffisant pour les applications de traçabilité où la variété de l'information est un objectif primordial.

La figure 99 illustre les effets de la mémoire lorsque le filtre est PRF appliqué. Le nombre de contacts collectés et utiles dans ce cas augmente de manière significative en faisant varier la taille de la mémoire des nœuds. La répartition des contacts pour 15 et 100 places (figures 100 et 101) montre une plus grande variété de contacts colportés au point de collecte lorsque 100 places sont attribuées aux nœuds mobiles. Le ratio  $\sigma_1/\sigma_2$  confirme ces résultats pour 15 et 100 places comme le montre le tableau 7.

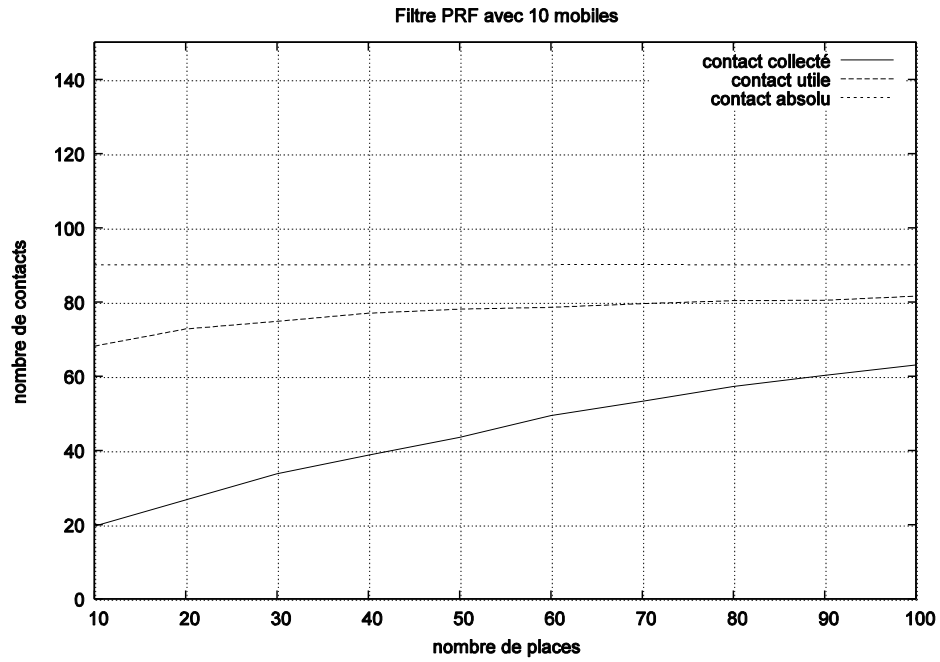


Figure 99: Effet de la taille mémoire pour le filtre aléatoire.

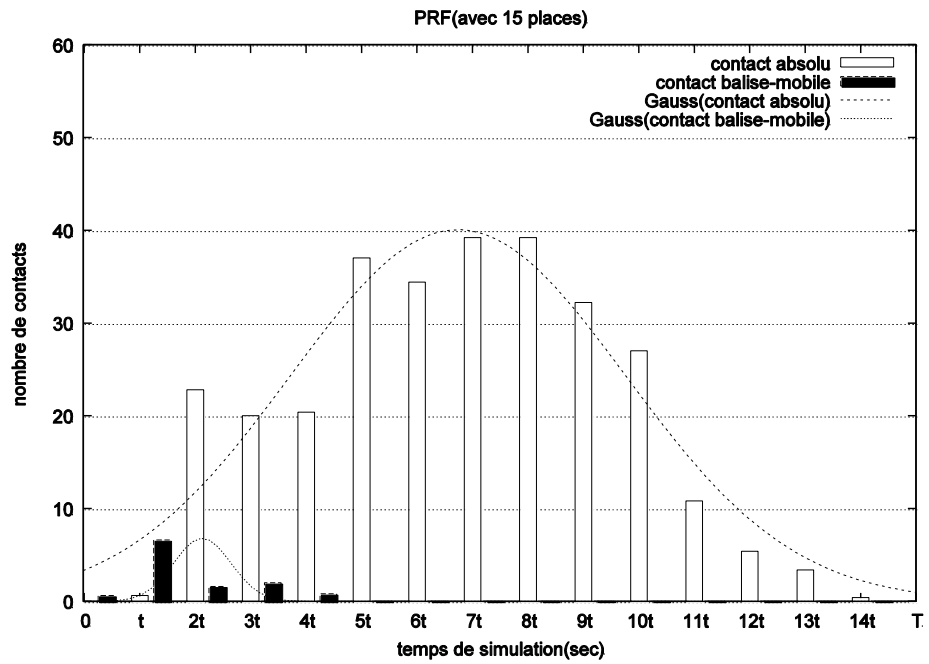
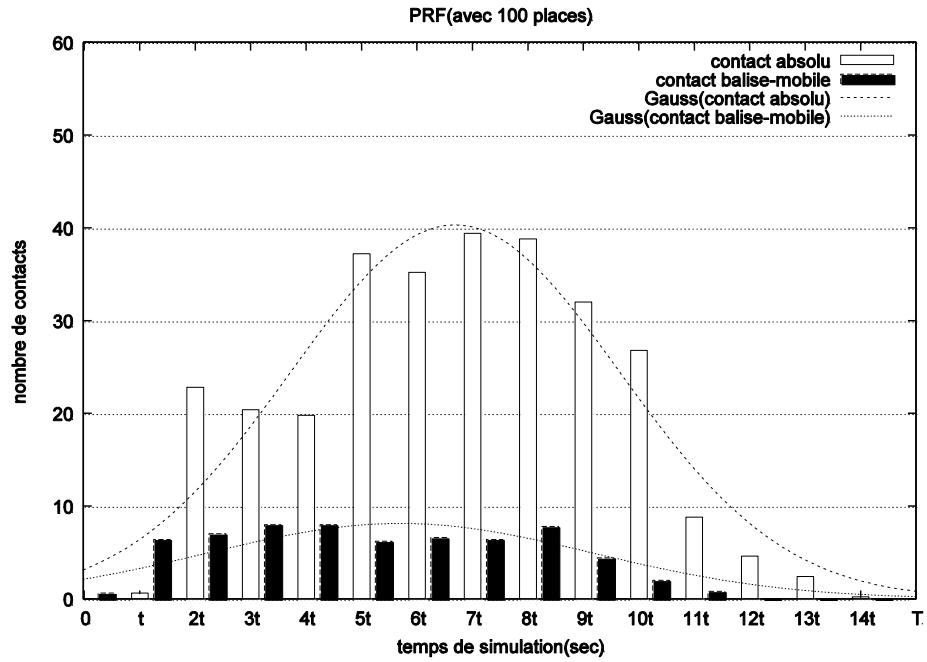


Figure 100 : Distribution des contacts pour le filtre aléatoire et pour une capacité de 15 places



**Figure 101: Distribution des contacts pour le filtre aléatoire et pour une capacité de 100 places.**

On note que les meilleures performances sont obtenues avec le filtre TNAF (figure 102) qui permet aux contacts de type balise-mobile d'avoir une meilleure chance d'atteindre le point de collecte. Un autre point important pour ce filtre est lorsque la taille de la mémoire atteint une valeur de 70 places, le gain en nombre de contacts collectés se stabilise. La répartition des contacts pour 15 et 100 places (figures 103 et 104) montre aussi une grande variété des contacts colportés. La diversité des contacts collectés est améliorée lorsque 15 places sont allouées avec  $\sigma_1/\sigma_2 = 90\%$  et la même valeur est obtenue avec 100 places (tableau 6).

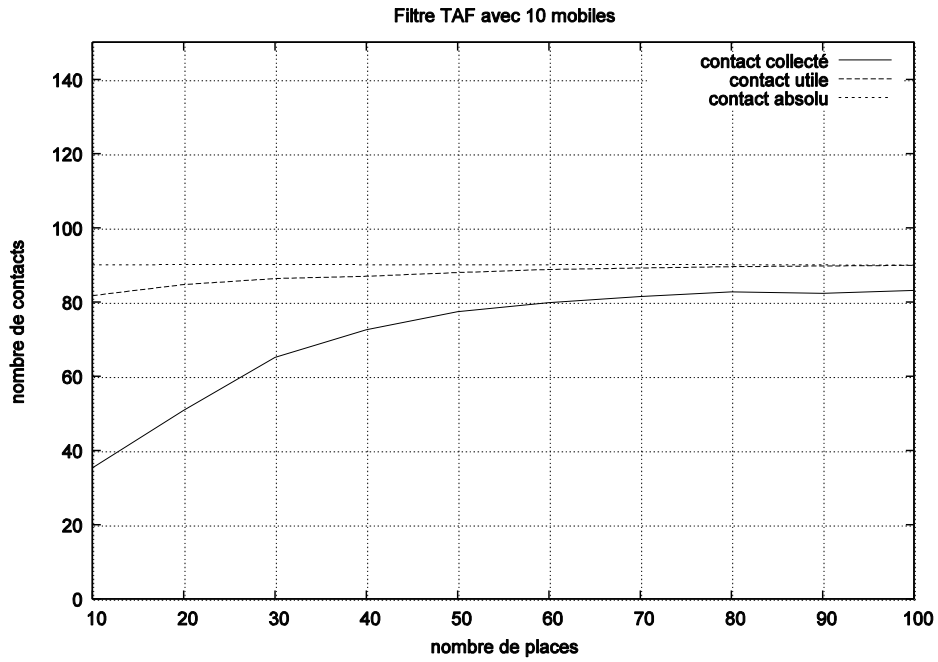


Figure 102: Effet de la taille mémoire pour le filtre avantage balise.

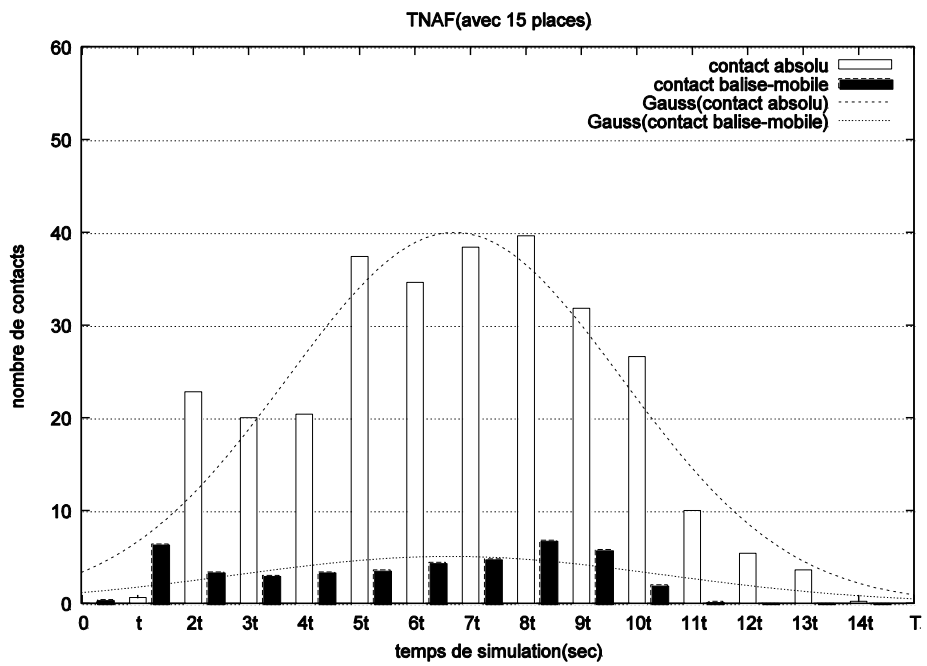
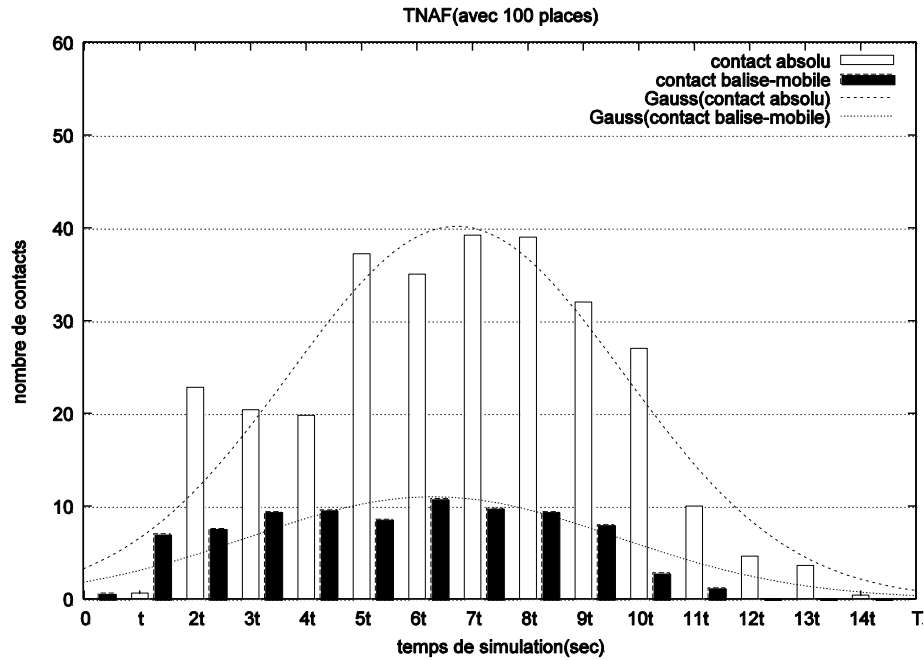


Figure 103: Distribution des contacts pour le filtre avantage balise et pour une capacité mémoire de 15 places.



**Figure 104: Distribution des contacts pour le filtre avantage balise et pour une capacité mémoire de 100 places.**

| Places mémoire | DOF | PRF | TNAF |
|----------------|-----|-----|------|
| 15 places      | 33% | 33% | 90%  |
| 100 places     | 50% | 90% | 90%  |

**Tableau 6 : Diversité des contacts selon la capacité mémoire et le type de filtre.**

Les résultats de simulation obtenus dans cette section confirment l'impact de la taille mémoire sur la quantité d'information colportée jusqu'au point de collecte. Cette taille de mémoire peut être calibrée à une certaine valeur en fonction du scénario de mobilité. La distribution des contacts dans le temps montre l'importance d'avoir une grande variété de contacts, cette diversité de connaissances peut être bénéfique pour des applications de traçabilité. Nous pourrions montrer de la même façon que pour une application de localisation l'usage de filtre qui avantage les événements de contact les plus récents donne les meilleurs résultats car il ne s'agit plus d'avoir une image de tout ce qui s'est passé mais de connaître la dernière position d'une entité

#### 4.4. Validation par maquettage

La partie de ce chapitre décrit l'expérimentation de notre application LoWCA. Les résultats obtenus par maquettage seront ensuite comparés à ceux obtenus par simulation afin de valider partiellement notre mécanisme de routage. Un autre objectif visé par le déploiement de ce projet est l'expérimentation d'applications de localisation dans la mine expérimentale CANMET au Québec.

#### 4.4.1. Implémentation sur des cartes TelosB

Le choix matériel pour l'implémentation de notre application de localisation a posteriori s'est porté sur la plateforme TelosB [86] proposée par l'intégrateur *Crossbow Technology* [88]. Cette plateforme a été conçue dans le but d'apporter une solution économe en énergie et capable d'intégrer plusieurs types de capteurs (capteurs de température, d'humidité, de lumière... etc.). L'architecture d'une plateforme TelosB est donnée dans la figure 105.

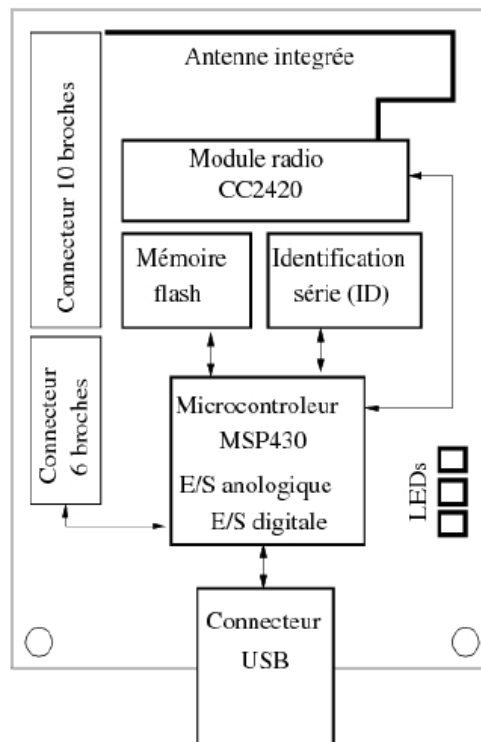


Figure 105: Architecture d'une plateforme *TelosB*.

Les caractéristiques de cette architecture sont les suivantes :

- Module radio : la plateforme *TelosB* est équipée d'un module radio de type *Texas Instruments CC2420* permettant une compatibilité avec le standard IEEE 802.15.4 et avec la norme ZigBee.
- Antenne interne : en milieu intérieur, la portée radio est estimée environ à 50 mètres et environ 125 mètres en extérieur.
- Microcontrôleur : le microcontrôleur incorporé dans la carte *TelosB* est de type MSP430 avec une cadence de 8 MHz, il est doté d'une mémoire interne SRAM de 48 Ko..

- Port USB : un port USB est présent sur la carte afin de simplifier l'interface utilisateur. Cette interface est aussi utilisée pour flasher la carte et pour la récupération des données.
- Faible consommation : deux piles de types AA permettent des expérimentations d'une durée largement suffisante dans notre cas.
- Système d'exploitation : une compatibilité avec le système d'exploitation TinyOS [89] dans sa version 1.1 est possible. Ce système *Open Source* est largement diffusé et utilisé par la communauté des développeurs de ce type de plateforme.

Un aperçu de cette carte ainsi qu'une description de ses différents composants sont donnés dans la figure 106.

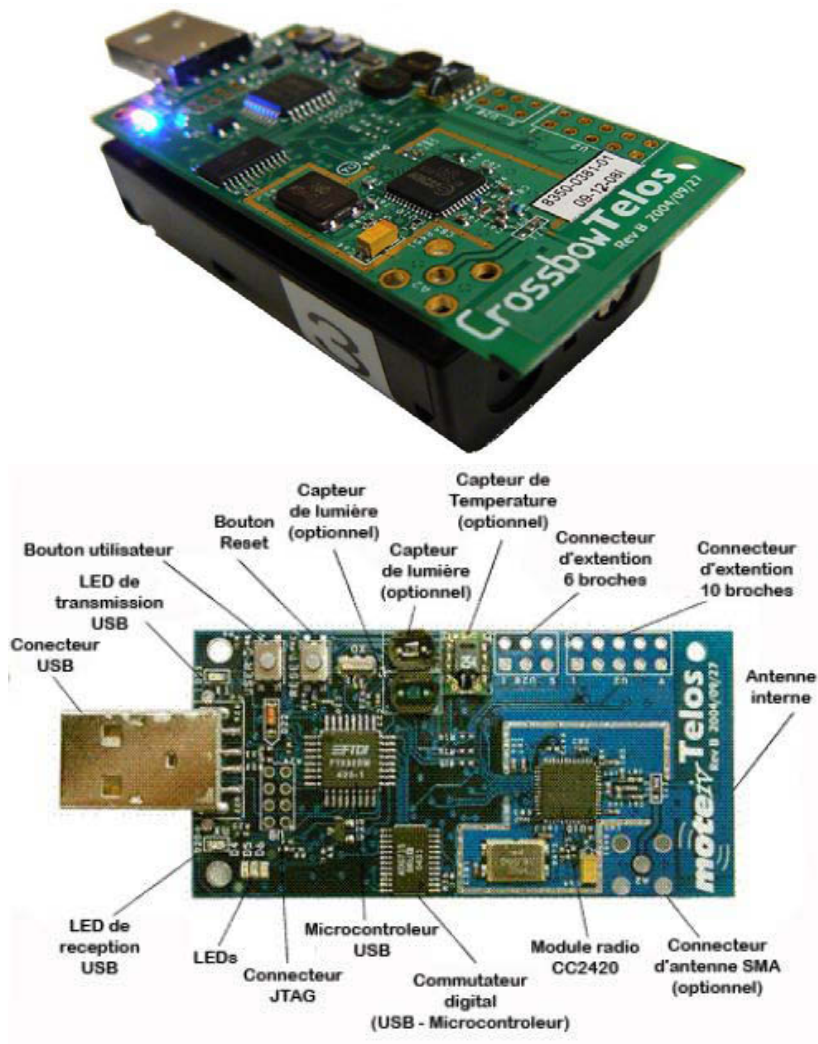


Figure 106: Présentation d'une plateforme *TelosB*.



#### 4.4.1.1. Scenarii des tests

Les tests ont été effectués au sein des bâtiments de l'IUT d'Auvergne. Pour ce faire, un scénario de déplacement impliquant 5 entités mobiles a été établi et reproduit par simulation. Ce scénario est assez simple, mais suffisamment complexe pour produire une panoplie de résultats permettant l'évaluation de mécanisme. Le scénario choisi consiste en une séquence de mouvements effectués par des étudiants dotés de cartes TelosB au sein des couloirs d'un bâtiment. Ce même scénario est ensuite rejoué sur le simulateur NS2. Le tableau 8 donne une description détaillée de ce scénario.

| Phase | Identifiants mobiles concernés | Action effectuée                            |
|-------|--------------------------------|---|
| 1     | M1 et M2                       | Positionnement en zone de départ            |
| 2     | M1 et M2                       | Déplacement vers la zone du nœud collecteur |
| 3     | M1 et M2                       | Déplacement vers la zone de la balise B1    |
| 4     | M1 et M2                       | Déplacement vers la zone de la balise B2    |
| 5     | M3, M4 et M5                   | Positionnement en zone de départ            |
| 6     | M3, M4 et M5                   | Déplacement vers la zone du nœud collecteur |
| 7     | M3, M4 et M5                   | Déplacement vers la zone de la balise B1    |
| 8     | M3, M4 et M5                   | Déplacement vers la zone de la balise B3    |
| 9     | M1 et M2                       | Retour vers la zone de la balise B1         |
| 10    | M1 et M2                       | Retour vers la zone du nœud collecteur      |
| 11    | M1 et M2                       | Retour vers la zone de départ               |
| 12    | M3 et M4                       | Retour vers la zone de la balise B1         |
| 13    | M3 et M4                       | Retour vers la zone du nœud collecteur      |
| 14    | M3 et M4                       | Retour vers la zone de départ               |

Tableau 7 : Scenarii des tests.

#### 4.4.1.2. Validation du mécanisme de colportage

Nous allons à présent exposer les différents résultats d'expérimentation et de simulation afin de donner une première validation du mécanisme de colportage. Une taille mémoire de 15 places a été choisie afin de montrer par la suite l'effet des filtres.

Les figures 107, 108 et 109 montrent les résultats obtenus en simulation et ceux obtenus par expérimentation. Dans chaque figure, le nombre théoriques de contact prévu a été ajouté pour plus de clarté.

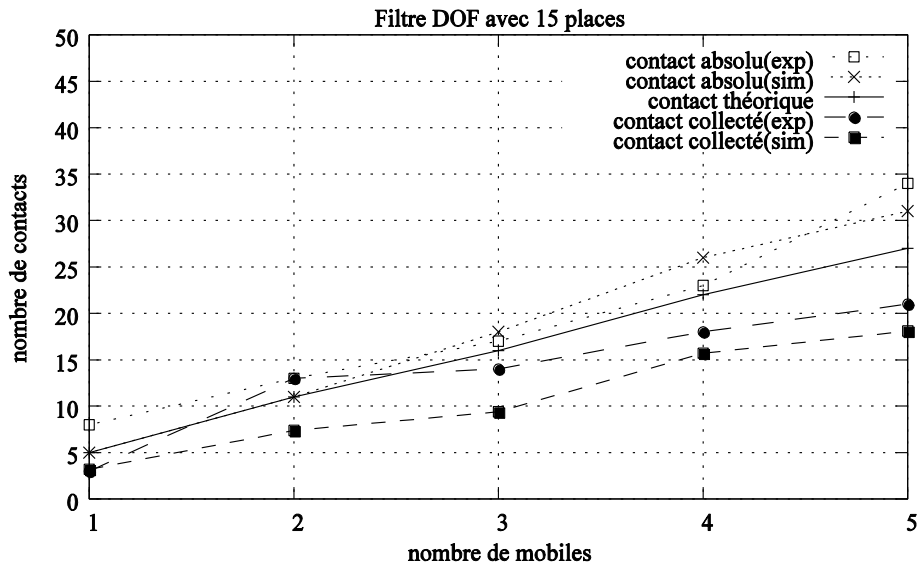


Figure 107: Simulation et expérimentation pour le filtre permanent.

La figure 108 présentant les résultats obtenus à l'aide du filtre DOF montre une similitude entre les résultats de simulations et d'expérimentations. Nous constatons dans cette figure le comportement concernant le nombre de contacts collectés au point de collecte qui reste inférieur au nombre de contact théorique attendu, cette perte est due au comportement aléatoire du médium. Le nombre de contacts absolus augmente de façon significative au niveau des expérimentations juste après que le nombre de mobiles atteint la valeur de 4 tandis que dans la simulation, ce nombre se rapproche du nombre théorique, ceci est dû au fait que les expérimentations créent plus de faux contacts que les simulations. Ces faux contacts sont produits lorsque les nœuds se retrouvent en bordure d'une cellule où la connectivité est souvent instable.

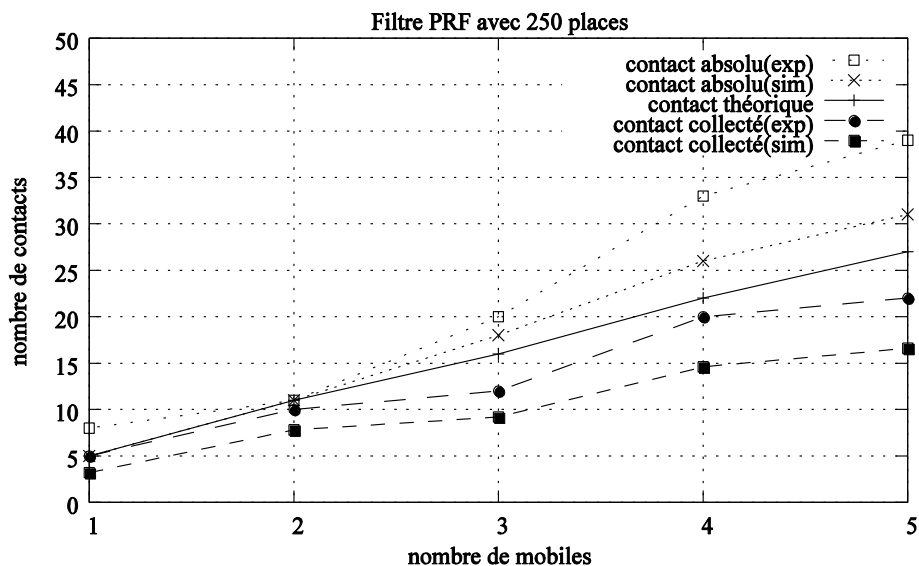


Figure 108 : Simulation et expérimentation pour le filtre aléatoire.

Les mêmes remarques s'appliquent lorsque le filtre aléatoire PRF est utilisé (figure 108) : les places sont alors libérées de manières aléatoires. Le jeu de courbes le plus homogène est obtenu avec le filtre

TNAF (figure 109) où une priorité est donnée aux événements de contact impliquant des nœuds balises.

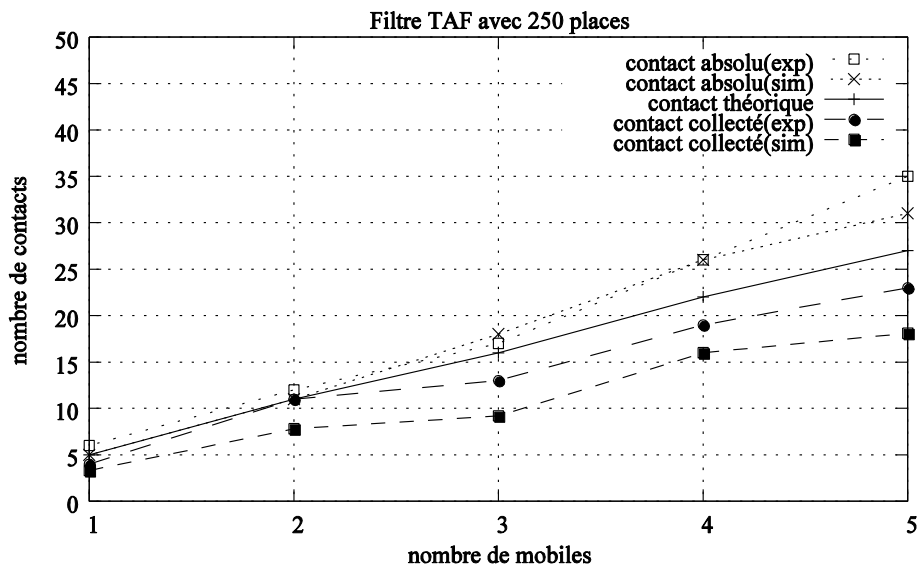


Figure 109: Simulation et expérimentation pour le filtre avantage balise.

Les résultats obtenus ci-dessus rappellent l'intérêt d'appliquer des stratégies de filtrage quand il s'agit d'obtenir une diversité de contact au niveau du point de collecte, et cela pour alimenter et améliorer les applications de localisation et de traçabilité. En comparant la simulation et les résultats expérimentaux avec le même modèle de mobilité, nous avons montré une première preuve de la faisabilité de notre solution de routage

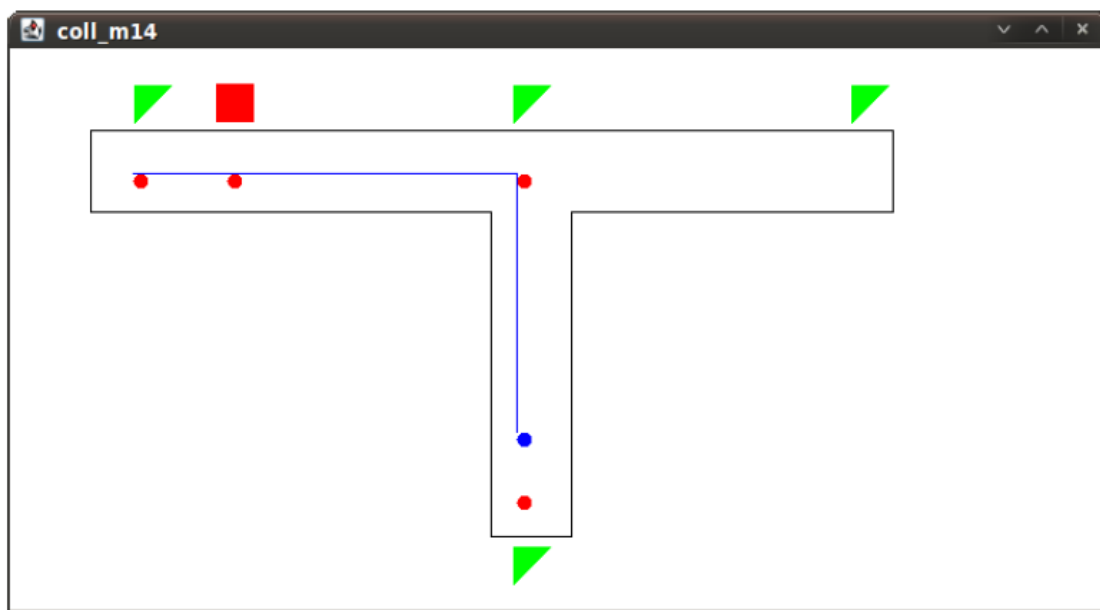
#### 4.4.2. Application graphique

Dans le but d'apporter une meilleure exploitation des résultats obtenus lors des simulations et des expérimentations, nous avons eu recours à des projets d'étudiants pour développer les bases d'une vitrine applicative. Ces projets ont consisté à développer une application graphique dont la tâche principale est de reproduire a posteriori les mouvements des entités mobiles évoluant au sein d'une mine dans le cas de la localisation, à partir d'informations extraites de résultats issus de simulation et de l'expérimentation. Les principales fonctionnalités de l'application graphiques sont résumées dans les points suivants :

- L'extraction et l'exploitation des informations requises pour la reproduction de la mine générique à partir de fichiers fournis par le simulateur NS2 (fichier Galerie qui décrit le polygone des parois de la mine aussi que le nombre, la position et le rôle de chaque entité).
- L'évaluation des mouvements effectués par les nœuds mobiles durant la simulation (fichier NAM : Network Animator produit par NS2).

- L'exploitation des fichiers résultats de simulation et d'expérimentation (fichier de collecte des évènements de contact) afin de reproduire le mouvement des stations mobiles dans la mine.
- L'affichage et la comparaison entre les parcours obtenus, par la simulation et par l'expérimentation.

Un exemple d'animation est donné dans la figure 110 où un mobile suit un parcours dans une mine générique.



**Figure 110 : Exemple d'une animation graphique.**

Sur cette figure on distingue d'abord le profil de la mine générique en forme de T. Ensuite, les nœuds fixes sont positionnés : à l'entrée de la mine en ce qui concerne le nœud collecteur, les trois autres stations fixes sont des balises placées aux extrémités de la galerie et une au milieu de la mine. Enfin, les différents points ronds représentent le déplacement du mobile au sein de la galerie. Cet exemple montre bien l'intérêt d'une telle application lorsqu'il s'agit d'apporter en complément la visualisation de la connaissance récoltée au niveau du collecteur. Néanmoins, quelques perspectives sont envisagées pour apporter quelques améliorations au niveau du traitement de la vitesse des mobiles, de la prise en charge des évènements de contact non finis et des contacts de type mobile-mobile.

## 4.5. Conclusions

Ce dernier chapitre de thèse a été consacré en majorité à l'évaluation de notre approche générique de colportage au sein d'une population mobile à travers des simulations à l'aide de l'outil de simulation NS2.

La première partie de ce chapitre a exposé les résultats de simulation pour l'étude du comportement des nœuds dans une cellule. Cette section a permis également d'étudier la méthode d'accès au médium CSMA/CA de la norme 802.15.4 dans le mode non beacon en surchargeant la cellule progressivement. Les résultats obtenus à l'aide du simulateur NS2 ont permis d'évaluer plusieurs caractéristiques de ce comportement comme le débit, le nombre de drops au niveau de la couche MAC, les collisions et le taux de perte de trames. L'objectif principal de cette étude était d'éviter la création de faux événements de contact produits particulièrement par les pertes de trames consécutives. Les résultats obtenus dans cette partie ont permis de fixer le nombre maximal de nœuds qu'une cellule peut supporter sans risquer une création trop grande de faux événements de contact.

L'évaluation du mécanisme de routage a été ensuite abordée en comparant les effets d'une liste de règles de filtrage lorsque la mémoire des nœuds est saturée. Les résultats ont montré une variation des performances selon le type de politique de filtrage appliqué. L'impact de la taille mémoire sur la qualité et la pertinence de la connaissance colportée a aussi été démontré à l'aide de ses simulations selon le scénario de mobilité choisi. De plus, selon les mêmes critères, une taille mémoire peut être fixée au préalable.

Une partie expérimentale a décrit le maquettage de notre application LoWCA au sein d'un bâtiment de l'IUT de l'université d'Auvergne. Les résultats obtenus par maquettage ont été ensuite comparés à ceux obtenus par simulation afin de donner une première démarche de validation de notre mécanisme de routage. Une des perspectives envisagées est de déployer à grande échelle cette solution de localisation d'entités mobiles a posteriori, dans une mine réservée à l'expérimentation au Québec.

# Conclusions et perspectives

## 5.1. Contribution de la thèse

De nos jours, la technologie des réseaux de capteurs sans fil (RCSF) représente une alternative à certaines problématiques liées à l'industrie moderne et à l'environnement. Les solutions offertes répondent à une variété de besoins lorsqu'il s'agit d'observer, de contrôler et de traiter des grandeurs physiques allant de la température ambiante à la pression atmosphérique, au mouvement voire même aux capteurs de luminosité. Une autre caractéristique comme la mobilité des nœuds rendent cette catégorie de réseaux utile aux domaines d'applications traitant de la localisation et de la traçabilité. En effet, la possibilité de suivre des personnes ou des matériaux susceptibles d'être des agents contaminants, de faire face à des infections nosocomiales ou bien d'étudier des propagations de virus dans un contexte médical sont autant de défis à relever. Ceci est d'autant plus pertinent que les solutions de type GPS, s'avèrent inappropriées dans des milieux dits confinés ou d'intérieurs en raison des propriétés liées à ce type d'infrastructure, et ne permettent pas de répondre à ses attentes de géolocalisation.

L'investigation réalisée dans le cadre de cette thèse tient compte de la richesse des différents domaines applicatifs des RCSF tout en prenant en considération leurs contraintes. Elle vise à proposer un modèle générique basé sur l'échange de proche en proche d'informations dites atomiques représentant des rencontres entre entités fixes et/ou mobiles. Lorsque le domaine visé traite de la reconstitution de parcours, la connaissance qu'une station mobile, qui est passée à proximité d'une autre station, s'avère une information primordiale.

Après avoir exposé, dans un premier chapitre, un état de l'art relatif à l'objet de notre travail, un second chapitre a modélisé de manière non exhaustive le principe de colportage dans une population mobile. A l'aide d'un outil de modélisation (UML), le mécanisme de diffusion, en même temps que les stratégies de filtrage associées, ont été expliquées sous forme de diagrammes. Ce chapitre a été suivi d'une troisième partie plus détaillée dans le but de formuler tous les aspects de la solution proposée, dédiée à des applications de localisation et de traçabilité a posteriori dans des milieux d'intérieurs. Cette solution a été appelée LoWCA. Les principaux points abordés dans ce chapitre ont été en premier lieu l'exposition de la topologie et du modèle de déplacement choisis afin de représenter des entités mobiles en activité. Une grande partie a été réservée à l'exposition de notre approche de routage destinée aux réseaux tolérants aux délais et une liste de tous les algorithmes de filtrage a été dressée.

Une démarche de simulation à l'aide de l'outil NS2 a été présentée dans le quatrième chapitre de cette thèse afin de proposer une évaluation du modèle générique. Les performances de la méthode d'accès au médium de la norme 802.15.4 version non slottée ont été évaluées dans la première section de ce chapitre. Les résultats obtenus ont servi à fixer des limites à l'intérieur d'une cellule dans le but d'éviter la création de fausses informations concernant les rencontres entre les entités fixes ou mobiles. Une seconde partie a été consacrée à l'évaluation des effets des filtres sur la qualité de la connaissance colportée jusqu'au point de collecte. Ces informations ainsi récoltées auront un impact direct sur les applications de localisation et de traçabilité a posteriori. Les résultats de simulations ont clairement mis en relief la nécessité d'appliquer des règles de filtrage et disposer de connaissances nécessaires pour apporter une amélioration du processus de localisation.

Dans le but de donner une première démarche de validation de cette approche d'échange de proche en proche, un maquettage à l'aide de carte TelosB a permis d'obtenir des résultats expérimentaux comparables à ceux obtenus par simulation.

## **5.2. Perspectives**

Afin de donner une dimension plus complète à la voie d'investigation explorée dans le cadre de cette thèse, nous envisageons d'approcher d'autres domaines applicatifs. En effet, même si l'un des principaux objectifs escomptés dans ce domaine de recherche a été la satisfaction de toutes les applications projetées, il n'en demeure pas moins que seul cas d'utilisation a été réalisé pour l'évaluation de l'approche suivie. D'autres axes complémentaires, notamment dans un domaine aussi sensible que la santé dans les enceintes hospitalières, nous semblent présenter un intérêt quant au mode de suivi des personnes, des matériaux pouvant être des agents contaminants, des infections nosocomiales et des propagations de virus...etc. Si maintenant il s'agit d'adapter cette étude par des applications d'ordre social, il faut imaginer et proposer des améliorations spécifiques. Ce serait le cas par exemple pour des applications d'analyse du comportement des visiteurs dans des expositions, des salons ou des lieux publics.

L'une des difficultés recensées au cours de cette thèse et plus précisément durant l'implémentation et l'évaluation de notre approche de routage a été les temps de simulations nécessaires pour obtenir des résultats. Malheureusement, en plus d'être un outil de simulation difficile à maîtriser, NS2 souffre aussi d'une grande lenteur de calcul lorsque la mobilité des nœuds est prise en considération. Ces temps de calcul peuvent durer des jours. C'est pour cette raison que l'évaluation de notre mécanisme de routage s'est limitée à une liste restreinte de filtres. De ce fait, l'étude de tous les filtres définis durant de cette thèse paraît être une perspective raisonnable.

Enfin, la partie expérimentale de ce rapport a donné lieu à une première approche de validation du mécanisme de routage utilisé. Néanmoins, les résultats obtenus lors des expérimentations effectuées au niveau des locaux de l'IUT d'Auvergne ont été limités par l'usage de quelques stations mobiles seulement (5 mobiles au maximum). Tenant compte de cette contrainte, nous envisageons de réaliser dans le futur des tests à grande échelle dans des galeries minières expérimentales.



## Liste des publications

01. C.Baouche, A.Freitas, and M.Misson. "A WPAN model to locate mobile entities in a mine gallery". International Conference on Wireless Communications in Underground and Confined Areas, 2008, p. 183-187, Val-d'Or, Canada.
02. C.Baouche, A.Freitas, and M.Misson. "Radio Proximity Detection in a WSN to Localize Mobile Entities within a Confined Area". Journal of Communications, volume 4, pages 232–240, 2009.
03. C.Baouche, A.Freitas, and M.Misson. "Architecture d'un réseau sans fil exploitant la mobilité pour la traçabilité et la localisation". Journées Doctorales en Informatique et Réseaux, 2009, Belfort, France.
04. C.Baouche, A.Freitas, and M.Misson. "Routing mechanism for a DTN using WSN nodes for localization applications". IEEE International Conference on Networking, Sensing and Control, 2011, pages 514–519, Delft, The Netherlands.
05. C. Baouche, A. Freitas, and M. Misson. "Diffusion Behavior of IEEE 802.15.4 Unslotted CSMA/CA in a Cell of Proximity-Based Localization Applications". 2nd International Conference on Sensor Networks: SENSORNETS, 24-26 February, 2012, Roma, Italia.
06. C. Baouche, A. Freitas, and M. Misson. "LoWCA: Localization and Tracking Techniques Using a Wireless Sensor Network in Confined Areas - Study of the Impact of the Memory Size of Nodes". In Proceedings of the 76th Vehicular Technology Conference: VTC2012-Fall, 3-6 September, 2012, Quebec City, Canada.

## Bibliographie

1. H.Karl and A.Willig, eds. *Protocols and Architectures for Wireless Sensor Networks*. 2007, Wiley-Interscience.
2. L.Hyuk, L.Soobin, and S.Heecheol. *Wireless sensor network design for tactical military applications: remote large-scale environments*. 2009. IEEE Press.
3. S.Kumar and D.Shepherd, *SensIT: sensor information technology for the warfighter*. Information Fusion, 2001.
4. A.Zouinkhi, E.Rondeau, and M.NaceurAbdelkrim, *Ambient Intelligence: Awareness Context Application in Industrial Storage*. Journal of Wireless Sensor Network, 2011. 3: p. 134-145.
5. A.Zouinkhi, E.Rondeau, and M.NaceurAbdelkrim, *Simulation and modeling of active products cooperation for active security system management*. Transactions on Systems, Signals & Devices, 2010. 5: p. 1-23.
6. J.Hart and K.Martinez, *Environmental sensor networks : A revolution in the earth system science?* Earth-Science Reviews, 2006: p. 177-191.
7. H.Baldus, K.Klabunde, and G.Muesch, *Reliable set-up of medical body-sensor networks*. EWSN, 2004: p. 353-363.
8. P.Baronti, P.Pillai, and V.Chook, *Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards*. Computer Communications, 2007: p. 1655-1695.
9. G.Djuknic and R.Richton, *Geolocation and Assisted GPS*. IEEE Computer, 2001: p. 123-125.
10. G.Chalhoub, A.Freitas, and M.Misson, *A novel approach for simulating a bio-contamination process*. BIODEVICES, 2008: p. 123-129.
11. H.Kalita and A.Kar, *Wireless Sensor Network Security Analysis*. International Journal of Next-Generation Networks (IJNGN),, 2009.
12. B.Sundararaman, U.Buy, and A.Kshemkalyani, *Clock Synchronization in Wireless Sensor Networks: A Survey*. Ad Hoc Networks, 2005: p. 281-323.
13. S.Ganeriwal, R.Kumar, and S.Adalaha, *Network-wide Time Synchronization in Sensor Networks*. Networked and Embedded Systems Lab (NESL), 2003.
14. V.Greunen and J.Rabaey. *Lightweight Time Synchronization for Sensor Networks*. in *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*. 2003. ACM.
15. K.Lee and J.Eidson, *IEEE-1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*. In 34 th Annual Precise Time and Time Interval (PTTI) Meeting, 2002: p. 98-105.
16. J.Elson, L.Girod, and D.Estrin, *Fine-Grained Network Time Synchronization using Reference Broadcasts*. SIGOPS Oper. Syst. Rev., 2002: p. 147-163.
17. K.Romer. *Time synchronization in ad hoc networks*. in *International symposium on Mobile ad hoc networking and computing MobiHoc*. 2001. ACM.
18. M.Sichitiu and C.Veerarittiphan. *Simple Accurate Time Synchronization for Wireless Sensor Networks*. in *Wireless Communications and Networking, WCNC*. . 2003. IEEE.
19. C.Baouche, A.Freitas, and M.Misson, *Radio proximity detection in a WSN to localize mobile entities within a confined area*. Journal of Communications, 2009: p. 232-240.
20. S.Capkun, M.Hamdi, and J.Hubaux, *GPS Free Positioning in Mobile Ad-Hoc Networks*. International Conference on System Sciences, 2001. 9: p. 3481-3490.

21. A.Savvides, C.Han, and M.Strivastava. *Dynamic fine-grained localization in Ad-Hoc networks of sensors*. in *International conference on Mobile computing and networking MobiCom*. 2001. ACM.
22. A.Ward, A.Jones, and A.Hopper, *A new location technique for the active office*. *Personal Communications, IEEE* 1997. 4(5): p. 42-47
23. L.Peralta and M.Rodriguez. *Collaborative Localization in Wireless Sensor Networks*. in *International Conference on Sensor Technologies and Applications SENSORCOMM*. 2007. IEEE Computer Society.
24. P.Rong and M.Sichitiu., *Angle of Arrival Localization for Wireless Sensor Networks*. *Sensor and Ad Hoc Communications and Networks*, 2006: p. 374-382
25. H.Du and P.Lee, *Simulation of Multi-Platform Geolocation Using a Hybrid TDOA/AOA Method*, 2004.
26. A.Roxin, J.Gaber, and M.Wack, *Survey of Wireless Geolocation Techniques*. *IEEE Globecom Workshops*, 2007: p. 1-9.
27. A.Hopper, A.Harter, and T.Blackie. *The active badge system*. in *Conference on Human factors in computing systems*. 1993. ACM.
28. N.Bulusu, J.Heidemann, and D.Estrin, *GPS-less Low Cost Outdoor Localization For Very Small Devices*. *IEEE Personal Communications Magazine*, 2000. 7: p. 28-34.
29. D.Fox, W.Burgard, and F.Dellaert, *Monte Carlo Localization: Efficient Position Estimation for Mobile Robots*. *National Conference on Artificial Intelligence*, 1999: p. 343-349.
30. Q.Cao and T.Abdelzaher, *Scalable logical coordinates framework for routing in wireless sensor networks*. *ACM Trans. Sen. Netw.*, 2004. 2(4): p. 557-593.
31. K.Pahlavan, X.Li, and J.P.Makela, *Indoor geolocation science and technology*. *IEEE Communications Magazine*, 2002. 40(2): p. 112-118.
32. I.Roula, *Application des techniques d'apprentissage à la géolocalisation par radio fingerprint*, in *Laboratoire Signaux, Modèles et Apprentissage Statistique 2010*, Université Pierre et Marie Curie - Paris VI Paris. p. 121.
33. R.Dalce, L.Gui, and T.Val. *Localisation par méthodes "range-based" et "range-free" de stations mobiles communicantes dans un réseau sans fil*. in *CFIP 2011 - Colloque Francophone sur l'Ingénierie des Protocoles*. 2011. CFIP.
34. *IEEE802.15, Part 15.4 : Wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (WPANs)*. ANSI/IEEE, Standard 802.15.4 R2006, 2006.
35. Zigbee, *Zigbee Specification*. ZigBee Standards Organization, 2008.
36. ZigBeeAlliance, *The Zigbee Alliance web site*. [http ://www.zigbee.org/](http://www.zigbee.org/).
37. F.Warthman, *Delay-Tolerant Networks (DTNs) A tutorial*, W. Associates, Editor 2003.
38. T.Small and Z.J. Haas. *Resource and performance tradeoffs in delaytolerant wireless networks*. in *ACM SIGCOMM workshop on Delay-tolerant networking*. 2005. ACM.
39. W.Lvancic, W.Eddy, and S.Curiel. *Use of the delay tolerant network bundle protocol from space*. in *59th International Astronautical Congress*. 2008. Glasgow.
40. E.Krotkov and J.Blitich, *The Defense Advanced Research Projects Agency (DARPA) Tactical Mobile Robotics Program*. *The International Journal of Robotics Research*, 1999. 18(7): p. 769-776.
41. P.Juang, H.Oki, and D.Rubenstein, *Energy-efficient computing for wildlife tracking : Design tradeoffs and early experiences with zebranet*. *SIGPLAN Not.*, 2002. 37(10): p. 96-107.
42. A.Pentland, R.Fletcher, and A.Hasson, *DakNet: rethinking connectivity in developing nations*. *Computer Communications*, 2004. 37(1): p. 78-83.

43. H.Frey, S.Rhhrup, and I.Stojmenovic, *Routing in Wireless Sensor Networks*. Guide to Wireless Sensor Networks, 2009: p. 81–111.
44. A.Vahdat and D.Becker, *Epidemic routing for partially-connected ad hoc networks*. Science, 2000.
45. T.Spyropoulos, K.Psounis, and C.Raghavendra. *Spray and wait: an efficient routing scheme for intermittently connected mobile networks*. in *SIGCOMM workshop on Delay-tolerant networking*. 2005. ACM.
46. T.Spyropoulos, K.Psounis, and C.Raghavendra, *Spray and focus: Efficient mobility-assisted routing for heterogeneous and correlated mobility*, in *International Conference on Pervasive Computing and Communications Workshops2007*, IEEE Computer Society. p. 79-85.
47. T.Small and Z.Haas. *Resource and performance tradeoffs in delay-tolerant wireless networks*. in *WDTN '05*. 2005. ACM.
48. T.Small and J.Haas. *The shared wireless infostation model: a new ad hoc networking paradigm (or where there is a whale, there is a way)*. 2003. ACM.
49. J.Burgess, B.Gallagher, and B.Levine, *MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks*. International Conference on Computer Communications., 2006: p. 1-11.
50. R.Sugihara and R.Gupta, *Speed control and scheduling of data mules in sensor networks*. ACM Trans. Sen. Netw, 2010. 7(1): p. 1–29.
51. K.Harras and K.Almeroth. *Transport layer issues in delay tolerant mobile networks*. in *IFIP NETWORKING*. 2006. Springer.
52. W.Zhao, Y.Chen, and E.Zegura. *Capacity enhancement using throwboxes in DTNs*. in *International Conference on Mobile Adhoc and Sensor Systems (MASS)*. 2006. IEEE.
53. N.Banerjee, M.Corner, and B.Levine. *An Energy-Efficient Architecture for DTN Throwboxes*. in *International Conference on Computer Communications. INFOCOM 2007*. IEEE.
54. T.Camp, J.Boleng, and V.Davies, *Evaluating mobility models within an ad-hoc network*. WCMC, 2002. 2: p. 483-502.
55. D.Johnson and M.David, *Dynamic Source Routing in Ad Hoc Wireless Networks*. Mobile Computing, 1996. 353: p. 153-181.
56. D.Johnson, D.Maltz, and J.Broch. *DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks*. in *Ad Hoc Networking*. 2001. Addison-Wesley.
57. <http://www.isi.edu/nsnam/ns/>.
58. E.Royer and P.Smith. *An analysis of the optimum node density for ad hoc mobile networks*. in *International Conference on Communications*. 2001. IEEE.
59. L.Blazevic, J.LeBoudec, and S.Giordano, *A Location-Based Routing Method for Mobile Ad Hoc Networks*. IEEE Trans. Mob. Comput, 2005. 4: p. 97-110.
60. Z.Haas, *A new routing protocol for the reconfigurable wireless networks*. IEEE 6th International Conference on Universal Personal Communications Record, 1997. 2: p. 562-566.
61. C.Barrett, A.Marathe, and M.Marathe. *Characterizing the Interaction Between Routing and MAC Protocols in Ad-hoc Networks*. 2002. ACM.
62. A.Krifa, *An optimal joint scheduling and drop policy for Delay Tolerant Networks*. International Symposium on a World of Wireless, Mobile and Multimedia Networks, 2008 p. 1-6
63. S.Dalu, M.Naskar, and C.Sarkar, *Implementation of a Topology Control Algorithm for MANETs Using Nomadic Community Mobility Model* Third international Conference on Industrial and Information Systems, 2008: p. 1-5.

64. J.Scott, P.Hui, and C.Diot. *Haggle: a Networking Architecture Designed Around Mobile Users*. in *Third Annual Conference on Wireless On-demand Network Systems and Services*. 2006.
65. A.Keranen, T.Karkkainen, and T.Ott, *Simulating Mobility and DTNs with the ONE*. *Journal of Communications (JCM)*, 2010. 5(2): p. 92–105.
66. H.Hong-Yu, L.Pei-En, and W.Min-You, *Performance Evaluation of SUVnet With Real-Time Traffic Data*. *IEEE Transactions on Vehicular Technology*, 2007. 56(6): p. 3381-3396.
67. Y.Pigne, G.Danoy, and P.Bouvry. *A vehicular mobility model based on real traffic counting data*. in *Third international conference on Communication technologies for vehicles*. 2011. Springer-Verlag.
68. J.Bernard, M.Zhang, and A.Chowdury, *Twitter power: Tweets as electronic word of mouth*. *JASIST*, 2009. 60: p. 2169-2188.
69. D.Macii, F.Trenti, and P.Pivato, *A robust wireless proximity detection technique based on RSS and ToF measurements*. *CORD Conference Proceedings*, 2011: p. 31-36.
70. C.Baouche, A.Freitas, and M.Misson. *Routing mechanism for a DTN using WSN nodes for localization applications*. in *ICNSC : Proceedings of the IEEE International Conference on Networking, Sensing and Control, ICNSC 2011, Delft*. 2011. IEEE.
71. P.Juang, H.Oki, and D.Rubenstein, *Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet*. *SIGPLAN Not*, 2002. 37: p. 96-107.
72. P.Gastmeier, D.Sohr, and C.Geffers, *Risk factors for death due to nosocomial infection in intensive care unit patients: findings from the Krankenhaus Infektions Surveillance System*. *Infection Control and Hospital Epidemiology: The Official Journal of the Society of Hospital Epidemiologists of America*, 2007. 28(4): p. 466-472.
73. J.Rumbaugh, I.Jacobson, and G.Booch, *Unified Modeling Language Reference Manual, The (2nd Edition) (Addison-Wesley Object Technology Series)*2004: Addison-Wesley Professional.
74. C.Baouche, A.Freitas, and M.Misson, *Architecture d'un réseau sans fil exploitant la mobilité pour la traçabilité et la localisation*. *Journées Doctorales en Informatique et Réseaux*, Belfort, France, 2009.
75. H.Belmonte and M.Misson, *Étude, implémentation et évaluation d'un réseau de capteurs sans fil exploitant le concept de colportage de l'information*, in *CNAM2011*, CNAM: Centre Régional Associé de Clermont-Ferrand. p. 166.
76. *The Network Simulator ns-2*. Available from: <http://www.isi.edu/nsnam/index.php>.
77. T.Friis. *A note on a simple transmission formula*. in *Proceedings of the IRE*. 1946.
78. Friis, T., *A Note on a Simple Transmission Formula* *Proceedings of the IRE*, 1946. 41(5): p. 254-256.
79. T.Rappaport, *Wireless communications : principles and practice*2001: Prentice Hall PTR.
80. ITU, *Propagation data and prediction method for the planning of indoor radio communication systems and local area networks in the frequency range of 900 MHz to 100 GHz*. Recommendation ITU-R P 1238-4, 2005.
81. S.Benferhat, F.Jacquet, and M.Misson, *Study of traffic conditions for a fleet of mobiles moving within a wireless cell in an industrial environment*. *Mosharaka International Conference on Communications, Computers and Applications*, 2008: p. 92-97
82. C.Baouche, A.Freitas, and M.Misson, *A WPAN model to locate mobile entities in a mine gallery*. *International Conference on Wireless Communications in Underground and Confined Areas*, Val-d'Or, Canada, 2008: p. 183-187.

83. G.Bianchi, *Performances Analysis of the IEEE 802.11 Distributed Coordination Function*. IEEE Journal on Selected Areas in Communications, 2000. 18(3): p. 535-547.
84. C.Baouche, A.Freitas, and M.Misson, *Diffusion Behavior of IEEE 802.15.4 Unslotted CSMA/CA in a Cell of Proximity-Based Localization Applications*. International Conference on Sensor Networks: SENSORNETS, Roma, Italia, 2012.
85. C.Holloway, D.Hill, and G.Hufford, *Radio wave propagation characteristics in lossy circular waveguides such as tunnels, mine shafts, and boreholes* IEEE Transactions on Antennas and Propagation, 2000. 48(9): p. 1354-1366
86. T.Sheltami, E.Shakshuki, and H.Mouftah. *Performance evaluation of TELOSB sensor network*. in *7th International Conference on Advances in Mobile Computing and Multimedia*. 2009. ACM.
87. C.Baouche, A.Freitas, and M.Misson, *LoWCA: Localization and Tracking Techniques Using a Wireless Sensor Network in Confined Areas - Study of the Impact of the Memory Size of Nodes*. 76th Vehicular Technology Conference: VTC2012-Fall, Quebec City, Canada, 2012.
88. S.John and J.Shana. *Distributed sensory systems and developer platforms from Crossbow technology*. in *Proceedings of the 3rd international conference on Embedded networked sensor systems*. 2005. ACM.
89. P.Levis, S.Madden, and D.Culler, *TinyOS: An Operating System for Sensor Networks*, ed. W.a.R. Weber, Jan M. and Aarts, Emile. Vol. 7. 2005: Springer-Verlag.