



UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE
ÉCOLE DOCTORALE SCIENCES TECHNOLOGIE SANTÉ (547)

THÈSE

pour obtenir le grade de
DOCTEUR DE L'UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE

Discipline : Génie informatique, automatique et traitement du signal

présentée et soutenue publiquement

par

Mengyi ZHANG

le 19 mai 2015

Titre :

**Optimisation de la couverture de communications
et de mesure dans les réseaux de capteurs**

JURY

M. Jean-Yves TOURNERET	Professeur des Universités	Président
M. Guillaume GELLÉ	Professeur des Universités	Directeur de Thèse
M. Hichem SNOUSSI	Professeur des Universités	Rapporteur
M. David BRIE	Professeur des Universités	Rapporteur
M. Alban GOUPIL	Maître de Conférences	Co-encadrant
M. Maxime COLAS	Maître de Conférences	Co-encadrant

*À toute ma famille
À tous mes amis
À toute personne
Qui m'a aidé à franchir un horizon dans ma vie...*

Remerciements

Ce mémoire marque la fin de trois années de préparation de thèse de doctorat au sein du laboratoire Centre de Recherche en Sciences et Techniques de l'Information et de Communication (CReSTIC) de l'Université de Reims Champagne-Ardenne.

Je tiens dans un premier temps à remercier vivement mon directeur de thèse, Guillaume GELLÉ et mes co-encadrants, Alban GOUPIL et Maxime COLAS, pour m'avoir encadrée au cours de ces trois années. Je ne saurais assez les remercier pour leur disponibilité, leur soutien indéfectible, leur enthousiasme scientifique et leurs encouragements qui m'ont permis de mener à bien mes travaux, surtout dans les moments les plus difficiles. Je remercie particulièrement Alban GOUPIL qui a investi beaucoup de temps et d'efforts dans ce manuscrit. Merci beaucoup à vous. Pendant ces trois années de thèse, j'ai beaucoup appris et progressé, et c'est en partie grâce à vous. J'ai eu la chance et le plaisir de travailler sous votre direction.

J'adresse mes remerciements à M. Hichem SNOUSSI et M. David BRIE pour avoir accepté d'être rapporteurs de mes travaux et membres du jury, je remercie également M. Jean-Yves TOURNERET qui a accepté d'évaluer ce travail en tant que président du jury.

Je souhaite également remercier tous les membres de l'équipe CReSTIC pour leur accueil, leur sympathie ainsi que leurs idées constructives. Un grand merci à la secrétaire du CReSTIC, Ida LENCLUME ainsi qu'à la secrétaire de l'école doctorale, Estelle ODINOT.

Je voudrais remercier très sincèrement tous mes collègues, en particulier Anas HANAF et Mohamed ABDELMOULA, et tous mes amis, en particulier Hui SHANG et Yugang LI avec qui j'ai partagé une excellente ambiance de recherche et d'enseignement.

Mes derniers remerciements sont pour ma famille. Un énorme merci à ma famille pour m'avoir toujours laissé libre de faire ce qui me plaisait et pour m'avoir encouragée dans mes choix.

Résumé

Un réseau de capteurs sans fil résulte du déploiement d'un ensemble de petites unités autonomes interagissant via un réseau construit grâce à leur module de communication qui observent leur environnement par des capteurs pour ensuite traiter et/ou sauvegarder cette information via leur capacité calculatoire et de stockage.

La couverture est la seule représentation disponible aux réseaux de capteurs de l'espace physique environnant. Par conséquent, il est essentiel de pouvoir qualifier et quantifier sa qualité notamment concernant la présence de trous.

Nos travaux utilisent la topologie algébrique pour répondre à ces problèmes. Plus précisément, nous définissons dans un premier temps une notion de trou de couverture d'un champ scalaire qui mesure la qualité de l'estimation par le réseau de capteurs sans pour autant connaître la position des capteurs. Cela permet d'utiliser l'homologie simpliciale pour déterminer la qualité de la couverture globale et accessoirement de mettre en veille certains capteurs surnuméraires tout en garantissant la couverture.

Puis, afin de rendre le résultat précédent facilement calculable par un réseau de capteurs grâce à une distribution du calcul qui supporte en plus le passage à l'échelle, nous utilisons la théorie de Morse discrète pour faire le calcul des groupes d'homologie nécessaires à notre application précédente.

Enfin, cette dernière approche est rendue suffisamment souple pour permettre le suivi temporel des modifications de la couverture de manière délocalisée. Cela permet non seulement de suivre la qualité de la couverture lorsque l'environnement se modifie mais aussi de proposer un schéma distribué de mise en veille des capteurs afin d'augmenter la durée de vie du réseau de capteurs tout en garantissant une couverture suffisante.

Mots clefs réseaux de capteurs, couverture, topologie algébrique, homologie

Abstract

A wireless sensor network consists of a set of small autonomous units that interact via a network built by their communication modules. They observe their environment by their sensors and then they manage this information according to their computational capacity and storage.

The coverage is the only representation available to the sensor network of its environment. Therefore, it is essential to quantify the quality of coverage especially related to the presence of holes.

Our work uses algebraic topology to solve these problems. We first define a notion of the coverage hole in a scalar field, which measures the quality of the estimation by the sensor network without knowing the positions of the sensors. It allows the simplicial homology tool to determine the quality of the overall coverage and put certain redundant sensors into sleeping mode with the guarantee of the coverage.

Then, to make the previous result easier to compute by a sensor network, the discrete Morse theory is used. It allows a distributed computation of the previous homology groups while supporting scalability necessary in sensor networks domain.

Finally, one flexible approach that allows time varying tracking which allows a coverage is proposed in a distributed way. When the environment changes, this approach can not only guarantee the capability of monitoring of coverage quality, but also proposes a scheme to send to sleep the redundant sensors in order to increase the lifetime of the sensor network with adequate coverage.

Keywords sensor networks, coverage, algebraic topology, homology

Table des matières

Résumé	vii
Abstract	ix
1 Introduction	1
1.1 Contexte et problématique	1
1.2 Contributions de la thèse	3
1.3 Structure de ce document	4
2 Les réseaux de capteurs sans fil	5
2.1 Introduction aux réseaux de capteurs sans fil	5
2.1.1 Principes élémentaires	6
2.1.2 Quelques applications des RCSF	8
2.1.3 Caractéristiques	11
2.1.4 Critères de conception	13
2.2 Couverture et connectivité	15
2.2.1 Connectivité	16
2.2.2 Couverture	17
2.2.2.1 La modélisation de la couverture	18
2.2.2.2 Différentes notions de couverture	19
2.3 Conclusion	22
3 Homologie et couverture dans les RCSF	25
3.1 Notion de topologie et de topologie algébrique	26
3.1.1 Représentation des espaces	27
3.1.2 Linéarisation	28
3.1.3 Groupes d'homologie	30
3.1.3.1 Le groupe H_0	31
3.1.3.2 Le groupe H_1	31
3.1.3.3 Les groupes $H_k(X)$	32
3.1.3.4 Les groupes d'homologie relative $H_k(X, A)$	33
3.1.3.5 Méthode de calcul des groupes d'homologie	33
3.1.3.6 Exemple complet de calcul de groupes d'homologie	36
3.1.4 Persistance homologique	40

3.1.5	L'homologie au secours des RCSF	43
3.1.5.1	La construction de de Silva et de Ghrist	43
3.1.5.2	Application simple à la gestion de l'énergie d'un RCSF	45
3.2	Éléments sur les processus gaussiens	47
3.2.1	Définition	48
3.2.2	Régression	50
3.2.3	Calcul des hyper-paramètres	52
3.2.4	Exemple de régression d'un champ déterministe	53
3.3	Application à la couverture	55
3.3.1	Description de l'application	55
3.3.2	Construction du complexe simplicial	57
3.3.3	Détection des trous de couverture	60
3.3.4	Utilisation de la persistance	60
3.3.5	Mise en veille de capteurs	62
3.3.6	Remarques sur l'algorithme	63
3.3.6.1	Distribution de l'algorithme	63
3.3.6.2	Optimisation des cycles	64
3.4	Conclusion	65
4	Théorie de Morse et RCSF	67
4.1	La théorie de Morse et sa version discrète	68
4.1.1	Version différentielle de la théorie de Morse	68
4.1.2	Historique et applications de la théorie de Morse discrète	71
4.1.3	Champ discret de vecteurs combinatoire	71
4.1.4	Le flot discret d'un champ de vecteur	73
4.1.5	Construction du complexe de Morse	74
4.1.6	Exemple de l'anneau	75
4.1.7	La fonction potentielle	76
4.1.8	Simplification du complexe de Morse	77
4.1.9	Construction aléatoire du champ discret de vecteur	80
4.2	Algorithme proposé pour le suivi des groupes d'homologie	81
4.2.1	Principe de l'algorithme	81
4.2.2	Mise à jour incrémentale	82
4.2.2.1	Mise à jour en cas d'insertion d'une cellule	83
4.2.2.2	Mise à jour en cas de suppression d'une cellule	84
4.2.2.3	Dégénérescence du champ de vecteurs	87
4.2.3	Comparaison avec d'autres algorithmes	87
4.3	Application à la couverture de mesure	88
4.3.1	Description de l'application	88
4.3.2	Résultats des simulations	89
4.4	Application à la gestion de l'énergie	92
4.4.1	Description de l'application	92
4.4.2	Cadre général des simulations	93
4.4.3	Sélection des capteurs éveillés	93

4.4.3.1	Première méthode	95
4.4.3.2	Seconde méthode	99
4.4.3.3	Comparaison des deux méthodes	101
4.4.4	Suivi de la couverture et activation automatique des capteurs	102
4.4.5	Améliorations de l'algorithme	105
4.4.6	Gestion de la couverture sans garantie	107
4.5	Conclusion	107
5	Conclusion et perspectives	111
5.1	Conclusion générale	111
5.2	Perspectives	112
	Bibliographie	115

Table des figures

2.1	Schéma de la mise en réseaux des RCSF	7
2.2	Synopsis d'un capteur	7
2.3	Exemples de réalisation de capteurs	8
2.4	Couverture ponctuelle	20
2.5	Couverture de zone	21
2.6	Couverture de frontière	22
2.7	Couverture glissante	23
3.1	Exemple d'espace combinatoire	27
3.2	Exemple d'une chaîne et de sa bordure	29
3.3	Exemple de chaînes	31
3.4	Les espaces et sous-espaces vectoriels d'un complexe	33
3.5	Exemple de surface à deux trous	36
3.6	Exemple de données pour la persistance	41
3.7	Filtration du complexe pour les distances 0.6, 1.8 et 2	41
3.8	Complexe d'un RCSF selon la méthode de de Silva et Ghrist	43
3.9	Extinction de capteurs avec couverture certifiée	45
3.10	Instantanés des capteurs éteints et éveillés	46
3.11	Exemples de réalisation de processus gaussiens unidimensionnels	50
3.12	Régression par les processus gaussiens unidimensionnels	51
3.13	Impact des hyper-paramètres sur la regression	52
3.14	Régression d'un champ déterministe par un processus gaussien	53
3.15	Régression d'un champ déterministe par un processus gaussien (contour)	54
3.16	Variance de l'estimation du champ	54
3.17	Le champ et la régression de l'exemple	55
3.18	Variance de l'estimation et position des capteurs	56
3.19	Contraintes sur le domaine	58
3.20	Complexe simplicial pour les seuils 7.0, 7.5 et 7.6	59
3.21	Cycles générateurs de H_1 pour les seuils 7.0, 7.5 et 7.6	60
3.22	Code-barres des groupes d'homologie H_0 en bas et H_1 en haut	61
3.23	Mise en veille de capteurs	62
3.24	Structure de donnée distribuée pour représenter un complexe	63
4.1	Déformation de l'anneau	69

4.2	Fonction de Morse et son complexe unidimensionnel	69
4.3	Fonction de Morse et son complexe bidimensionnel	70
4.4	Champ de vecteur discret pour un complexe simplicial	72
4.5	Flot d'un champ discret de vecteurs	73
4.6	Complexe de Morse discret associé et sa version simplifiée	74
4.7	Exemple d'un champ de vecteur sur l'anneau et son complexe de Morse associé	75
4.8	Exemple de Forman de simplification de Morse	77
4.9	Exemple de simplification de Morse d'un complexe homologue à un cerlce .	78
4.10	Exemple de l'échec d'une simplification de Morse abusive d'un complexe homologue à un cerlce	79
4.11	Étapes de la mise à jour lors d'ajout de cellules	83
4.12	Mise à jour lors de l'ajout d'un ensemble de cellules	84
4.13	Étapes de la mise à jour lors de retraits de cellules	85
4.14	Mise à jour lors du retrait d'un ensemble de cellules	86
4.15	Début de la construction du complexe	90
4.16	Instantanés de la construction du complexe représentant la couverture . . .	91
4.17	Couverture initiale	94
4.18	Construction de la couverture par la première méthode	97
4.19	Construction de la couverture par la seconde méthode	100
4.20	Configuration finale des capteurs selon les deux méthodes	102
4.21	Instantanés des capteurs éteints et éveillés	104
4.22	Comparaison entre l'algorithme de base, à gauche, et la version corrigée par la simplification, à droite	106
4.23	Instantanés des capteurs éteints et éveillés sans couverture garantie	108

Chapitre 1

Introduction

1.1	Contexte et problématique	1
1.2	Contributions de la thèse	3
1.3	Structure de ce document	4

1.1 Contexte et problématique

Les réseaux de capteurs sans fil (RCSF), ou Wireless Sensor Networks (WSN) en anglais, soulèvent de nouvelles problématiques en traitement du signal. En effet, les algorithmes doivent prendre en compte les spécificités de tels réseaux, comme la distribution des calculs, la gestion de l'énergie notamment pour le coût des communications [ZG04, ZST06].

Ces réseaux de capteurs sont constitués d'un ensemble de nœuds captant leur environnement par des mesures et qui ont la capacité de les traiter localement et d'échanger des informations avec leurs voisins [FSR09, DP10]. Ainsi les RCSF permettent d'obtenir de la diversité dans les mesures aussi bien spatiale que temporelle. Cette diversité permet d'une part la couverture d'un domaine de captation plus vaste et d'autre part, grâce aux corrélations entre celles-ci, la précision donnée par l'ensemble des capteurs est plus grande que la précision de chaque mesure prise séparément.

Bien que les mises en œuvre des RCSF soient fortement liées aux applications [MT10], plusieurs traitements de bases sont suffisamment génériques pour construire un socle commun [ZG04]. Nous comptons parmi ces traitements la gestion de la couverture et la reconstruction du champ mesuré à partir des échantillons peu précis fournis par les capteurs.

Sur ce dernier point, plusieurs approches sont possibles pour généraliser le théorème d'échantillonnage de Shannon-Nyquist à plusieurs dimension. De plus, les capteurs ne sont pas régulièrement disposés sur le domaine et par conséquent l'échantillonnage est non uniforme. Une reconstruction par la théorie de l'information est envisageable [KIR11] qui permet de placer les capteurs si ces derniers peuvent l'être. Une seconde solution

consiste à construire une approximation grâce à un espace de Hilbert et une fonction noyau bien choisis [RM09] ou encore une régression par un filtrage linéaire peut être suffisante [NCV08a, NCV08b].

Toutefois ces différentes approches sont ici limitées par la nécessité de connaître parfaitement la position des capteurs. Or cette dernière est rarement disponible si le RCSF est déployé en intérieur ou si les balises GPS ne sont pas sur les nœuds. La recherche des positions des capteurs peut donc se révéler délicate [PAK⁺05]. Une modélisation du champ mesuré par un processus gaussien [AT07] permet de répondre partiellement à cette difficulté. En effet, les corrélations entre les nœuds sont parfois suffisantes pour faire une régression et donc reconstruire le champ aux points d'intérêt [RW05].

Concernant la couverture, selon l'application, le domaine couvert par le réseau de capteurs ne doit pas contenir de trous de couverture. Plus précisément [MKT13], les points d'intérêt doivent être couverts par le RCSF — problème du « point coverage » — ou une zone complète doit être surveillée — problème de l'« area coverage » — ou encore, le RCSF doit s'assurer que rien ne traverse une barrière fixée — problème du « barrier coverage ».

Il est étonnant de constater que les solutions aux problèmes de couverture utilisent des modèles de captation très simples tels que celui d'un graphe de disque de rayon préfixé. Un traitement plus adéquat de ce problème est nécessaire en prenant en compte les spécificités du champ de mesure. Nous pouvons toutefois citer les travaux [WXX12] pour la sélection de capteurs dans un champ gaussien.

Les mesures réalisées par les capteurs sont nécessairement localisées mais l'information intéressante pour l'application nécessite la plupart du temps une intégration de toutes ces mesures locales. Il s'agit là du leitmotiv de la topologie qui est la boîte à outils mathématiques qui permet de passer d'un niveau local à une information globale.

La topologie [Arm83, Cro05, HY61] est souvent considérée comme une théorie difficile et très abstraite mais elle commence à trouver un nombre important d'applications dans un grand nombre de domaines scientifiques [AF07, Bas06, Ghr14, KMM03] et notamment en analyse de donnée [Car09].

Par conséquent, le choix de cette théorie comme outil pour le traitement des mesures est naturelle. Toutefois, il faut pour être utilisable, que les données, c'est-à-dire les espaces topologiques, soient représentables de façon combinatoire pour pouvoir être manipulés par des algorithmes et que de plus les calculs soient aussi réalisables par une machine.

La topologie algébrique [Mas91, Hat01, Gib10] permet de répondre à ces demandes car elle utilise les outils d'algèbre linéaire, et se réduit souvent à des calculs matriciels *in fine*. Nous voyons actuellement apparaître un nouveau champ de la topologie, la topologie algorithmique [EH10, Koz08, Zom05] dont l'article [Ghr08b] présente rapidement quelques applications.

1.2 Contributions de la thèse

Nos travaux ont porté dans un premier temps sur les problèmes de couverture en prenant en compte le modèle de captation. De plus la localisation des capteurs n'y est pas nécessaire. Nous avons en effet utilisé la modélisation par un processus gaussien qui repose uniquement sur les corrélations entre les mesures des capteurs.

Ainsi, un trou de couverture correspond à un endroit du domaine de captation où la qualité de l'estimateur est jugée insuffisante. Pour découvrir de tels trous de couverture, nous avons construit un espace combinatoire abstrait de ce problème de couverture pour permettre l'utilisation de l'algèbre topologique, et principalement le calcul des groupes d'homologie pour la découverte effective de ces trous. Cette solution est partiellement basée sur les travaux de de Silva et Ghrist [dSG07a, dSG07b, dSG06] pour que la position des capteurs ne soient pas nécessaire, car les autres méthodes reposent sur cette position et utilisent des algorithmes géométriques.

Ces résultats ont été présentés à la conférence internationale IEEE Wireless Communications and Signal Processing (WCSP) en 2013 [ZGC13].

La distribution du calcul de notre solution reste cependant lourde bien que possible en utilisant une approche telle que [DGJ12]. De plus le suivi de l'évolution du champ de mesure et des trous reste difficile car les calculs doivent être repris périodiquement.

Ainsi, nous avons utilisé la théorie de Morse discrète développée par Forman [For02b] pour rendre l'algorithme facilement utilisable pour les réseaux de capteurs. En effet les données représentant l'espace combinatoire de notre solution peut être facilement partagées entre les capteurs et les calculs effectués par les capteurs restent majoritairement localisés à leur voisinage.

Ainsi, nous avons développé un algorithme qui permet le suivi des caractéristiques topologique d'un espace qui est particulièrement simple et qui passe bien à l'échelle grâce à la localisation des calculs. Cet algorithme a été appliqué à notre solution de découverte de trous de couverture et présenté à WCSP en 2014 [ZGC14].

Notre algorithme est assez souple que nous ayons pu le modifier pour intégrer une gestion efficace de l'énergie du réseau de capteurs en mettant en sommeil un certain nombre de capteurs et en gardant d'autres éveillés tout en garantissant la couverture. Les capacités de suivi de notre algorithme permet de gérer les modifications dues à la disparition de capteurs qui manque d'énergie. Ainsi, la durée de vie du réseau sera améliorée. Cette application avec les modifications de notre algorithme seront l'objet d'une publication future.

1.3 Structure de ce document

Dans le chapitre suivant « **Les réseaux de capteurs sans fil** », nous proposons une description plus détaillée des réseaux de capteurs sans fil avec leurs caractéristiques et quelques exemples d'applications. Nous y développerons aussi les notions de couverture et de connectivité.

Le chapitre « **Homologie et couverture dans les RCSF** » présente notre première contribution. Pour cela, quelques notions de topologie et surtout de topologie algébrique y seront présentées notamment par un exemple numérique. Le critère de couverture de de Silva et de Ghrist clôture cette présentation par son application à un RCSF simple.

La suite de ce chapitre concerne les processus gaussiens et leur utilisation pour la régression. Une fois terminées, ces deux bases serviront de support au développement de notre contribution à la fin du chapitre, à savoir la création d'un critère pour la détection des trous de couverture pour un champ de mesure modélisé par un processus gaussien.

La présentation de la théorie de Morse et de sa version discrète débute le chapitre « **Théorie de Morse et RCSF**. » Cette théorie servira de base au développement de notre algorithme de suivi de groupe d'homologie qui sera l'objet de la section suivante.

Enfin, deux applications termineront ce chapitre. La première reprend la simulation du chapitre précédent mais en y incluant la notion de variation temporelle. La gestion intelligente de l'énergie sera l'objet de la seconde application pour laquelle une sélection des capteurs à maintenir éveillés est effectuée.

Le dernier chapitre sera la conclusion qui présente aussi un certain nombre de perspectives qui sont issues d'une part des développements de nos travaux mais aussi de l'utilisation d'autres aspects de la topologie, notamment différentielle et des faisceaux, pour intégrer les données issues des capteurs d'un réseau.

Chapitre 2

Les réseaux de capteurs sans fil

2.1	Introduction aux réseaux de capteurs sans fil	5
2.1.1	Principes élémentaires	6
2.1.2	Quelques applications des RCSF	8
2.1.3	Caractéristiques	11
2.1.4	Critères de conception	13
2.2	Couverture et connectivité	15
2.2.1	Connectivité	16
2.2.2	Couverture	17
2.3	Conclusion	22

Nos travaux s'intéressent principalement à la couverture dans les réseaux de capteurs sans fil au sens général. C'est pourquoi ce chapitre tente de faire un rapide état des lieux des RCSF et de la problématique de la couverture.

Pour ce faire, la section 2.1 présente les réseaux de capteurs sans fil, leurs caractéristiques, leurs structures et quelques applications. Ensuite, la section 2.2 suivante s'intéresse et présente plus précisément aux différents problèmes de couverture. Enfin, la section 2.3 conclura ce chapitre.

2.1 Introduction aux réseaux de capteurs sans fil

L'émergence, ces dernières années, des réseaux de capteurs sans fil a eu un impact important en science et en technologie. Ce sujet a attiré l'attention des secteurs industriels et militaires et reste l'objet d'intenses recherches académiques.

Contrairement à une technologie réseau traditionnelle, les RCSF intègrent seulement des unités limitées dotées d'une unité de communication sans fil et d'un module sensoriel.

De part leurs aspects génériques, les champs d'application des RCSF sont très variés et demandent souvent une application *ad hoc*.

On peut fixer l'acte de naissance des réseaux de capteurs sans fil en 1978 à l'Université Carnegie Mellon. En effet, un groupe financé par la DRAPA (Defense Advanced Research Projects Agency) y travaille sur un réseau de capteurs distribués (Distributed Sensor Nets Workshop) [Car78]. Ces travaux avaient pour objectif la mise en place d'un système distribué de surveillance militaire [Hew01]. Mais en raison des contraintes techniques encore non résolues à cette époque, l'application reste très limitée.

Depuis, la technologie des capteurs s'est considérablement améliorée, la conception des nœuds du réseau est maintenant bien maîtrisée. Cela permet aux actuelles recherches académiques et industrielles de se concentrer sur les aspects réseaux mais aussi sur le traitement du signal distribué.

Vers les années 1990–2000 les RCSF se sont fortement développés au niveau académique en voyant apparaître plusieurs applications. Par exemple, l'Université de Californie, Los Angeles, a développé une architecture de RCSF nommée WINS [PK00]; de même Berkeley avait son programme PicoRadio [RAdS⁺00] ou Smart DUST [KSJPB99] et le MIT le projet μ AMPS [Cha]. La création de groupe de travail pour des communications numériques sans fil pour des réseaux avec des nœuds peu puissant a permis l'émergence de la pile ZigBee basée sur la couche normalisée IEEE 802.15.4.

2.1.1 Principes élémentaires

Un réseau de capteurs est la mise en réseau de petites unités choisies de façon *ad hoc* selon l'application. Cette mise en réseau suit souvent le schéma de la figure 2.1 où les nœuds peuvent communiquer entre eux sans fil. Chaque nœud mesure l'environnement dans son voisinage. Ces nœuds peuvent alors faire un traitement localisé ou distribué. Mais de proche en proche, les informations pertinentes sont remontées à une station de base pour être gérées soit localement soit à distance via une connexion à réseau plus classique, type Internet.

Pour pouvoir accomplir la tâche qui leur incombe, les petites unités sont essentiellement composées de quatre parties, comme le montre la figure 2.2,

- une unité sensorielle qui capte l'état de l'environnement immédiat. Elle mesure par exemple la température, la luminosité, l'humidité, la présence gaz, l'intrusion, la moisissure et bien d'autres paramètres encore. Elle fournit l'interface entre le monde physique et l'application ;
- un émetteur-récepteur qui permet de communiquer à courte distance avec le reste du réseau ;
- une unité de traitement, souvent un micro-contrôleur avec une petite mémoire de stockage, qui traite l'information issue des capteurs et du réseau pour accomplir l'objectif du dispositif. Cette partie est aussi en charge de la gestion du réseau au niveau des couches OSI (Open Systems Interconnection) les plus hautes ;

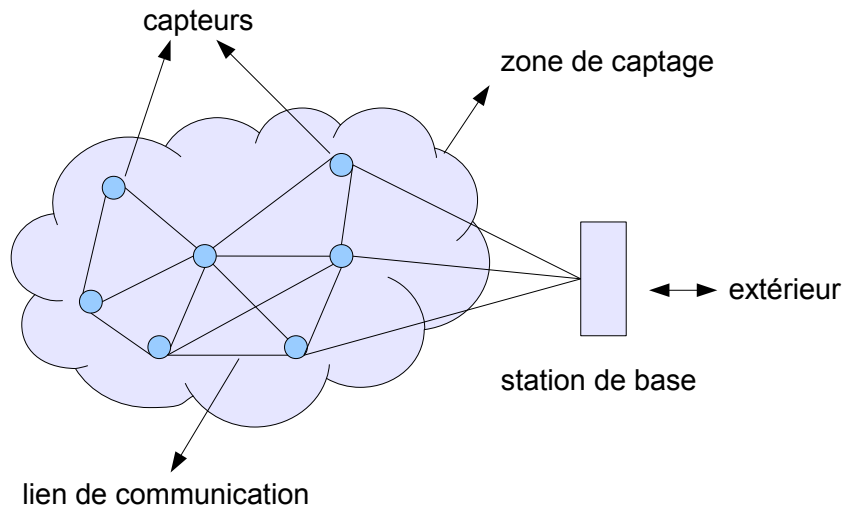


FIGURE 2.1 – Schéma de la mise en réseaux des RCSF

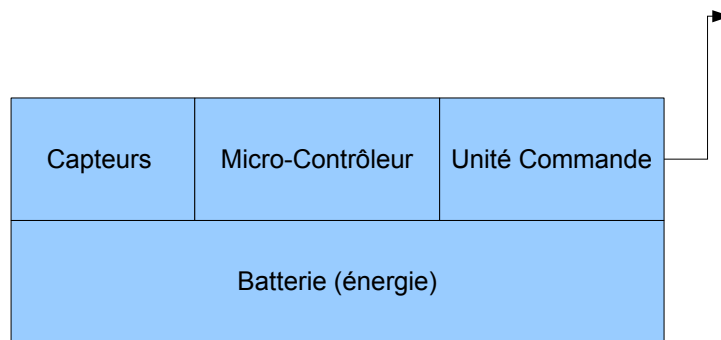


FIGURE 2.2 – Synopsis d'un capteur

- une unité de gestion de l'énergie, supportée la plupart du temps par une batterie et parfois par une unité de production électrique.

Souvent les capteurs sont petits et peu onéreux grâce à leur fabrication à grande échelle. Le coût de leur déploiement en un RCSF dense diminue en conséquence. Cependant pour des raisons de coût unitaire, les ressources énergétiques, de stockage, de calcul, et de communication pour chaque capteur sont très limitées.

La figure 2.3 présente quelques réalisations de capteurs qui deviennent de plus en plus petit. Il existe même des capteurs entièrement intégrés dans une seule puce — System On Chip.

Les RCSF peuvent être facilement déployés dans un grand nombre de situations et d'environnements spécifiques tels qu'à l'intérieur de bâtiments, en zones urbaines ou encore en environnements hostiles où le câblage d'un réseau est difficile voire impossible.

Comme le dispositif de transmission sans fil miniaturisé doit avoir une faible consommation énergétique, la structure de réseau des RCSF devient primordiale pour pouvoir remonter l'information à l'utilisateur final et pour permettre la réalisation de calculs dis-

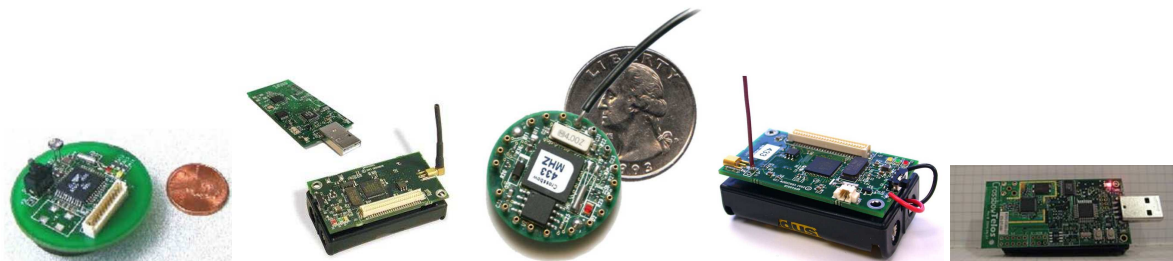


FIGURE 2.3 – Exemples de réalisation de capteurs

tribués à l'intérieur du réseau.

Cette mise en réseau permet la collaboration entre les différentes unités sensorielles grâce aux échanges d'information permises par le réseau. Ce changement de point de vue permet de suppléer la qualité des capteurs par leur quantité ; au traitement du signal de gérer la mauvaise qualité des mesures pour atteindre l'objectif grâce à la grande diversité et à la spatialisation de l'information que permet les RCSF.

2.1.2 Quelques applications des RCSF

La large gamme des capteurs sensoriels — thermique, mécanique, optique, vibrations, etc — couplée à une taille de plus en plus miniaturisée et à la possibilité de mise en commun d'information via un réseau ad-hoc permet aux réseaux de capteurs d'envahir de nouveaux domaines d'applications tels que le domaine militaire, environnemental, santé, sécurité et commercial. D'autres catégories peuvent être considérées telles que l'exploitation de l'espace, le traitement chimique et le contrôle des désastres [YMG08].

De plus, la mise en réseaux de informations localisées spatialement, permet l'apparition d'applications des réseaux de capteurs sans fil, que ce soit dans les domaines militaire et policier contre le terrorisme ou pour le contrôle des émeutes que dans les domaines civils comme pour la surveillance de l'environnement ou le suivi des personnes à domicile [ASS02, RLS04]. Ces applications ne peuvent pas être réalisées par un système unique central même puissant.

Nous faisons dans cette section un rapide survol d'applications des RCSF selon leur domaine.

Domaine environnemental De part ses possibilités de couverture d'une large surface, les RCSF sont naturellement employés à des fins d'observation de l'environnement. De plus, la protection de l'environnement et sa surveillance sont des questions de plus en plus importantes dans nos sociétés. Les RCSF permettent de recueillir des données chiffrées sur l'évolution de l'environnement et l'impact de différents facteurs environnementaux.

La surveillance des glaciers et leur évolution annuelle ont été par exemple l'objet d'études

par l'EPFL [IBS⁺10] ou par d'autres équipes [MBE10] afin d'en analyser l'évolution et par corrélation suivre l'évolution climatique d'une région.

Hormis les aspects écologique, la surveillance de l'environnement peut aussi prévenir et aider les secours en cas de catastrophes naturelles comme des éruptions volcaniques. Les réseaux de capteurs sont alors appropriés pour cette surveillance [WALW⁺06]. De même, les RCSF permettent de réduire considérablement les coûts des systèmes d'alarme préventive pour les tremblements de terre [NR11]. Il en est de même des réseaux de surveillance des feux de forêt [YWM05] qui est l'archétype de l'application des réseaux de capteurs.

La météorologie et la climatologie d'une région sont aussi sujettes à la mise en réseau de capteurs. Par exemple, le système ALERT [AUG] est un premier pas vers les RCSF pour la surveillance du niveau des eaux en relation avec la météorologie et d'autres mesures environnementales.

La surveillance de l'habitat des animaux fournit aussi beaucoup d'applications aux RCSF. Par exemple, l'article [PSM⁺04], voir aussi [Kum04], décrit un réseau de capteurs dédié à l'étude environnementale d'une petite île du Maine. Les capteurs mesurent l'humidité, la luminosité, la température, la pression atmosphérique ainsi que l'occupation des terriers de pétrel.

Les RCSF servent aussi à contrôler la qualité des sols pour les animaux [METS⁺06]. Naturellement, ce contrôle peut s'étendre pour créer le domaine de l'agriculture intelligente dont l'impact écologique se réduit et qui ne consomme que le nécessaire.

Les réseaux de capteurs sont aussi efficaces pour suivre l'évolution de bâtiment comme les torsions des ponts notamment le golden bridge [KPC⁺07] de San Francisco par un RCSF développé à Berkeley. La qualité des infrastructures des bâtiments peuvent aussi être contrôlé par un RCSF [XRC⁺04] ou bien les pipelines [SNM07].

Domaine commercial et industriel Les applications précédentes servaient principalement à étudier un phénomène physique, naturel ou environnemental par le monde de la recherche et du développement. Par conséquent, elles sont moins contraintes au niveau économique ou au niveau de la robustesse que les applications à visées commerciales. La mise en place de standard pour le déploiement rapide d'applications des RCSF pour le grand public ou pour l'industrie devient pressant. En effet, cette standardisation permettrait de s'abstraire simplement de la pile protocolaire du réseau et de sa gestion pour se concentrer sur la partie applicative. Cette standardisation semble tendre par exemple pour la partie réseau vers l'utilisation du ZigBee ou de ses dérivées [WC07].

De même les applications des réseaux de capteurs devenant de plus en plus nombreuses, une mise en commun de leurs unités peut être envisagés pour permettre le développement d'applications sans toutefois recourir à un nouveau déploiement. Pour ce faire une certaine virtualisation des services est envisageable, voir par exemple [ILH13] dans le cadre des RCSF déployer dans une maison.

Les réseaux de capteurs sont prometteurs dans l'industrie ou dans le tertiaire, que se

soit pour limiter le gaspillage énergétique — ou plus exactement entropique — via une gestion des éclairages [SAA04], ou que ce soit pour faciliter la gestion des stocks de données périssables sensibles à la température comme la nourriture [QXF⁺14]. Notons que certaines réalisations existent dans ce contexte, par exemple l'article [VRM⁺12] décrit une application de surveillance d'une usine de café instantané de la conception du système embarqué à la mise en place d'une interface de contrôle sous LabVIEW.

Toutefois les réseaux de capteurs dans l'industrie ne se limitent pas à la surveillance d'un système ; leurs utilisations potentielles dans les smart grid [ZLZ12] incluent aussi bien la surveillance que le contrôle de certaines parties de la gestion afin de répondre au plus vite à la demande énergétique.

Naturellement, les réseaux de capteurs s'introduiront vraisemblablement chez les particuliers par la domotique [TVP⁺11] pour gérer automatiquement les services de la maison comme l'éclairage, le chauffage, etc.

Domaine médical Certaines applications de surveillance médicale formeront la seconde voie d'intrusion des réseaux de capteurs à domicile. En effet, les réseaux de capteurs viendront compléter à terme une gamme de produits déjà existant [Sta09, SCD⁺05]. Par exemple, la surveillance de malades ou de personnes âgées à domicile réduiront les coûts de surveillance tout en la renforçant car cette dernière se fera continuellement alors que les visites d'une infirmière sont naturellement limitées en nombre. Si le réseau de capteurs détecte un problème, un système d'alerte [MSMSM13] est alors enclenché. Mais cette surveillance à domicile peut aussi s'étendre aux hôpitaux [MSMSM13], ce qui permettra de redonner de la liberté de mouvement aux résidents [VWS⁺06].

Cependant, la surveillance du patient doit être opérée par un réseaux de capteurs qui associe à la fois un réseau local et un réseau personnel d'une dizaine de capteurs placés directement sur le patient [MOJ06] ; ce qui impacte fortement la mise en œuvre.

De plus, les données médicales sont particulièrement sensibles car elles doivent être manipulées ou transmises sans en divulguer la teneur afin de respecter la vie privée. L'article [GRS08] propose une mesure pour mesurer l'intégrité des données et [SP04] présente plus généralement les défis des réseaux de capteurs pour le domaine de la sécurité des données et du réseau.

Quelques prototypes d'application médicale ont déjà été testés comme le système de surveillance CodeBlue [SCL⁺05] avec une conception allant des capteurs à l'architecture du système d'information dédié ou encore MEDiSN [KLC⁺10] qui présente une architecture de réseau de capteurs qui détecte et gère les situations médicales d'urgence.

Domaine militaire La capacité de surveillance des réseaux de capteurs et leur simplicité de déploiement en font des atouts majeurs dans le domaine militaire. En effet, leur faible coût et leur capacité de projection sur le théâtre des opérations rendent très attrayant les réseaux de capteurs dans le domaine militaire. Les types de capteurs embarqués

sont nombreux [DTD12], tels que des capteurs acoustiques pour la détection et la localisation de tireurs ou le repérage de menaces ennemies, ou encore des capteurs chimiques ou biologiques pour les attaques de cette nature ou encore optiques ou vibratoires.

Le nombre d'applications militaires des réseaux de capteurs semblent sans limite et leur flexibilité impose des contraintes fortes qui sont autant de défis aussi bien théoriques que pratiques [WTH08]. Quelques conceptions sont proposées, par exemple, dans [LLS09] pour des réseaux de capteurs déployé à grande échelle et dans [LTD11] pour une hiérarchisation du réseau permettant de multiples accès au système en des lieux différents et mobiles.

L'un des attraits des réseaux de capteurs et leur capacité de détecter et de suivre une cible sur le terrain des opérations [BHK⁺06]. Toutefois cette surveillance et cette poursuite ne se réalise pas sans soulever des difficultés énergétiques [HKS⁺04].

À l'instar des applications dans le domaine médical mais pour des raisons évidentes et différentes, la sécurité des réseaux de capteurs dans le domaine militaire est primordiale. Toutefois, les solutions [SP04, GRS08] d'un domaine peuvent se transposer à l'autre. Ajoutons toutefois que la protection des communications militaires est plus flexible car elle peut aussi mettre à profit des méthodes de brouillage au niveau de la couche physique des émetteurs et des récepteurs sans fil.

Autres domaines Les réseaux de capteurs ne se limitent pas aux exemples présentés ci-dessus. Ils feront partie intégrante de « l'agriculture intelligente » en surveillant les zones agricoles [MYM11] ou en proposant une gestion optimale des produits phytosanitaires notamment en fonction des besoins localisés à des niveaux fins de précision spatiale.

Les zones urbaines proposeront aussi un terrain de jeux pour les applications des réseaux de capteurs. Ils permettront une gestion plus fine de l'éclairage public ou des ordres ménagères. Le trafic routier d'une « ville intelligente » sera rendu plus fluide en aidant les usagers de la route à se garer ou à circuler selon un itinéraire optimisé en temps réel [DMA14].

La recherche aérospatiale n'est pas mise de côté. Par exemple l'article [GGB⁺09] propose un prototype de réseau de capteurs qui surveille la température d'une turbine à gaz pour pouvoir analyser des tests ou encore pour la maintenance.

La plage d'application reste encore large, les réseaux de capteurs pour servir d'aide à l'arbitrage sportif, etc.

2.1.3 Caractéristiques

Un réseau de capteurs sans fil est un réseau informatique indépendant dont les unités de base sont les nœuds de capteurs. Malgré leur grande diversité, ces réseaux ont en commun plusieurs caractéristiques essentielles.

La première caractéristique concerne l'étendue spatiale de leur champ d'observation. En effet, il est souvent simple de projeter un grand nombre de nœuds sur le terrain d'application. Par exemple, l'installation des nœuds sur le théâtre des opérations peut se faire par des tirs d'artillerie. Ou encore, la mise en place d'un RCSF dans un milieu difficile pour la surveillance de l'environnement comme la montagne ou les zones humides peut se faire par un largage des unités. Dans ces exemples, le placement des unités est aléatoire, leur nombre permettant d'avoir une couverture suffisante. Toutefois, il peut être judicieux de placer les nœuds de façon appropriée lorsque cette opération est possible ; le nombre de nœuds sera plus faible et leur complexité pourra augmenter en conséquence à coût équivalent à une disposition aléatoire [MBE10]. La mobilité des nœuds élargit encore les possibilités pour le déploiement des capteurs en le corrigeant *a posteriori* [CS09].

Les nœuds des RCSF se doivent d'être autonome énergétiquement. Cela impose une gestion de l'énergie fine pour augmenter la durée de vie de l'ensemble du réseau. L'économie de l'énergie se fait principalement par deux processus : la mise en veille des capteurs pour limiter la consommation et le traitement local de l'information pour éviter des pertes dues à un surplus des communications. En effet, le coût énergétique de traitement d'un bit est 100 fois moindre que sa transmission [RF10]. Certains capteurs sont associés à une unité de production d'énergie — solaire, chimique, biologique, mécanique — qui permettent de recharger le nœud. Toutefois, ces capteurs ne sont pas exempts de la gestion de l'énergie car ces sources d'énergie sont aussi limitées et imposent des contraintes fortes sur le traitement et les communications [CTT10, IOT12, TCC13].

Les contraintes énergétiques ne doivent pas seulement être considérées lorsque le déploiement du réseau de capteurs est effectué mais aussi en amont, lors de sa conception et une gestion inter-couche notamment avec la couche PHY est primordiale [SCI⁺01].

La coopération entre les nœuds du réseau est nécessaire et primordiale car aucun nœud, hormis peut-être la station de base, n'a connaissance de toute les informations spatialement étiquetées. Comme le traitement doit faire vraisemblablement une fusion de toute ces informations, soit une transmission complète à un nœud calculateur soit un calcul distribué est nécessaire. Dans chaque cas la coopération est nécessaire : soit pour le transport des données soit pour le calcul [AB12]. Par exemple, les données peuvent être agrégées le long d'un arbre couvrant minimal ou encore en utilisant des traitements séparés par cluster puis les différents résultats sont directement remontés à la station de base [CSZ⁺09].

Dû fait du type de leur déploiement aléatoire, les RCSF nécessitent une grande capacité d'auto-organisation [Mil07]. En effet, la position, les liens entre les nœuds — donc la topologie du réseau — sont rarement connus par avance. Le rôle de chaque nœud doit être attribué lors de ce déploiement. De plus, comme les nœuds du réseau peuvent disparaître, apparaître ou se déplacer lorsque des capteurs mobiles sont présents, la topologie est changeante et le RCSF doit pouvoir s'adapter rapidement à ces modifications pour garantir le bon fonctionnement. La redondance spatiale devient un facteur clef pour permettre cette auto-organisation et ainsi augmenter la durée de vie du réseau au niveau organisationnel [DP10].

La grande variabilité potentielle de la topologie réseau ainsi que la présence de données localisées au niveau de capteur a engendré de nouveau point de vue sur la gestion des RCSF. Il est maintenant naturel de considérer les RCSF comme étant gérable comme une base de données distribuées [BGS00, SZG⁺08].

Cette nécessité d'auto-organisation impacte principalement les couches réseaux OSI des RCSF pour créer, lors de la phase d'initialisation, ou pour garder valide l'infrastructure lors de la phase de maintenance. Cependant, la variabilité de la topologie du réseau doit aussi être prise en compte dans le traitement. Nous verrons dans le chapitre 4 que la disparition des nœuds peut influencer sur la veille des capteurs afin de garantir une bonne couverture de sensorielle.

2.1.4 Critères de conception

Afin de concevoir un réseau de capteurs en vue d'une application, il faut prendre en compte différents aspects. Ces aspects concernent d'une part les équipements à embarqués tel que les capteurs, les modules de communication et de traitement, etc. D'autre part, l'environnement dans lequel un réseau de capteurs est déployé influence, les choix de sa conception. Nous décrivons ci-dessous les différents critères génériques à satisfaire pour concevoir un RCSF pertinent.

Tolérance aux fautes Comme les capteurs ont une structure particulièrement simple, que leur capacité d'alimentation est limitée et que leur fabrication est à bas coût, ils sont particulièrement sensible aux moindres défauts de fabrication ou de positionnement. Ils peuvent être facilement endommagés avant le déploiement, lors de celui-ci ou par l'environnement lors du fonctionnement du RCSF. Il faut par conséquent éviter que les dommages causés par ou sur un petit nombre du nœuds aient pour résultat la paralysie complète du réseau.

Ainsi, la prise en compte d'une qualité insuffisante des capteurs doit être intégrée aux algorithmes de traitement du réseaux de capteurs. L'article [KYX14] propose, par exemple, de détecter par un algorithme décentralisé ces mauvaises mesures par un calcul de corrélations croisées.

Passage à l'échelle Le nombre de capteurs est une variable à fixer finement. Il peut varier de quelques centaines ou quelques milliers à quelques millions. Le modèle de calcul distribué doit être particulièrement efficace dans toutes ces situations et doit donc passer à l'échelle. De même la gestion de l'infrastructure ne peut se faire de façon purement centralisée et statique pour éviter la congestion des transmissions. De même la gestion des interférences reste un sujet sensible au niveau physique des communications [GK00].

Coûts Le coût global d'un RCSF est principalement influencé par le prix des nœuds capteurs et de leur logiciel. Il faut donc réduire au maximum le coût unitaire des capteurs. De même, le logiciel de chaque capteur doit être identique et seul un identifiant marqué matériellement permettra de distinguer les différents nœuds entre eux.

Contraintes matérielles Bien que les composants électroniques des nœuds deviennent de plus en plus intégrés, le choix de la technologie influence le coût mais aussi la qualité des mesures. Le compromis entre un nombre important de capteurs de mauvaise qualité et une quantité plus faible mais de meilleure facture devient le premier élément de choix. D'autres contraintes matérielles s'ajoutent comme l'intégration des capteurs dans l'application : invisibilité des capteurs, accroche des capteurs sur des supports physiques ou biologiques, etc.

Transmission Selon l'environnement de mesure, le support de transmission du réseau de capteurs doit être choisi avec soin. Les communications peuvent par exemple reposer sur de l'infra-rouge, des transmissions radio ou encore optique. Le canal de transmission sera différent selon ce choix. De même, si des capteurs sont mobiles, leur retransmission pourra se faire par exemple par des communications en champs proche (NFC) lors de leur passage à proximité d'un récepteur NFC fixe. Naturellement, le développement de protocoles tels que ZigBee ou 6LoWPAN basés sur la norme IEEE 802.15.4 tendent à fixer les media de transmission aux ondes radio et leurs bandes de fréquences.

De même la compression des données est primordiale pour limiter les communications et augmenter la capacité réelle du réseau. Cette compression et sa relation avec la transmission doit être particulièrement étudiée notamment dans des environnements particuliers comme les tunnels [HLH14].

Environnement Les capteurs sont sensés détecter leur environnement. Selon l'application, cela peut être la maison, les terres agricoles, les cours d'eau, collines, bâtiments, et même la pelouse ou encore des zones dangereuses comme les champs de bataille ou dans les profondeurs inconnues de l'espace humain comme les océans, les cratères, etc. Par conséquent, les capteurs doivent être conçus pour résister à leur environnement tout en recueillant une grande variété de mesures environnementales.

Infrastructure Le choix de la gestion de l'infrastructure du réseau est primordial car la topologie du RCSF variera profondément pendant sa vie. On peut décomposer en trois phases la mise en place, l'utilisation et la maintenance du RCSF :

1^{ère} phase : la planification et la phase déploiement Cette phase concerne principalement la façon de disposer les nœuds dans la zone cible ainsi que leur densité spatiale. Dans certains environnements, le déploiement des capteurs peut être soigneusement planifié à l'avance en se basant sur un modèle *ad hoc*. Mais dans la plupart des circonstances, par exemple pour une dispersion dans les océans, les

forêts et autres champs de bataille, la prédiction de disposition des capteurs disséminés aléatoirement sur la zone cible est impossible.

2nde phase : initialisation de l'infrastructure En théorie, après que les capteurs soient complètement déployés, les nœuds ont formé une architecture complète de réseau de capteurs sans fil. Toutefois la position des capteurs est facilement influencée par des facteurs environnementaux, comme la pluie, le vent. De même l'épuisement des batteries, ou des dysfonctionnements internes entraînent des modifications profondes de la topologie du réseau et donc de son infrastructure. Le système doit faire face à de telles situations pour réagir intelligemment. Par conséquent, la conception des logiciels des nœuds de capteurs des RCSF doit prendre en compte ces possibles modifications.

3^{ème} phase : déploiement supplémentaires Après une certaine période, de nombreux capteurs peuvent être indisponibles en raison d'une grave pénurie d'énergie ou d'un endommagement extérieur. Le nombre et la position de ces capteurs indisponibles peuvent rendre le réseau fragile. Il devient alors nécessaire, si possible, d'intervenir extérieurement par un redéploiement, ciblé ou non, de capteurs dans la zone pour palier aux manques et ainsi augmenter la durée de vie du RCSF. L'auto-organisation du réseau rend cette opération plus simple mais le choix de l'instant de cette opération est importante tout comme la quantité de capteurs et le ciblage de leur redéploiement.

La quantité importante de critères présentés ci-dessus ainsi que les vastes possibilités d'application des réseaux de capteurs rend difficile les choix de conception. L'article [RM04] fait l'inventaire et crée un « espace » de conception des réseaux de capteurs.

2.2 Couverture et connectivité

Les nœuds d'un RCSF interagissent avec d'une part les autres nœuds pour former un réseau de communication et d'autre part avec le monde physique via les capteurs. Ces interactions sont fortement liées à la position des capteurs sur le terrain des opérations et donc à la stratégie de déploiement du RCSF.

Naturellement, il est nécessaire de disposer de mesures qualitatives et quantitatives de ces interactions entre les nœuds et avec le monde physique pour pouvoir comparer objectivement les déploiements des RCSF. Mais ces mesures sont aussi très utiles pour configurer le réseau en temps réel afin d'augmenter sa durée de vie tout en respectant les contraintes de son application.

La connectivité du réseau de communication est bien sûr tout indiquée pour fournir la mesure adéquate des interactions entre les capteurs. Pour les relations avec les phénomènes captés par le RCSF, il n'y a pas de mesure universelle mais plusieurs notions de couverture qu'il faut adapté selon l'application. Ces deux aspects sont discutés plus en détail ci-dessous.

2.2.1 Connectivité

Comme les unités d'un RCSF communiquent entre eux sans fil, il est naturel de construire le graphe du réseau des communications. Chaque sommet du graphe est une unité et deux unités sont reliées entre elles si elles peuvent communiquer entre elle directement. Les articles de la littérature concernant la connectivité dans les réseaux considèrent majoritairement que les deux unités peuvent communiquer dans les deux sens, toutefois si la communication est unidirectionnelle, il reste possible, *mutatis mutandis*, de considérer des graphes orientés.

À partir du graphe des communications, nous pouvons calculer la connectivité, c'est-à-dire savoir si tous les nœuds peuvent communiquer avec tous les autres nœuds en utilisant si nécessaire plusieurs nœuds intermédiaires. Comme le graphe est fini, cela est équivalent au fait qu'il n'y a qu'une unique composante connexes ou encore qu'entre deux sommets du graphe, il existe au moins un chemin.

Ainsi la connectivité assure la cohésion du réseau de communication. Toutefois, elle ne mesure pas la capacité du réseau à résister à la suppression de certains liens de communications. Le critère de k -connectivité permet de répondre à cela en calculant le nombre minimal de chemins entre deux nœuds distincts du réseau. Selon le théorème de Menger [Die00], la k -connectivité est reliée à la résistance du réseau.

L'utilisation de la théorie des graphes pour étudier les problèmes de connectivité est naturelle et permet l'utilisation d'une grande richesse théorique pour en faire l'analyse [GR01] mais aussi un grand nombre d'algorithmes pour les calculs effectifs [Jun05].

Connaître la connectivité d'un réseau de capteur à chaque instant permet d'en augmenter la durée de vie en modifiant sa configuration. En effet, la diminution de la puissance d'émission implique une consommation énergétique plus faible et donc une plus grande durée de vie. Il faut donc si possible limiter au maximum cette puissance d'émission pour chaque capteur tout en garantissant une bonne connectivité globale. Un avantage collatéral de la diminution de la puissance d'émission est la limitation induite de l'interférence au niveau des récepteurs. Ainsi, les communications seront moins affectées et la probabilité d'erreur améliorée. La capacité du réseau au sens de [GK00] est augmentée au niveau des constantes des bornes asymptotiques du débit.

Toutefois la puissance n'est peut-être pas modulable finement. Une approche plus radicale consiste à mettre en veille, autant que faire se peut, les modules de communication des nœuds qui ne sont pas primordiaux pour maintenir le réseau. Leur réveil se fera lorsque le besoin s'en fera ressentir. Une autre solution consiste à mettre en place un ordonnancement des périodes de veille et de sommeil pour chaque capteur sous la contrainte de connectivité du réseau de communication [GD08, ZZS12].

La connectivité initiale, c'est-à-dire au moment du déploiement du réseau est essentielle. En effet, si cette dernière n'est pas suffisante il faudra faire un redéploiement si possible ou bien déplacer des nœuds si la possibilité en est offerte soit par une unité mobile soit par intervention de l'opérateur du RCSF.

Heureusement, la connectivité d'un déploiement aléatoire est bien étudiée par la théorie de la percolation [FM07]. Sous un modèle de déploiement aléatoire suivant un processus spatial de Poisson et pour plusieurs modèles de communication, il existe une densité critique θ telle que si la densité du réseau est supérieure à θ alors le réseau est connecté avec une forte probabilité. Ces résultats sont valables pour des réseaux infinis mais ils restent pertinents dans la situation d'un RCSF suffisamment dense.

Ainsi le déploiement aléatoire est une bonne solution si la densité des nœuds est suffisante. Il est de plus possible de moduler l'homogénéité du réseau selon les zones à couvrir tout en ayant la garantie de connectivité.

D'autres stratégies de déploiement plus déterministes sont envisageables pour répartir de façon la plus homogène les capteurs et qui garantissent la connectivité. Toutefois, ces stratégies sont aussi contraintes par les problèmes de couverture que nous verrons ci-dessous.

Comme nous l'avons vu, la partie réseau des RCSF se mesure notamment par la connectivité et la k -connectivité du graphe des communications. Il en va donc de même pour le routage. Toutefois, les spécificités des RCSF doivent être prises en compte dans la mise en place du routage. D'une part, le routage doit être suffisamment solide pour résister à la défaillance des nœuds. Une solution basée sur la topologie algébrique est proposée dans [ZSL⁺10]. Plus précisément, le graphe de communication est immergé dans un espace hyperbolique puis le revêtement universel est calculé. Une fois fait, le routage devient un simple routage glouton et géré uniquement localement.

Cette idée d'immerger dans un espace euclidien puis de modifier la représentation du graphe des communications permet de plus d'équilibrer les routes entre les nœuds. Par exemple la solution précédemment citée [ZSL⁺10] ou celle de [GNB⁺14] évitent la création « d'autoroutes » dans le routage car celles-ci tendent à utiliser quelques nœuds clés qui, par conséquent, utilisent d'avantage leur réserve d'énergie.

2.2.2 Couverture

La connectivité est importante pour le réseau du point de vue des communications entre les nœuds mais la couverture sensorielle est la véritable raison d'être des réseaux de capteurs. En effet, le déploiement d'un tel réseau est justifié par son application qui découle des mesures physiques faites sur le terrain. C'est pourquoi le problème de la connectivité du réseau est souvent transformé en pratique en une contrainte à satisfaire pour l'optimisation de la couverture.

La couverture est le cœur des applications des réseaux de capteurs car elle mesure la relation entre ce réseau et l'environnement. Naturellement, la quantification de cette couverture dépend fortement des besoins de l'application, c'est pourquoi plusieurs notions de couverture sont apparues.

Nous appellerons dorénavant couverture la couverture sensorielle.

2.2.2.1 La modélisation de la couverture

Quelque soit le type de couverture du réseau de capteurs souhaitée, elle se déduit de la couverture associée à chacune de ses unités. Il est donc nécessaire d’avoir un modèle de la couverture d’une seule unité. Ce modèle est souvent très simplifié par rapport à la réalité car la mesure n’est viable qu’à la position du capteurs et l’inférence de la mesure sur une surface entourant le capteurs n’est pas immédiate ni même parfois possible.

Toutefois, selon le modèle le plus simple possible, une unité couvre un disque d’un rayon fixé, dit rayon de couverture [ZZS12]. La couverture globale du réseau est alors la juxtaposition des disques autour de chaque unité.

Naturellement ce modèle simpliste présente de nombreux défauts. Il ne permet pas de prendre en compte les spécificités locales autour des capteurs comme les accidents du terrain entraînant un domaine de couverture non circulaire. Cependant, ce modèle a l’avantage de sa simplicité et permet de se forger une intuition sur les notions de couverture.

Un modèle plus réaliste peut être obtenu en introduisant en plus un anneau autour du disque de couverture dans lequel, avec une certaine probabilité, le capteur détecte ou non un événement. Ce modèle peut encore être affiné en affectant une probabilité de détection qui décroît avec la distance de l’événement au capteur.

La forme de disque du domaine de détection pour chaque capteur est contraignante. Toutefois, par la topologie et grâce à la définition d’une intégrale utilisant la mesure d’Euler [BG10], il est possible de s’en détacher partiellement pour compter le nombre de cibles [BG09] même si le domaine de détection de chaque capteur n’est pas un disque et même si les capteurs ne peuvent pas identifier les cibles.

Les modèles précédents sont particulièrement attractifs pour les problèmes de détection car alors, il est particulièrement simple de construire la couverture de détection ne serait-ce que sous forme probabiliste en juxtaposant les disques de couverture de chaque unité. Mais cette approche reste très limitée si l’application du réseau de capteurs s’appuie sur des problèmes d’estimation du champ de mesure.

D’autres modèles sont alors nécessaires pour reconstruire le champ à partir des mesures faites par les capteurs. Il s’agit bien maintenant de problème d’échantillonnage non uniforme d’un champ bidimensionnel ou tridimensionnel. Une approche similaire un théorème de Shannon–Nyquist est possible en limitant la plage de fréquences du champ comme dans l’article [KIR11] qui prend en compte la piètre qualité des capteurs ou de la quantification des mesure.

D’autres approches plus pragmatiques sont basées sur une reconstruction du champ grâce à des fonctions noyaux si le champ est homogène et isotrope [RM09]. La résolution de la construction de l’estimateur se réduit à une approche linéaire comme dans l’article [NCV08a] qui permet aussi de prendre en compte une imprécision dans la localisation des capteurs. Si le réseau de capteurs comprend des unités mobiles, elles peuvent être mises à contribution pour améliorer la reconstruction du champ [UV13].

Naturellement, il faut, parallèlement à l'utilisation de ces modèles basés sur l'échantillonnage spatial, redéfinir la notion de couverture pour pouvoir répondre aux questions concernant l'insuffisance ou l'absence de la couverture. Nous verrons à la section 3.3 comment nous avons répondu à cette difficulté grâce à la reconstruction par régression gaussienne d'un champ bidimensionnel.

Dans tous les cas, les modèles précédents supposent que la localisation de chaque capteur est disponible pour pouvoir reconstruire soit le champ soit la zone couverte pour enfin répondre aux problèmes de la couverture. Il semble en effet intuitif que pour connaître la couverture d'un réseau de capteurs il faille connaître leurs positions, au moins approximative. Or, même sans cette information ou alors dans une version très dégradée, certaines connaissances sur la couverture peuvent être extraites ! En effet, la présence d'un ou plusieurs trous dans la couverture grâce à des outils topologiques reste détectable à la suite des travaux de Ghrist, de de Silva et de Muhammad [GM05, dSG06, dSG07a, dSG07b]. Leurs travaux forment les fondations de nos résultats et de notre approche présentées au prochain chapitre, à la section 3.3.

2.2.2.2 Différentes notions de couverture

Quelque soit le modèle de la couverture utilisée, les applications ne recherchent pas forcément le même objectif. Il est d'usage de réutiliser dans le domaine des réseaux de capteurs la classification introduite par Gage dans [Gag92] pour les systèmes multi-robots.

Il faut par conséquent distinguer quatre notions de couverture selon l'objectif [IM05] qui sont la couverture ponctuelle, de zone, de frontière ou glissante.

La couverture ponctuelle ou « point coverage » en anglais, consiste, comme cela est illustré sur la figure 2.4 à assurer la surveillance de certains points particuliers de la zone où le réseau de capteurs est déployé.

Ce type de couverture semble antinomique avec le déploiement d'un réseau de capteurs qui, échantillonnant un espace bidimensionnel ou tridimensionnel, cherche à surveiller une zone. Toutefois, la surveillance de multiples points d'une zone permet d'approcher la surveillance d'une zone si les points à surveiller sont suffisamment dense et bien placés [YDC05].

Il est en effet plus simple de déterminer si un réseau de capteurs surveille quelques points [HB06] que de certifier s'il surveille parfaitement une zone. De plus, une approche aléatoire de déploiement, par exemple en utilisant un processus spatial de Poisson, est grandement simplifiée par cette approche.

La couverture de zone ou « area coverage, » est la couverture généralement sous entendue dans le domaine des réseaux de capteurs. Il s'agit de certifier si une zone prédéterminée est couverte par l'ensemble du réseau comme l'illustre la figure 2.5. En effet,

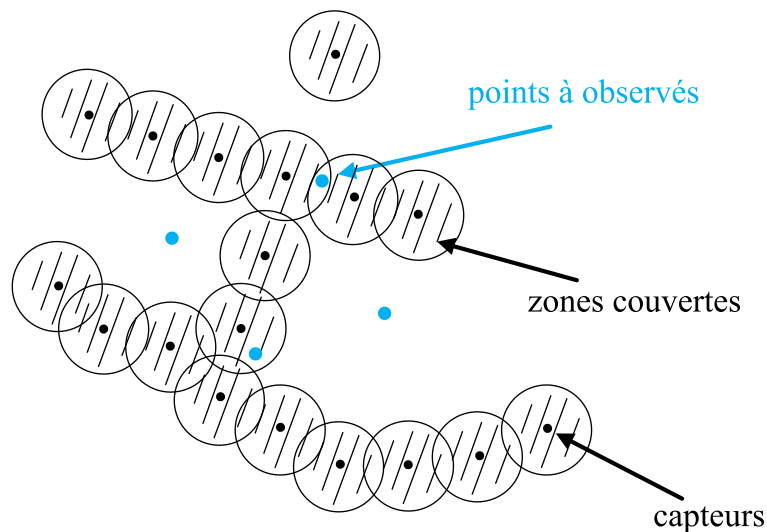


FIGURE 2.4 – Couverture ponctuelle

toutes les applications environnementales, militaires ou autre reposent sur la surveillance d'une zone.

À l'instar de la connectivité, il est aussi possible d'introduire la notion de k -couverture [HT03] qui indique que chaque point de la zone à observer l'est par au moins k capteurs.

Naturellement, certifier la couverture d'une zone à partir des capteurs dépend du modèle de couverture de chaque unité comme ceux présentés à la section précédente. Les algorithmes de certification reposent souvent sur la position bruitée ou non des capteurs et utilisent la géométrie à partir des positions et d'un modèle basé sur un rayon de couverture.

Toutefois, quelques approches [ZH05, JS08] utilisent un moyen détourné pour revenir à la méthode géométrique précédente en assumant une relation entre le rayon de communication et le rayon de couverture et en travaillant alors sur le réseau de communication pour inférer indirectement la couverture.

Nous verrons dans le prochain chapitre 3 que, grâce à la topologie algébrique la position des capteurs n'est pas si importante et que cette donnée peut être relâchée. Ainsi un modèle géométrique basique de la couverture peut être remplacé par un modèle plus réaliste basé sur les corrélations entre les mesures des capteurs.

Décider si une zone est couverte ou non est nécessaire pour définir la stratégie de déploiement. En effet, si le déploiement n'est pas aléatoire ou si la densité locale des capteurs peut être choisie, alors il faut positionner les capteurs de façon optimale tout en assurant la couverture [BKX⁺06] ou la k -couverture [BXY⁺08, BYX⁺08] mais aussi la connectivité du réseau.

La sélection des capteurs forme la seconde utilisation des algorithmes déterminant la couverture. En effet, si le réseau est densément déployé, il y a certainement beaucoup de

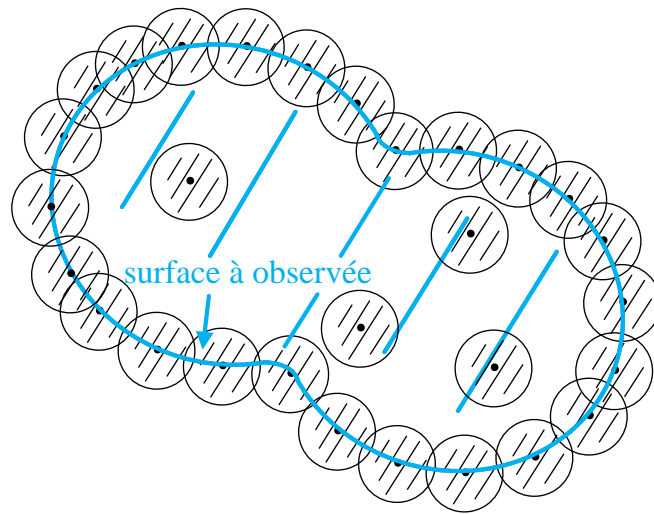


FIGURE 2.5 – Couverture de zone

redondance entre les capteurs. Il est donc possible de n'en sélectionner qu'une partie tout en garantissant la couverture et la connectivité [ZH05, SRL07, JS08].

Il est possible par la sélection des capteurs d'augmenter drastiquement la durée de vie du réseau en mettant en veille les capteurs redondants. Le réveil des capteurs ne se fait que si nécessaire par exemple lorsque des capteurs disparaissent en créant un trou de couverture.

Les algorithmes de sélection de capteurs sont des réminiscences d'heuristique résolvant le problème dit du « sac à dos » et sont donc difficiles au sens théorique du terme. Toutefois, sans chercher à atteindre l'optimal, il est possible de sélectionner quelques capteurs avec de la redondance résiduelle mais plus simplement. Nous verrons qu'un résultat connexe à notre algorithme du chapitre 4 permet de faire de la sélection de capteur adaptative.

La couverture de frontière ou « barrier coverage, » est principalement utilisée pour la détection d'intrusion. Il est nécessaire que le réseau de capteurs couvre la frontière ou la barrière à surveiller comme l'illustre la figure 2.6.

La couverture de frontière est la version unidimensionnelle de la couverture contrairement à la version bidimensionnelle de la couverture de zone et de la version ponctuelle. Mais à l'instar du théorème de Green, connaître les passages d'une cible au travers d'une frontière permet de connaître le contenu d'une zone. Cela permet de faire de la poursuite de cible simplement [SG13].

La problématique est double. Il faut soit certifier la couverture d'une barrière [KLA05, CKL07] soit créer une barrière d'un bord à l'autre d'une zone de déploiement d'un réseau de capteur [LDW08]. Dans ce dernier cas, un réseau est déjà déployé et il faut s'assurer par exemple qu'il n'existe pas de route allant de l'est à l'ouest qui soit indétectable par le réseau.

Si le déploiement est aléatoire, la probabilité de présence d'une barrière dans un réseau

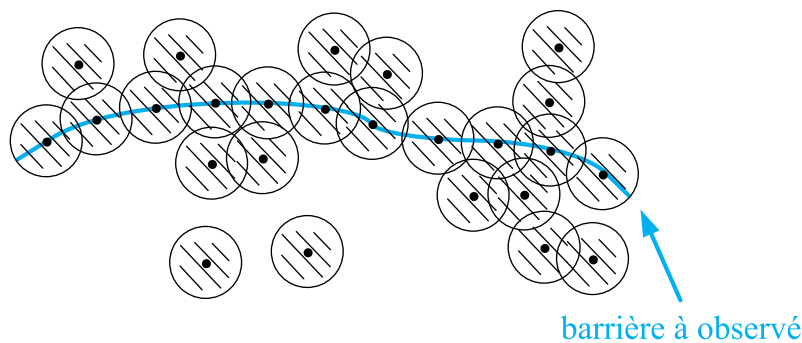


FIGURE 2.6 – Couverture de frontière

de capteur est fortement reliée à la théorie de la percolation. Cette théorie indique qu'il y a une densité minimale de capteurs au delà de laquelle, avec une grande probabilité, la présence d'une barrière est assurée [FM07].

La couverture glissante ou « sweep coverage, » fait intervenir des unités mobiles dans le réseau de capteurs. La mobilité permet à un capteur de couvrir une zone plus grande comme le montre la figure 2.7.

L'introduction du temps étend le problème de couverture. Il faut par exemple déterminer un chemin qui assure la couverture dans un temps donné d'une série de points d'intérêt [LCL⁺11] ou d'une zone. La couverture de frontière est aussi modifiée par la dimension temporelle : il s'agit alors de certifier qu'aucune intrusion ne peut être détectée lors de la trajectoire des différents mobiles. Ainsi, la couverture glissante n'est pas spécifiquement une notion de couverture mais plus exactement la prise en compte de la mobilité de certains capteurs.

Il faut donc pouvoir déterminer algorithmiquement si la couverture est garantie étant donné le réseau et les mouvements. Pour cela, la topologie algébrique peut aider [Ada13]. L'application de ce type d'algorithmes est alors de planifier les trajectoires des capteurs mobiles.

2.3 Conclusion

Nous avons passé en revue dans ce chapitre différents aspects des réseaux de capteurs allant de leur définition à leurs applications en passant par leurs caractéristiques. Parmi celles-ci, les notions de connexité et de couverture ont été d'avantage approfondies.

La suite de la présentation de nos travaux en sera simplifiée car nous avons mis en exergue les particularités de la couverture qui est à la base de nos choix. Par exemple, la difficulté de la localisation précise des capteurs nous a amené dans le prochain chapitre à choisir

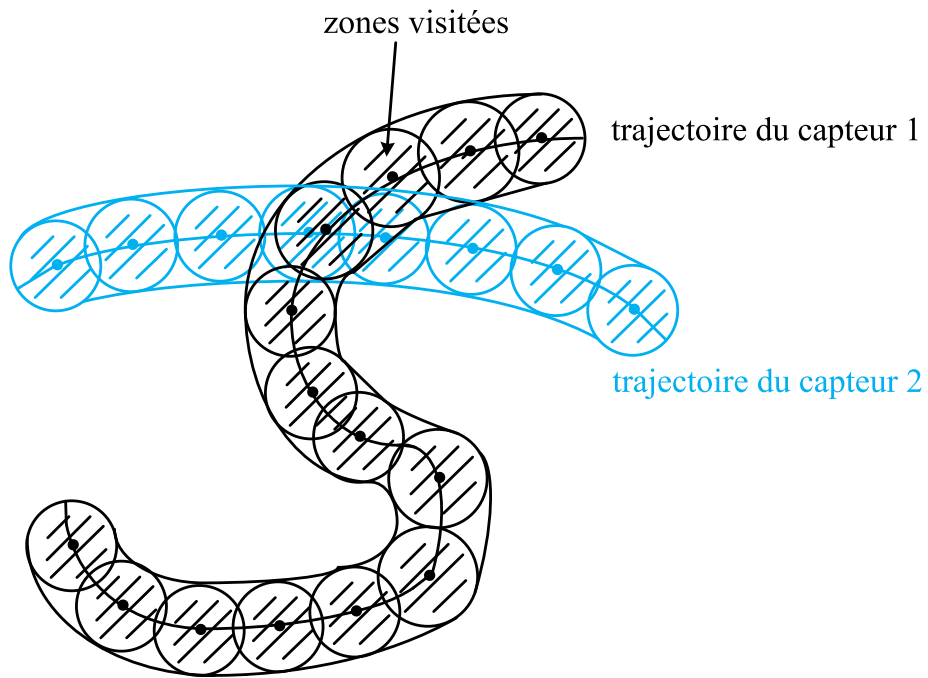


FIGURE 2.7 – Couverture glissante

une détection des trous de couverture basée sur une approche purement topologique qui ne présuppose aucune connaissance des positions des capteurs.

De même, nous avons concilié une modélisation stochastique avec la détection de trou de couverture par la topologie pour répondre à la rusticité des modèles de couverture sensorielle qui sont classiquement utilisés dans la littérature et que nous avons survolés dans ce chapitre

Chapitre 3

Homologie et couverture dans les RCSF

3.1	Notion de topologie et de topologie algébrique	26
3.1.1	Représentation des espaces	27
3.1.2	Linéarisation	28
3.1.3	Groupes d'homologie	30
3.1.4	Persistence homologique	40
3.1.5	L'homologie au secours des RCSF	43
3.2	Éléments sur les processus gaussiens	47
3.2.1	Définition	48
3.2.2	Régression	50
3.2.3	Calcul des hyper-paramètres	52
3.2.4	Exemple de régression d'un champ déterministe	53
3.3	Application à la couverture	55
3.3.1	Description de l'application	55
3.3.2	Construction du complexe simplicial	57
3.3.3	Détection des trous de couverture	60
3.3.4	Utilisation de la persistence	60
3.3.5	Mise en veille de capteurs	62
3.3.6	Remarques sur l'algorithme	63
3.4	Conclusion	65

Les réseaux de capteurs sans fil gèrent localement des données localisées qui doivent être ensuite intégrées et fusionnées pour fournir la mesure globale nécessaire à l'application. Ce passage du local au global est fondamentalement inhérent à la topologie du réseau. En effet la localisation est prise en compte par la notion de proximité alors que le passage au global est fait par le calcul d'invariants topologiques qui concernent l'espace entier. Prenons l'exemple de la formule de Gauss-Bonnet qui relie la courbure d'un espace, notion

purement locale, à la caractéristique d'Euler-Poincaré de l'espace, notion topologique purement globale qui compte algébriquement le nombre de « trous » de l'espace [Ghr14].

La topologie permet aussi des raisonnements inverses qui montrent l'impossibilité de ce passage vers le global. Par exemple, Penrose démontre dans [Pen93] par la topologie algébrique que la fameuse tri-poutre d'Escher est impossible à construire dans l'espace. Plus précisément, un calcul de groupe de cohomologie d'un espace bien choisi montre une obstruction à sa construction.

Certains livres récemment publiés [Ghr14] ou [Rob14a] présentent bien l'intérêt de l'approche topologique du passage du local au global dans un certain nombre d'applications dont les RCSF.

C'est également ce que nous souhaitons montrer en introduisant la topologie et la topologie algébrique dans la section 3.1. Pour pouvoir l'appliquer, nous aurons besoin de revenir sur la régression d'un champ de mesure grâce aux processus gaussiens présentés à la section 3.2. Ensuite, l'application à la détection de trous de couverture de mesure sera l'objet de la dernière section 3.3. Enfin, la section 3.4 conclura ce chapitre.

3.1 Notion de topologie et de topologie algébrique

La topologie est la branche des mathématiques qui étudie les espaces avec une notion de proximité entre les points [AF07, Cro05, Mas91]. Cette notion de proximité doit être prise en un sens très large ; elle ne repose pas nécessairement sur la donnée d'une métrique car alors celle-ci produit des résultats quantitatifs.

La notion de proximité est encodée grâce à un ensemble de sous-ensembles, dont l'espace entier, stable par réunion, par intersections finies ou dénombrables et par complémentaire. Ces sous-ensembles sont appelés les ouverts de l'espace et caractérisent la topologie. Les fonctions continues sont les fonctions entre espace qui respectent la topologie, c'est-à-dire la notion de proximité de l'ensemble de départ et transporté vers l'ensemble d'arrivée. Si ces notions sont équivalentes pour les espaces de départ et d'arrivée d'une fonction, les espaces sont qualifiés d'homéomorphiques et la fonction un homéomorphisme.

La topologie fournit des invariants très souples sur les espaces comme le nombre de composants connexes, la compacité, le genre des surfaces, etc. Ces souplesses proviennent de la généralité des fonctions continues.

La théorie mathématique de la topologie commence depuis un certain nombre d'années à être utilisée en sciences appliquées [EH10, Ghr14].

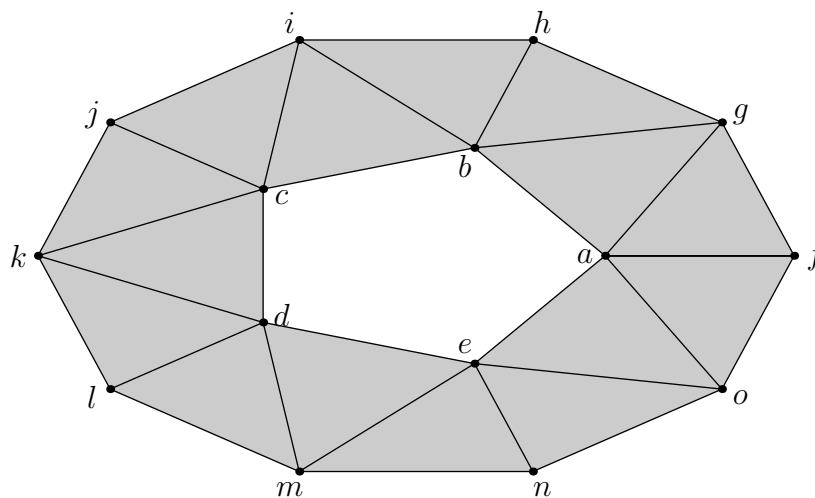


FIGURE 3.1 – Exemple d'espace combinatoire

3.1.1 Représentation des espaces

Toutefois, l'approche algorithmique de la topologie reste difficile sur des espaces trop généraux. Il devient nécessaire de rendre les espaces manipulables et donc définis combinatoirement.

L'approche classique consiste à approcher un espace continu donné sous forme d'une variété, différentielle ou non, par un complexe simplicial. Par exemple une surface est approché par une triangulation. La manipulation des complexes simpliciaux ne nécessite pas la plupart du temps sa réalisation géométrique.

Les espaces sont dans nos travaux des espaces combinatoires comme les complexes simpliciaux ou cellulaires, trips pour Kozlov [Koz08]. Ils sont directement définis par un ensemble de points, puis un ensemble d'arêtes, puis un ensemble de triangles suivi par un ensemble de tétraèdres, etc.

Ces espaces définis combinatoirement permettent l'utilisation d'algorithmes. Algorithmes rassemblés au sein d'une nouvelle branche de la topologie dite topologie combinatoire algébrique [Koz08].

Un espace X de dimension n est décomposé en cellules de dimension $0, 1$ à n . Le recollement entre ces cellules se fait par l'intermédiaire de la notion de bord : le bord d'une cellule de dimension k est un ensemble de cellule de dimension $k - 1$. Une cellule τ qui est sur le bord de la cellule σ est appelée une face de σ alors que σ est une coface de τ .

Si les cellules sont des simplexes, points, arêtes, triangles, tétraèdres, etc. Ces espaces sont appelés des complexes simpliciaux.

La figure 3.1 montre une triangulation d'un anneau par un complexe simplicial qui servira d'exemple tout au long de cette introduction à la topologie algébrique. Cette triangulation est composée de

- 15 cellules de dimension 2, ses triangles ;
- 30 cellules de dimension 1, ses arêtes ;
- 15 cellules de dimension 0, ses sommets.

La bordure d'un triangle est composée de ses arêtes. Par exemple la bordure de abg est la somme de ab , ag et bg . Cependant, les sommets ne font pas partie du bord de abg .

À l'instar de la résolution par Euler du problème des « sept ponts de Königsberg » grâce à une modélisation par un graphe, l'analyse d'une situation utilisant les méthodes topologiques commence souvent par sa formalisation spatiale sous forme de cellules et de relations de bord. Par exemple, les interférences entre les nœuds d'un réseau sont directement modélisées par des complexes simpliciaux dans [Rob13a]. De même [VDM13b, VDM13a] utilisent les complexes simpliciaux pour modéliser la planification des fréquences, la gestion de l'énergie ou encore la couverture.

Les structures de données qui permettent la manipulation des espaces combinatoires reposent surtout sur la relation de bordure. En effet, il suffit de retenir les cellules et la relation d'incidence entre faces et entre cofaces. Si le complexe est simplicial alors des structures plus performantes et moins gourmandes en mémoire sont proposées dans [BM12] et [BDM13]. Par contre si la structure provient d'une triangulation d'une variété dont les triangles sont orientés par la géométrie alors [Zom05] décrit une structure de données, nommée « quadedge data structure, » qui permet une gestion plus précise de cette triangulation.

3.1.2 Linéarisation

Les invariants proposés par la topologie générale sont souvent qualitatifs et donc difficiles à calculer pour un ordinateur. La topologie algébrique [Hat01, Koz08, Mas91, BT82] répond à cela en calculant des invariants quantitatifs. Par contre, ces invariants ne sont pas nécessairement des valeurs scalaires mais plutôt des groupes, des espaces vectoriels ou d'autres structures algébriques.

Nous nous intéresserons surtout aux groupes d'homologie ou de cohomologie. Leur définition nécessite de linéariser l'espace topologique X . À partir de cet espace, nous construisons un complexe qui est un ensemble d'espaces vectoriels C_k et d'applications linéaires $\partial_k : C_k \rightarrow C_{k-1}$,

$$0 \xrightarrow{\partial_{n+1}} C_n \xrightarrow{\partial_n} C_{n-1} \xrightarrow{\partial_{n-1}} \dots \xrightarrow{\partial_{k+1}} C_k \xrightarrow{\partial_k} C_{k-1} \xrightarrow{\partial_{k-1}} \dots C_1 \xrightarrow{\partial_1} C_0 \xrightarrow{\partial_0} 0 \quad (3.1)$$

Les espaces C_k sont formés par les combinaisons linéaires des cellules de dimension k . Un vecteur de C_k est donc de la forme $\mathbf{c} = \sum_{\sigma \in X_k} c_\sigma \sigma$; autrement dit \mathbf{c} représente un ensemble fini de cellules σ de dimension k pondérées par leur poids c_σ . Ces vecteurs sont appelés des chaînes. Nous noterons C_\star le complexe entier lorsque toutes les dimensions sont prises en compte.

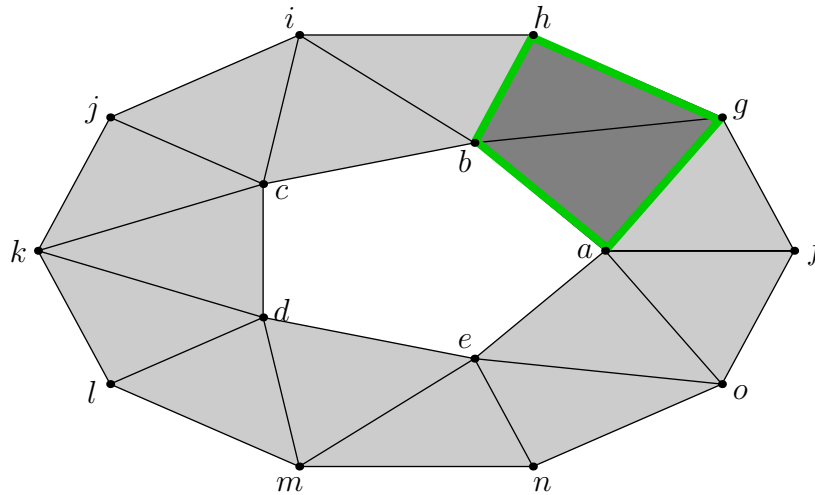


FIGURE 3.2 – Exemple d'une chaîne et de sa bordure

Comme le complexe est de dimension n , il n'y a pas de cellule de dimension $n + 1$. Par conséquent l'espace vectoriel C_{n+1} est trivial ce qui explique l'espace 0 à gauche de (3.1). De même la dimension -1 est ici incongrue d'où un espace trivial pour C_{-1} et la présence du second 0 de l'équation (3.1).

Le poids c_σ des chaînes est souvent pris entier ou appartenant à un corps. L'utilisation de corps finis est intéressante en algorithmique car les erreurs d'arrondis y sont absentes. Plus encore, l'utilisation du corps binaire \mathbb{Z}_2 permet d'omettre l'orientation des cellules car un élément est l'égal de son opposé. Une chaîne est donc représentable comme une collection de cellules. Par la suite, si la perte d'information concernant l'orientation n'est pas primordiale dans nos applications, le corps binaire sera utilisé.

Les opérateurs ∂_k permettent de passer d'une chaîne de dimension k vers une chaîne de dimension $k - 1$ qui représente le bord de la chaîne de départ. Pour simplifier les notations, les indices k des opérateurs ∂_k sont souvent omis car sous-entendus par le contexte.

Les complexes de chaînes et les opérateurs de bord satisfont la propriété fondamentale

$$\partial_k \partial_{k-1} = 0, \quad (3.2)$$

qui est la relation algébrique indiquant que le bord d'un bord est vide. Nous verrons que cette propriété est à la base de l'homologie.

En continuant l'exemple précédent, à une cellule abg , nous associons la chaîne élémentaire (abg) . Sur la figure 3.2, la partie grisée $abhg$ est représenté par la chaîne $(abg) + (bgh)$. L'opérateur de bordure est particulièrement simple, $\partial(abg) = (ab) + (bg) + (ag)$ et $\partial(bgh) = (bg) + (gh) + (bh)$. Par linéarité, nous avons

$$\partial((abg) + (bgh)) = \underbrace{(ab) + (bg) + (ag)}_{\partial(abg)} + \underbrace{(bg) + (gh) + (bh)}_{\partial(bgh)} = (ab) + (ag) + (gh) + (bh), \quad (3.3)$$

car nous travaillons sur \mathbb{Z}_2 et donc $(bg) + (bg) = 0$. Le bord de $abhg$ est donc la chaîne représentée en trait fort vert sur la figure 3.2.

La linéarisation de l'espace d'intérêt en une série d'espaces vectoriels et la linéarisation de la notion de bordure en un opérateur linéaire permet d'utiliser l'algèbre linéaire pour calculer plusieurs invariants. Par exemple, la caractéristique d'Euler-Poincaré $\chi(X)$ d'un espace général X peut s'avérer compliquée à calculer directement. Toutefois, la triangulation de X puis sa linéarisation sous la forme d'un complexe C_\star permet d'y accéder de façon plus calculatoire grâce à la relation

$$\chi(X) = \chi(C_\star) = \sum_{\sigma \in X} (-1)^{\dim \sigma} = \sum_{k=0}^n (-1)^k \dim C_k. \quad (3.4)$$

La caractéristique d'une surface plane est reliée à son genre¹ g par $\chi = 2g - 2$ si la surface est orienté et $\chi = g - 2$ sinon. Notre exemple possède une caractéristique d'Euler-Poincaré de $\chi = 15 - 30 + 15 = 0$ qui indique ici la présence d'un unique trou.

La caractéristique d'Euler-Poincaré peut être vue comme une mesure sur les espaces car $\chi(A \cup B) = \chi(A) + \chi(B) - \chi(A \cap B)$. Tout une théorie de l'intégration [BG10, Cur13] et de transformations intégrales [GR11] peut se développer avec cette approche avec en application [BG09] le comptage de cible grâce à un RCSF dont les capteurs ne peuvent que compter le nombre de cibles alentour sans les identifier. Cette théorie intégrale permet une approche topologique du traitement du signal [CGR12].

3.1.3 Groupes d'homologie

Poincaré [Poi95] a développé la topologie algébrique à la fin du XIX^{ème} et au début du XX^{ème} siècle pour quantifier les espaces topologiques avec des objets plus génériques que des nombres comme les groupes d'homotopie $\pi_k(X)$ dont le groupe fondamental $\pi(X)$.

Toutefois le calcul de ces groupes est difficile et algorithmiquement complexe. La version linéarisée qui repose sur les complexes présentés précédemment est beaucoup plus accessible car ne dépendant que de l'algèbre linéaire mais elle est moins puissante car elle discrimine moins les espaces topologiques.

Les complexes permettent de représenter un sous espace par un vecteur. Cependant, tout les sous espaces ne sont pas intéressants en topologie algébrique ; les groupes d'homologie gardent les aspects essentiels des sous-espaces intéressants tout en retirant les sous-espaces sans intérêt.

1. le genre est le nombre maximum de courbes fermées simples sans points communs pouvant être tracées sur cette surface sans la déconnecter.

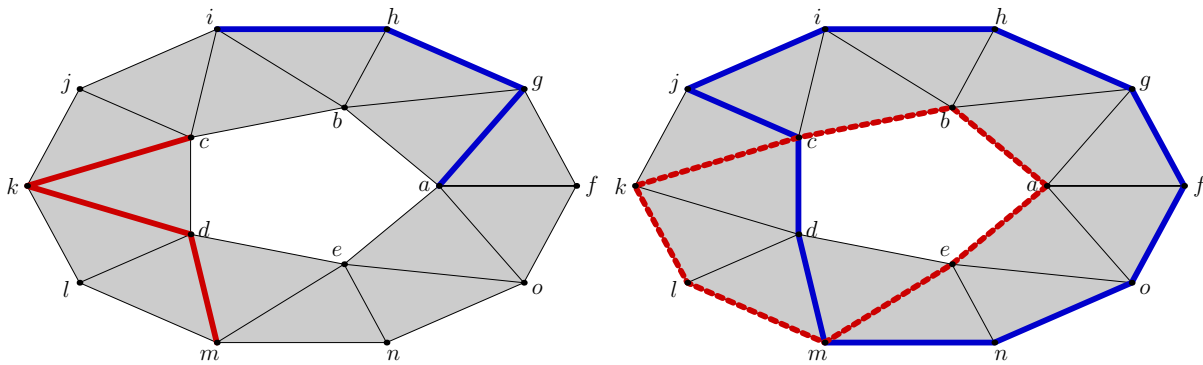


FIGURE 3.3 – Exemple de chaînes

3.1.3.1 Le groupe H_0

Le groupe d'homologie $H_0(X)$ tente de capturer la notion de connectivité du complexe X . Par exemple, sur la gauche de la figure 3.3 nous voyons que la chaîne en bleu $\mathbf{c} = (ag) + (gh) + (hi)$ connecte le sommet (a) et le sommet (i) . Par conséquent (a) et (i) appartiennent à la même composante connexe.

Il reste à transporter ce que nous voyons vers l'algèbre linéaire. Le bord de la chaîne \mathbf{c} vaut $\partial\mathbf{c} = \partial(ag) + \partial(gh) + \partial(hi) = (g) - (a) + (h) - (g) + (i) - (h) = (i) - (a)$ car nous sommes restés général et nous avons pris en compte l'orientation des cellules. Cette dernière équation s'écrit aussi $(i) = (a) + \mathbf{c}$ qui exprime le fait que le sommet (i) s'atteint à partir du sommet (a) en suivant la chaîne d'arêtes \mathbf{c} . Nous retrouvons donc la notion de connexion entre (a) et (i) ; autrement dit, (a) et (i) sont équivalents. De même (c) et (m) sont équivalents car la seconde chaîne, en rouge, de la figure 3.3 les connecte.

Une chaîne d'arêtes quelconque \mathbf{c} connecte son bord $\partial\mathbf{c}$. Par conséquent, le sous espace de C_0 donnée par l'image de ∂C_1 fournit l'ensemble des sommets connectés entre eux. Le bord d'un sommet est quant à lui toujours nul car $\partial_0 = 0$. Donc tous les sommets sont dans $\ker \partial_0$. Le groupe d'homologie $H_0(X)$ s'écrit

$$H_0(X) = \ker \partial_0 / \text{Im } \partial_1, \quad (3.5)$$

qui est un espace vectoriel dont les éléments sont les sommets équivalents comme (a) et (i) le sont.

Comme tout les sommets de la figure 3.3 sont atteignables à partir de (a) en suivant une chaîne, le groupe H_0 est un espace vectoriel de dimension 1 signifiant qu'il n'y a qu'une seule composante connexe.

3.1.3.2 Le groupe H_1

Le groupe $H_1(X)$ recherche les « trous » de X . L'idée est de trouver une chaîne qui fasse le tour du trou sans pouvoir être réduite à zéro.

Notons, pour les besoins de l'exemple, \mathbf{c} et \mathbf{d} les deux chaînes de dimensions 1 à droite de la figure 3.3 qui sont respectivement en trait fort bleu et en trait tireté rouge.

Tout d'abord une chaîne qui entoure un sous-espace ou un trou est nécessairement sans bord. En effet $\partial\mathbf{c} = 0$ et $\partial\mathbf{d} = 0$. Autrement les chaînes n'entoureraient rien. Par contre, certaines chaînes sont inintéressantes car elles entourent un sous-espace plein. Par exemple la chaîne $(ab) + (bh) + (hg) + (ga)$ de la figure 3.2 entoure la chaîne $(abg) + (bgh)$ qui n'est pas un trou car elle appartient à C_2 .

Par conséquent une chaîne d'arête entoure un trou si elle n'est pas la bordure d'une chaîne de cellules de dimension 2. Autrement dit, les chaînes dans $\text{Im } \partial_2$ ne sont pas à prendre en compte.

De même les chaînes \mathbf{c} et \mathbf{d} entourent le même trou. Ils faut donc une méthode pour les rendre équivalentes. En fait, en prenant $\mathbf{b} \in C_2$ la chaîne de dimension 2 qui contient tous les triangles sauf (ick) avec les bons signes, on trouve que $\mathbf{d} = \mathbf{c} + \partial_2\mathbf{b}$; c'est-à-dire que les chaînes \mathbf{c} et \mathbf{d} prises ensemble entourent un sous-espace inclus dans X . Il est donc possible de déformer la chaîne \mathbf{c} en la chaîne \mathbf{d} sans la « déchirer. »

L'espace vectoriel de toutes les chaînes équivalentes décrites ci-dessus s'écrit $H_1(X)$ qui est donné à l'instar de $H_0(X)$ par

$$H_1(X) = \ker \partial_1 / \text{Im } \partial_2. \quad (3.6)$$

La dimension de $H_1(X)$ fournit le nombre de trous détectés par l'homologie. Ses vecteurs sont des classes d'équivalence dont les représentants sont des chaînes « entourant les trous. » Malheureusement, il n'y a pas de représentants canoniques; c'est-à-dire que \mathbf{c} et \mathbf{d} sont tout aussi valables pour représenter le trou de l'anneau. D'autres considérations doivent être utilisées pour choisir entre les deux comme cela sera discuté à la section 3.3.6.2.

3.1.3.3 Les groupes $H_k(X)$

Les constructions précédentes H_0 et H_1 peuvent naturellement et sans difficulté être étendues à toutes les dimensions pour définir les groupes d'homologie

$$H_k(X) = \ker \partial_k / \text{Im } \partial_{k-1}. \quad (3.7)$$

La dimension β_k de $H_k(X)$ s'appelle le k -ème nombre de Betti qui mesure le nombre de trous de dimension $k + 1$. Ainsi β_0 mesure les trous entre les composantes connexes, donc leur nombre, β_1 mesure le nombre de trous, β_2 le nombre de cavités, etc.

Les constructions précédentes de H_k sont possibles car $\partial\partial = 0$ et par conséquent $\text{Im } \partial_{k+1} \subset \ker \partial_k$ ce qui permet l'utilisation du quotient.

Les éléments du sous espace vectoriel $\ker \partial_k$ s'appellent les k -cycles ou plus simplement les cycles. Ce sous-espace est classiquement noté Z_k . Les éléments de $B_k = \text{Im } \partial_{k+1}$ sont

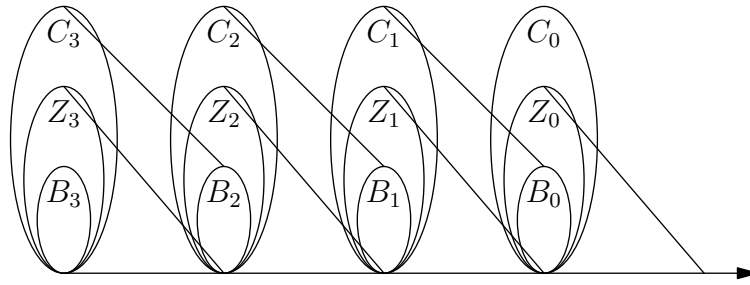


FIGURE 3.4 – Les espaces et sous-espaces vectoriels d'un complexe

appelés les bords. Les bords sont des cycles, $B_k \subset Z_k$ car $\partial\partial = 0$ et par conséquent le groupe $H_k = Z_k/B_k$ regroupe les cycles qui ne sont pas des bords. La figure 3.4 schématise la relation entre les différents espaces vectoriels concernés par l'homologie.

La notation H_\star fait référence à tous les groupes d'homologie pris dans leur ensemble. Ces groupes forment une catégorie [Pie91, ML78] au sens mathématique du terme. Ainsi, si un espace X est transformé en un espace Y , par une fonction continue f , les groupes $H_k(X)$ et $H_k(Y)$ sont aussi reliés par un morphisme de groupe $H_k(f)$.

3.1.3.4 Les groupes d'homologie relative $H_k(X, A)$

Il est parfois utile de ne pas prendre en compte une partie A de l'espace X pour faire le calcul des groupes d'homologie. Nous parlons alors d'homologie relative $H_k(X, A)$. Il faut pour cela réduire complètement le sous-espace A pour qu'il « disparaisse » lors du calcul.

La relation de sous-partie $A \subset X$ est représentée par une relation de sous-complexe A_\star au niveau du complexe C_\star qui est la linéarisation de X . Cela permet ainsi le calcul de $H_k(X, A)$ qui se fera grâce aux complexes C_\star et A_\star .

Réduire A_\star dans C_\star revient à considérer que les chaînes de A_\star sont nulles. Les quotients des espaces vectoriels C_k/A_k jouent ce rôle. Les éléments de ces espaces vectoriels sont des chaînes de C_k donc les cellules de A_k ne sont pas prises en compte. Ainsi, certains cycles peuvent apparaître dans C_k/A_k alors qu'ils n'existaient pas dans C_k et tous les cycles complètement inclus dans A_k disparaissent.

Les groupes d'homologie relative $H_k(X, A)$ sont calculés comme étant les groupes d'homologies qui utilisent ces nouveaux espaces vectoriels C_k/A_k . Il faut toutefois, pour la définition précédente de l'homologie relative et pour leur calcul, que A_\star soit bien un sous-complexe et notamment que le bord d'une chaîne de A_\star soit dans A_\star .

3.1.3.5 Méthode de calcul des groupes d'homologie

Le calcul des groupes d'homologie se fait par réduction de matrice [EH10]. Si les coefficients de l'homologie sont dans un corps alors la réduction est similaire à un pivot de

Gauss et si les coefficients sont des entiers alors une réduction sous forme normale de Smith est nécessaire mais plus demandeuse en coût de calcul. Nous nous restreignons dorénavant au premier cas.

Si l'opérateur de bord est exprimé dans une base adéquate, les générateurs des groupes d'homologie se lisent directement dans la représentation matricielle de ∂ . Comme la matrice de cet opérateur est donnée dans une base non nécessairement intéressante, il faut faire un changement de base intelligent qui est fourni par une réduction matricielle.

Cette réduction est calculable par un algorithme équivalent à une réduction de Gauss, et n'utilise que des opérations élémentaires sur les colonnes de la matrice ∂ . Le pseudo-code est donné sur l'algorithme 3.1 où $M_{.i}$ signifie la i -ème colonne de la matrice M .

Ces opérations élémentaires peuvent s'écrire sous forme de multiplication matricielle à droite et peuvent être inversées sous la même forme. Celles-ci sont :

- ajouter un multiple de la colonne j à la colonne i ;
- échanger les colonnes i et j ;
- multiplier la colonne i par un scalaire α .

Grâce à la réduction de ∂ nous pouvons trouver deux matrices \mathbf{Q} et \mathbf{V} telles que $[\mathbf{Q} \mid \mathbf{0}] = \partial_k \mathbf{V}$ où les colonnes de \mathbf{Q} sont indépendantes entre elles. Plus précisément, la décomposition précédente s'écrit sous la forme

$$\underbrace{\left[\begin{array}{c|c|c|c|c|c|c} \hline | & & | & | & | & | & | \\ \hline b_1 & \cdots & b_n & 0 & \cdots & 0 & \\ \hline \end{array} \right]}_{\substack{\underbrace{\hspace{10em}}_{B_{k-1}} \quad \underbrace{\hspace{10em}}_{\partial Z_k} \\ \dim C_{k-1}}} = \partial_k \underbrace{\left[\begin{array}{c|c|c|c|c|c|c} \hline | & & | & | & | & | & | \\ \hline v_1 & \cdots & v_n & z_1 & \cdots & z_m & \\ \hline \end{array} \right]}_{\substack{\underbrace{\hspace{10em}}_{\partial^{-1} B_{k-1}} \quad \underbrace{\hspace{10em}}_{Z_k} \\ \dim C_k}}. \quad (3.8)$$

Par une lecture directe, les vecteurs z_1, \dots, z_m forment une base des cycles Z_k . De même les vecteurs b_1 à b_n sont indépendants par construction et génèrent tous les bords de dimension $k - 1$.

Nous remarquons, grâce à cette décomposition, la relation $\dim C_k = \dim B_{k-1} + \dim Z_k$ qui permet de calculer autrement la caractéristique d'Euler-Poincaré,

$$\chi(C_\star) = \chi(H_\star(X)) = \sum_k (-1)^k \dim H_k = \sum_k (-1)^k \beta_k. \quad (3.9)$$

La réduction précédente appliquée aux opérateurs de bord permet de récupérer une base pour tous les B_k et les Z_k . Le calcul de $H_k = Z_k/B_k$ consiste à réduire la base de Z_k sur celle de B_k . Cette réduction s'obtient par exemple en mettant sous forme échelonnée réduite en colonne la matrice $[\mathbf{B} \mid \mathbf{Z}]$ — où les colonnes de \mathbf{B} et \mathbf{Z} sont les vecteurs des bases de B_k et Z_k respectivement — pour récupérer une matrice sous la forme $[\mathbf{B}' \mid \mathbf{H} \mid \mathbf{0}]$ où les colonnes de \mathbf{H} sont des représentants générateurs indépendants de H_k .

Toutes les réductions précédentes peuvent s'effectuer en une seule opération sur une grande matrice regroupant toute les matrices de bord. La seule réduction de cette grande matrice

Algorithme 3.1 : Réduction sous forme échelonnée réduite par colonne

Input : Matrice de bord ∂ de taille $m \times n$ **Output** : Matrices Q et V telles que $Q = \partial V$ et Q sous forme réduite et le rang de ∂ **Data** : rang**Data** : premiers : tableau de n indices

```

1  $Q \leftarrow \partial$ 
2  $V \leftarrow I_n$ 
3 rang  $\leftarrow 0$ 
4 for  $i \leftarrow 1$  to  $m$  do
5    $c \leftarrow i$ -ème colonne de  $Q$ 
6   /* Réduction du vecteur  $c$  */
7   for  $j \leftarrow 1$  to rang do
8     if  $c_{\text{premiers}[j]}$  est non nul then
9       /* Mise à zéro de  $c_{\text{premiers}[j]}$  */
10       $c \leftarrow c - c_{\text{premiers}[j]} Q_{:j}$ 
11       $V_{:i} \leftarrow V_{:i} - c_{\text{premiers}[j]} Q_{:j}$ 
12    end
13  end
14   $Q_{:i} \leftarrow c$ 
15  /* Mise à jour du rang si nécessaire */
16  if  $c \neq 0$  then
17    /*  $c$  est un vecteur indépendant des précédents */
18    rang  $\leftarrow$  rang + 1
19    premiers[rang]  $\leftarrow$  indice du premier coefficient non nul de  $c$ 
20    Trier par ordre croissant premiers
21    Effectuer les mêmes permutations sur les colonnes de  $V$ 
22  end
23 end

```

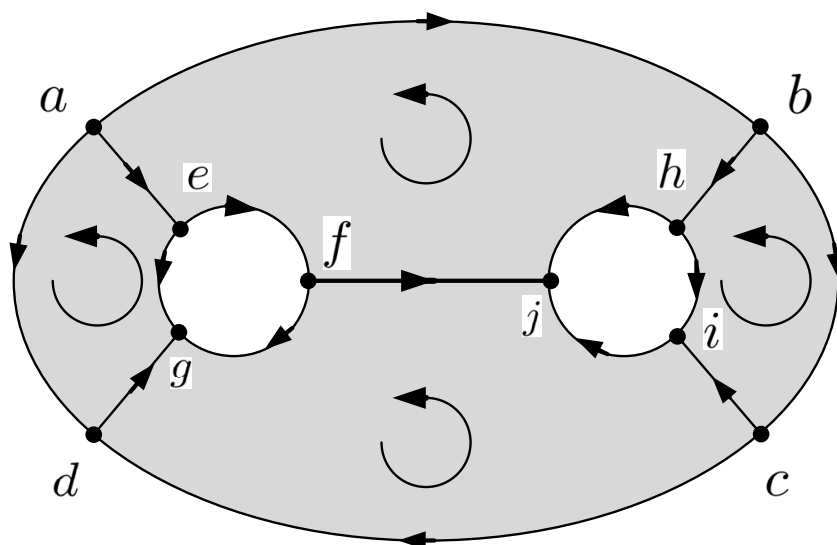


FIGURE 3.5 – Exemple de surface à deux trous

permet de récupérer directement les groupes d'homologie. Il faut remarquer que cette approche est utilisable car les matrices de bord sont des matrices très creuses. En utilisant cette spécificité et les algorithmes de réduction adaptés, le résultat reste manipulable en mémoire d'un ordinateur.

3.1.3.6 Exemple complet de calcul de groupes d'homologie

Afin de rendre les explications ci-dessus plus concrètes, nous nous proposons de faire un calcul des groupes H_0 , H_1 et H_2 du complexe cellulaire de la figure 3.5. Intuitivement, cet espace est en un seul morceau avec deux trous et ne présente aucune cavité ; mathématiquement $\beta_0 = 1$, $\beta_1 = 2$ et $\beta_2 = 0$.

Le type de complexe utilisé dans notre exemple est un Δ -complexe [Hat01], et il faut bien remarquer qu'il y a identification des bords et des sommets. Ces complexes sont simples à manipuler et nécessitent souvent moins de cellules pour représenter un espace.

Le complexe C_\star est composé de trois espaces vectoriels. Nous ne ferons la distinction par la notation entre une cellule c et son vecteur associé (c).

- L'espace vectoriel C_0 est de dimension 10 et est généré par les sommets (a) à (j). Une chaîne est donc de la forme $c_a(a) + c_b(b) + c_c(c) + \dots + c_j(j)$. Le vecteur $(b) + 4(h)$, par exemple, représente la réunion du sommet b et du sommet h avec une multiplicité de 4 ;
- C_1 de dimension 15 est généré par les chaînes $(ab), (bc), \dots$, associées respectivement aux arêtes ab, bc, \dots . Par exemple, la chaîne $(ab) + (bc) + (cd) + (da)$ représente le tour du cercle extérieur ;
- C_2 de dimension 4 a pour base les chaînes $(abhjfe), (bhic), (cijfgd)$ et $(aegd)$ correspondant aux différentes cellules $abhjfe, bhic, cijfgd$ et $aegd$.

Nous remarquons que la caractéristique d'Euler-Poincaré de l'exemple vaut $\chi(X) = \dim C_0 - \dim C_1 + \dim C_2 = 10 - 15 + 4 = -1$ et son genre 1.

Dans la base naturelle des cellules, les matrices de bord sont les matrices d'incidence sommets-arêtes ou arêtes-triangles. Toutefois les coefficients seront pris entiers dans notre exemple pour prendre en compte l'orientation des cellules qui a été fixée et représentée par les flèches sur la figure 3.5.

Le bord de l'arête (ab) est formé de ses sommets a et b mais en prenant en compte l'orientation, nous avons $\partial_1(ab) = (b) - (a)$. Ce même processus permet d'obtenir la matrice de bord ∂_1 ,

$$\partial_1 = \begin{matrix} & (ab) & (ad) & (ae) & (bc) & (bh) & (cd) & (ci) & (dg) & (ef) & (eg) & (fg) & (fj) & (hi) & (hj) & (ij) \\ \begin{matrix} (a) \\ (b) \\ (c) \\ (d) \\ (e) \\ (f) \\ (g) \\ (h) \\ (i) \\ (j) \end{matrix} & \left[\begin{array}{cccccccccccccccc} - & - & - & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ + & \cdot & \cdot & - & - & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & + & \cdot & - & - & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & + & \cdot & \cdot & \cdot & + & \cdot & - & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & - & - & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & - & - & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & - & - & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & - \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & + & + \end{array} \right] & \end{matrix} \quad (3.10)$$

où $-$ indique l'entrée -1 , $+$ correspond au coefficient $+1$ et un point à 0 .

L'orientation est choisie pour que, à partir des notations, la flèche aille du sommet d'identifiant le plus vers le sommet d'identifiant le plus grand selon l'ordre alphabétique. Cette convention permet d'éviter les difficultés liées à la mémorisation de l'orientation et nos mises en œuvre sur machine la suivent.

De même en suivant l'orientation définie sur la figure 3.5, l'opérateur de bord est repré-

sentée par la matrice

$$\mathbf{\partial}_2 = \begin{matrix} & & & & (abhjfe) & (adeg) & (bchi) & (cdfgij) \\ \begin{matrix} (ab) \\ (ad) \\ (ae) \\ (bc) \\ (bh) \\ (cd) \\ (ci) \\ (dg) \\ (ef) \\ (eg) \\ (fg) \\ (fj) \\ (hi) \\ (hj) \\ (ij) \end{matrix} & \left[\begin{array}{cccc} - & \cdot & \cdot & \cdot \\ \cdot & + & \cdot & \cdot \\ + & - & \cdot & \cdot \\ \cdot & \cdot & - & \cdot \\ - & \cdot & + & \cdot \\ \cdot & \cdot & \cdot & - \\ \cdot & \cdot & - & + \\ \cdot & + & \cdot & - \\ + & \cdot & \cdot & \cdot \\ \cdot & - & \cdot & \cdot \\ \cdot & \cdot & \cdot & + \\ + & \cdot & \cdot & - \\ \cdot & \cdot & + & \cdot \\ - & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & + \end{array} \right] \end{matrix}. \quad (3.11)$$

L'orientation est choisie selon le sens trigonométrique, toutes les arêtes du bord d'une cellule qui suivent ce sens sont comptabilisées positivement et les autres négativement.

À partir de la matrice de bord $\mathbf{\partial}_1$, grâce à une réduction des colonnes sous forme d'échelon, nous avons une représentation de la forme donnée par l'équation (3.8). En appliquant cette réduction à (3.10), nous trouvons la décomposition $\mathbf{Q}_1 = \mathbf{\partial}_1 \mathbf{V}_1$ donnée par,

$$\begin{matrix} \begin{matrix} (a) \\ (b) \\ (c) \\ (d) \\ (e) \\ (f) \\ (g) \\ (h) \\ (i) \\ (j) \end{matrix} \left[\begin{array}{cccccccccccc} + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ - & + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & - & - & + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & - & + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & - & - & - & + & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & - & + & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & - & \cdot & \cdot & \cdot \end{array} \right] = \mathbf{\partial}_1 \begin{matrix} \begin{matrix} (ab) \\ (ad) \\ (ae) \\ (bc) \\ (bh) \\ (cd) \\ (ci) \\ (dg) \\ (ef) \\ (eg) \\ (fg) \\ (fj) \\ (hi) \\ (hj) \\ (ij) \end{matrix} \left[\begin{array}{cccccccccccc} - & + & + & \cdot & - & - & - & \cdot & + & \cdot & \cdot & + & \cdot & + & + \\ \cdot & - & - & + & \cdot & \cdot & + & \cdot & \cdot & - & - & - & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & - & + & + & \cdot & - & + & + & \cdot & \cdot & - & \cdot & - \\ \cdot & \cdot & + & \cdot & \cdot & \cdot & - & + & \cdot & \cdot & + & - & \cdot & + & \cdot \\ \cdot & \cdot & \cdot & \cdot & - & - & - & + & \cdot & \cdot & \cdot & \cdot & + & + & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & - & + & \cdot & \cdot & \cdot & - & \cdot & + \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & + \end{array} \right] \end{matrix} \end{matrix} \quad (3.12)$$

Nous remarquons dans un premier temps que les neuf premières colonnes de \mathbf{Q}_1 sont non nulles et indépendantes entre elles, car sous forme échelonnée. Ces colonnes représentent

Autrement dit, H_0 est un espace vectoriel de dimension 1 généré par la classe d'équivalence $(i) + B_0$. C'est-à-dire que le complexe est composé d'une seule composante connexe.

Le groupe H_1 est, quant à lui, de dimension 2 et généré par les deux vecteurs indépendants $(ef) + (fg) - (eg) + B_1$ et $(hi) + (ij) - (hj) + B_1$. Les deux cycles représentatifs sont $u = (ef) + (fg) - (eg)$ et $v = (hi) + (ij) - (hj)$ qui entourent bien les deux trous de l'espace de la figure 3.5.

Le groupe H_2 est nul car il n'y a pas de cycles de dimension 2 : $Z_2 = 0$. Autrement dit il n'y a pas de cavité dans l'espace de l'exemple.

Il faut remarquer que l'ordre des colonnes des matrices de bord ∂_1 et ∂_2 influence énormément les représentants obtenus par notre algorithme. En effet, il est possible que l'algorithme retourne, tout en restant correct, que H_0 soit généré par $(a) + B_0$.

De même, un changement d'ordre des cellules pourrait fournir comme représentants de H_1 les vecteurs $(ab) + (bc) + (cd) - (ad)$ et $(ef) + (fg) - (eg)$. Ici, le premier cycle « entoure » les deux trous et le second uniquement le trou à gauche de la figure 3.5. Il n'y en en effet aucun ensemble de représentant canonique, c'est-à-dire naturellement défini. De même aucune base n'est canonique, par exemple $\{u, v\}$ forme une base tout aussi valide que $\{u + v, v\}$ ou $\{u, u + v\}$.

La dimension de H_0 et celle de H_1 donnent respectivement les nombres de Betti $\beta_0 = 1$ et $\beta_1 = 2$. La caractéristique d'Euler-Poincaré est donc $\beta_0 - \beta_1 = -1$; qui est en accord avec le calcul impliquant la dimension des espaces de chaînes C_k .

3.1.4 Persistance homologique

Le complexe C_\star est supposé jusqu'ici fixe. Toutefois les cellules peuvent être ajoutées les unes après les autres à des instants différents. La suite de ces ajouts de cellules se présente alors sous forme d'un « film » dans lequel le complexe se construit petit à petit par ajout successif de cellule. À chaque instant le complexe doit rester valide, il faut que le bord d'une cellule du complexe soit déjà dans ce complexe. La construction du complexe instant par instant s'appelle la filtration. Naturellement plusieurs filtrations peuvent exister pour un même complexe.

Une cellule de dimension k ajoutée à un instant t peut créer un nouveau k -cycle, c'est-à-dire un nouveau « trou », ou détruire un $k + 1$ cycle pré-existant. La persistance homologique développée dans [ELZ02] et [ZC05] permet d'obtenir un résumé des cycles de leur instant de naissance à celui de leur mort. Ainsi un cycle ayant une longue durée de vie sera topologiquement plus informatif alors qu'un cycle de faible durée de vie sera dû à du bruit par exemple en analyse topologique de données [Car14]. Ce leitmotiv est supporté par la stabilité du diagramme de persistance aux modifications de la filtration [CSEH07].

Pour exemple, une demi centaine de points sont choisis sur deux cercles de rayon 1.7 et 1.2 avec un bruit complexe gaussien affectant leur position. Ces points sont illustrés sur la figure de gauche de la figure 3.6.

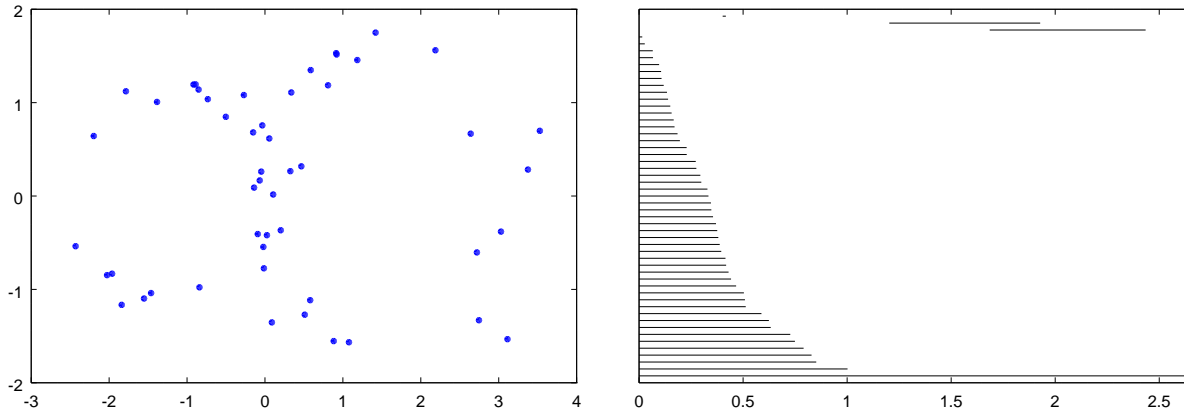


FIGURE 3.6 – Exemple de données pour la persistance

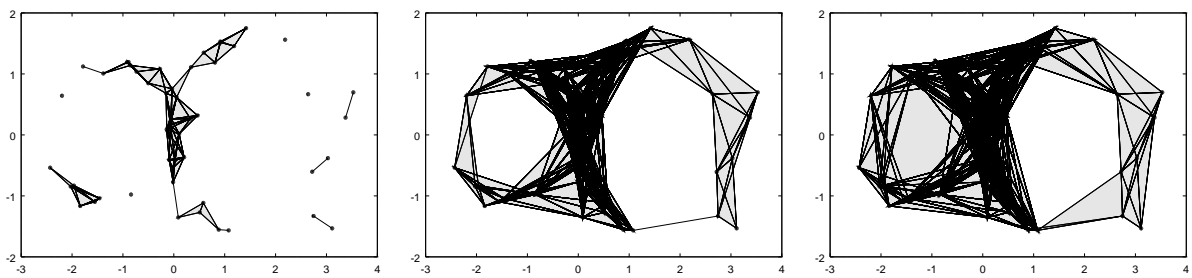


FIGURE 3.7 – Filtration du complexe pour les distances 0.6, 1.8 et 2

À partir de ces points un complexe est construit en fonction du temps. Initialement, tous les points sont des sommets du complexe puis les arêtes sont ajoutées par exemple de la plus petite à la plus grande. Si trois sommets sont reliés pendant ce processus, le triangle correspondant est ajouté au complexe. Ainsi la construction du complexe se fait en fonction d'un paramètre qui représente la distance maximale des sommets de chaque simplexe. La figure 3.7 donne trois arrêts sur image du film de la construction du complexe lorsque la distance maximal des sommets des simplexes est inférieure respectivement à 0.6, 1.8 et 2. Plus la distance maximale augmente plus le complexe est complet jusqu'à la limite où tous les sommets sont connectés. Il semble naturel que les trous des cercles ne seront pas comblés rapidement mais que les connections se feront de proche en proche. Par conséquent les cycles représentant les deux « trous » resteront vivants suffisamment longtemps pour être détectés par la persistance homologique.

Le complexe et la filtration en découlant sont utilisés pour calculer la persistance dont le code-barre [Ghr08a] est affiché à droite de la figure 3.6. Les barres qui sont en bas concernent H_0 et celles du haut concernent H_1 . La vie d'un cycle de sa naissance à sa mort est représentée par un intervalle $[a, b[$ où a et b valent respectivement les valeur du paramètre de filtration à l'instant de la création du cycle et à l'instant de sa destruction. À droite de la figure 3.6, chacun de ces intervalles est représenté par un segment entre les abscisses a et b .

Les segments en partie basse de la figure sont associés aux cycle de H_0 , c'est-à-dire aux composantes connexes. Initialement, chaque sommet est isolé, puis ils se connectent peu à peu entre eux via des arêtes. À chaque connexion entre deux composantes connexes, un cycle de H_0 disparaît terminant ainsi une segment du code-barre. Lorsque tout les sommets sont connectés entre eux, il ne reste que l'ultime segment en bas de la figure 3.6.

Le nombre de barres pour H_1 est plus réduit. Les deux plus grandes barres, d'intervalle $[1.2; 1.9]$ et $[1.7; 2.4]$, représentent les deux cercles détectables *de visu*. La persistance homologique permet donc de repérer ces caractéristiques à partir d'un simple échantillon. Cela est à la base de l'étude topologique de l'espace des images naturelles [CIS08]. La barre plus petite de H_1 autour de l'abscisse 0.4 est un petit cycle créé par la disposition bruitée des points. Il est le résultat d'un bruit topologique qui peut être facilement filtré par la largeur de l'intervalle correspondant. Certains travaux [BFL⁺13, BMS14] ont introduit une modélisation probabiliste des code-barres pour pouvoir faire de l'inférence dont des tests d'hypothèses sur la pertinence des cycles.

Des algorithmes rapides de calcul de la persistance homologique ont vu le jour dernièrement. Le premier [EH10, Zom05] est une réduction de matrice optimisée. Cette version a été améliorée dans [CK11] qui reste très simple et suffisamment rapide dans la plupart des cas. D'autres algorithmes profitent des dualités entre les groupes d'homologie et de cohomologie [dSMVj11a] ou réduisent le complexe initial avant le calcul effectif [MW10, MN13, AKL13]. Il y a aussi la possibilité de distribuer l'algorithme pour plus de rapidité [BKR14].

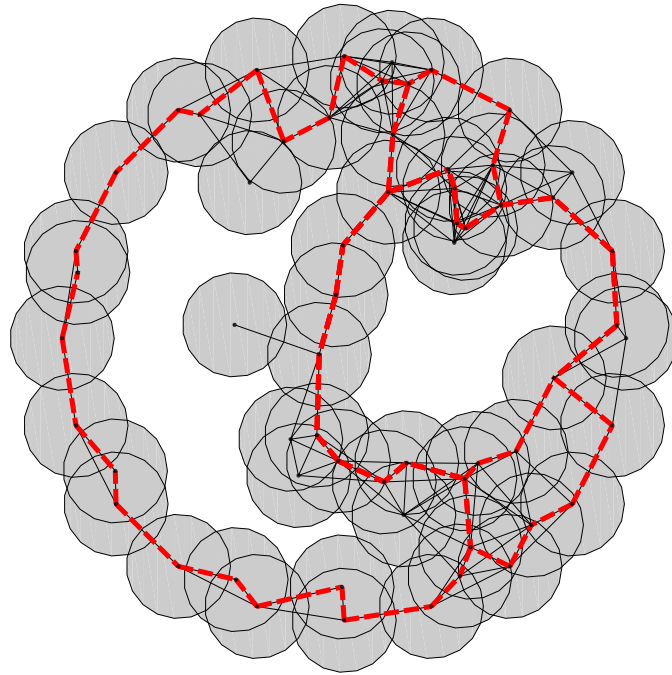


FIGURE 3.8 – Complexe d'un RCSF selon la méthode de de Silva et Ghrist

3.1.5 L'homologie au secours des RCSF

De Silva et Ghrist proposent dans [dSG06] d'utiliser les groupes d'homologie pour détecter des trous de couverture et ils étendent leur méthode dans [dSG07a] en utilisant la persistance. Les idées de base de ces approches ont aussi été introduites dans [GM05] et l'utilisation de capteurs mobiles est considérée dans [dSGM05].

Nous présentons ci-dessous leur construction et les deux critères, l'un évident, le second moins mais plus informatif puis nous utiliserons leur résultat pour présenter une application sur la gestion de l'énergie qui sera reprise au prochain chapitre.

3.1.5.1 La construction de de Silva et de Ghrist

Il est en effet assez naturel d'utiliser l'homologie pour les problèmes de couverture en RCSF. Il faut dans un premier temps construire un espace qui approche le champ de couverture puis en calculer l'homologie H_1 . Nous utiliserons la même méthode à la section 3.3 mais avec un modèle de couverture différent.

Le modèle de couverture de de Silva et Ghrist est particulièrement simple : les nœuds du RCSF captent le champ autour d'eux sur un rayon prédéterminé r_c . De plus les nœuds peuvent communiquer entre eux sur une distance r_b .

Le complexe représentant la couverture est construit à partir du graphe des communications. Les sommets du complexe sont les nœuds du réseau. Si deux nœuds sont à une distance suffisamment faible pour communiquer, c'est-à-dire inférieure à r_b , alors l'arête entre les sommets est aussi ajouté au complexe. Enfin si trois ou quatre nœuds peuvent communiquer deux à deux alors le triangle ou le tétraèdre est aussi ajouté au complexe. Cette construction s'appelle le complexe de Rips de distance r_c .

La figure 3.8 représente un complexe construit selon la procédure précédente. Une soixantaine de capteurs sont disposés dans le disque unité dont une vingtaine sur le pourtour. Les disques gris indique le champ couvert par chaque capteur et les arêtes indiquent quels capteurs sont en communication directe.

Si le rayon de couverture r_c est suffisamment grand — plus précisément si $r_c \geq r_b/\sqrt{3}$ — alors le complexe de communication construit précédemment est une approximation suffisante du champ de couverture. Selon cette condition, si aucun trou n'est détecté alors la couverture est bonne. L'article [YMD12] fait une étude plus avancée de la précision de la méthode avec ce modèle.

Nous avons aussi tracé en traits forts tiretés rouge sur la figure 3.8 les générateurs des cycles découvert par le calcul du groupe d'homologie H_1 . Nous remarquons que les trous de couverture sont bien détectés par la méthode. Le petit cycle en haut à droite entoure lui aussi un trou de couverture qui est à peine visible. Cependant le petit cycle en bas à droite de la figure n'entoure pas un véritable trou de couverture ; ces cas pathologiques sont des faux positifs générés par la méthode. Ces cas restent heureusement rares et ne concernent souvent qu'une petite partie du réseau. Ils sont principalement dus à la difficulté d'inférer la couverture à partir du rayon de communication. Notons toutefois que cette méthode ne produit pas de faux négatifs. En effet, le critère développé dans [dSG06, dSG07a] est une contraposée. Il indique que s'il n'y a pas de trou de couverture alors le groupe d'homologie relative $H_2(C_*, F)$ est non vide où F est la bordure du RCSF.

Le critère précédent apporte une nouvelle information. En effet si $H_2(C_*, F)$ n'est pas vide alors il n'a qu'un générateur possible. Un représentant du générateur est composé de triangles. L'ensemble de ces triangles garantit la couverture encerclé par la bordure F . Par conséquent, les sommets qui n'interviennent pas dans les triangles d'un représentant de l'unique générateur de H_2 , sont suffisants pour satisfaire la contrainte de couverture. Les autres sommets peuvent donc être éteints sans crainte.

La figure 3.9 affiche en gris la couverture par les capteurs éveillés, en bleu tiretés la bordure et les 94 nœuds éteints sont en rouge. Nous remarquons qu'il n'y a effectivement pas de trou dans la couverture en utilisant uniquement les capteurs non éteints.

Les arêtes dessinées sont incidentes à deux des 306 sommets éveillés. Nous remarquons que l'extinction des nœuds surnuméraires permet de satisfaire la couverture tout en simplifiant le sous-complexe représentatif.

En étudiant plus précisément le complexe, nous pouvons encore espérer le simplifier tout en garantissant la couverture. En effet, le calcul d'un représentant du générateur du groupe

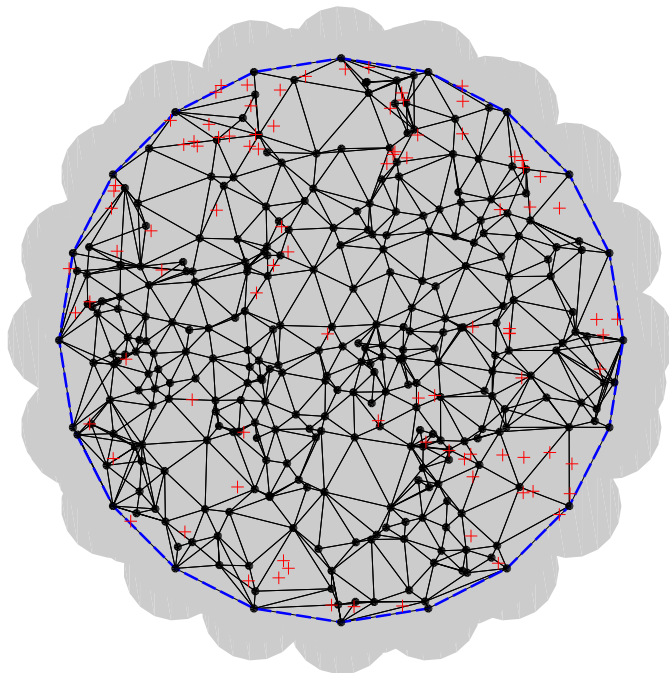


FIGURE 3.9 – Extinction de capteurs avec couverture certifiée

d'homologie $H_2(C_\star, F)$ n'est pas optimisé. Cet aspect sera l'occasion d'une discussion et des solutions à ce problème seront décrites à la section 3.3.6.2.

3.1.5.2 Application simple à la gestion de l'énergie d'un RCSF

Donnons un avant-goût de l'utilisation du critère de de Silva et de Ghrist pour la mise en veille dans un RCSF par un exemple qui cherche à limiter le nombre de capteurs éveillés tout en garantissant la couverture. Cette application sera expliquée plus en détail et développée pour une version distribuée dans la section 4.4 du chapitre suivant.

La figure 3.10 présente un réseau de 211 capteurs déployés dans le cercle unité X dont 21 d'entre eux forment la bordure F du champ de couverture. Chaque capteur couvre un disque de rayon 0.15.

Le complexe C_\star qui représente la couverture est le complexe de clique du graphe de connexion où deux capteurs sont connectés si leur distance est inférieur à 0.3. Le calcul d'un générateur du groupe d'homologie $H_2(X, F) = H_2(C_\star, F)$ se fait en utilisant une variante de l'algorithme de persistance avec le twist de [CK11].

À chaque instant un capteur éveillé épuise la totalité de son énergie et disparaît du réseau et donc du complexe C_\star associé. Le groupe $H_2(C_\star, F)$ est alors recalculé.

Les vignettes de la figure 3.10 sont des instantanés de l'état du RCSF à différents instants,

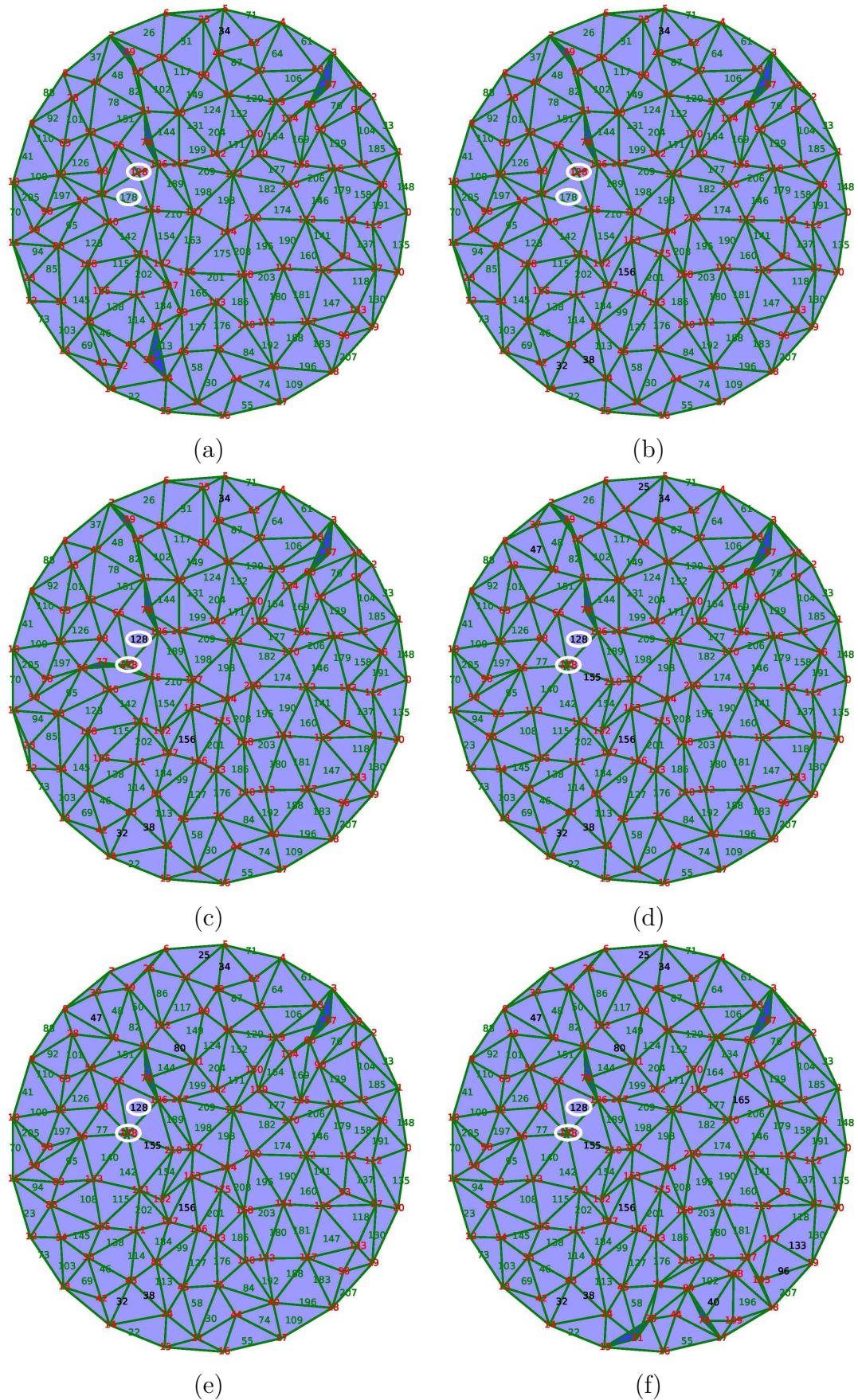


FIGURE 3.10 – Instantanés des capteurs éteints et éveillés

plus précisément aux itérations 1, 4, 5, 8, 9 et 13 (a–f). Les étiquettes rouges indiquent l'emplacement des capteurs qui sont éveillés alors que le vert correspond aux capteurs endormis ; enfin les étiquettes noires représentent les capteurs éteints car à court d'énergie. Cette classification des capteurs est obtenue en récupérant les nœuds intervenant dans un représentant du générateur de $H_2(C_*, F)$.

Nous remarquons que l'algorithme est stable et suffisamment performant pour endormir plus de la moitié des capteurs tout en gardant la couverture intacte. Aucune optimisation n'est faite et la qualité de ce représentant dépend fortement de la mise en œuvre de l'algorithme de calcul d'homologie.

Seuls les nœuds nécessaires à la couverture sont réveillés et ceci de façon très localisée. Par exemple, lorsque le nœud 128 disparaît du réseau, seul le nœud 178 est réveillé pour garder la couverture complète. Notons que ce nœud 178 était un voisin direct du capteur 128.

Cette propriété de localisation sera utilisée au prochain chapitre pour pouvoir rendre cette application utilisable sans devoir centraliser les calculs pour le calcul du groupe $H_2(C_*, F)$ et en limitant les calculs locaux. C'est pourquoi nous détaillerons cette application à la section 4.4.

3.2 Éléments sur les processus gaussiens

Les nœuds des RCSF captent leur environnement en mesurant divers paramètres physiques à leur emplacement. Par conséquent les seules informations réellement disponibles concernent un ensemble de mesures restreint aux seules positions des capteurs.

Pour pouvoir inférer des résultats concernant l'ensemble du champ de captation du RCSF, il est nécessaire de faire intervenir un modèle qui permet le passage des données locales aux résultats globaux.

Plusieurs modélisations sont possibles. La plus simple est de considérer que la mesure d'un capteur correspond à la valeur autour de ce capteur sur un rayon fixé. Ce type de modèle est souvent considéré de façon implicite dans la littérature.

D'autres modèles qui reposent sur les phénomènes physiques mis en jeu peuvent être proposés. Par exemple un modèle simple de propagation selon l'équation de la chaleur permet d'obtenir une interpolation intelligente de la température. Naturellement les traitements spatiaux des données selon ces modèles physiques peuvent être très demandeurs de ressources calculatoires et ne peuvent se faire que rarement dans le réseau.

La difficulté de cette modélisation qui permet le passage du local au global est due aux phénomènes d'échantillonnage spatial non uniforme. Comme nous ne sommes pas dans un cadre uni-dimensionnel, le théorème de Shannon–Nyquist ne peut pas être appliqué simplement. Il existe cependant des résultats et des cadres qui permettent de travailler sur ces aspects — notamment grâce aux modèles physiques. Certaines de ces méthodes

sont mêmes issues de la topologie algébrique et qui reposent notamment sur la notion de faisceau [Rob13b, Rob14a, Rob14b] qui indique le niveau d’ambiguïté des mesures et les obstructions à une parfaite reconstruction spatiale du champ.

Le modèle utilisé dans ce chapitre est un modèle probabiliste très souple qui permet de prendre en compte un grand nombre de situations tout en autorisant des calculs effectifs : les processus gaussiens.

Le choix de cette modélisation n’est pas nouvelle [GH12, YXCO11] pour les RCSF et elle s’explique par l’omniprésence des processus gaussiens en modélisation statistique [Abr97] et en traitement du signal [RW05] ou des images.

3.2.1 Définition

Un processus gaussien paramétré par un espace X à valeur dans un espace euclidien E est une « fonction aléatoire » $f(x)$ où x appartient à X et $f(x)$ est dans E . La référence d’Adler et Taylor [AT07] en donne une définition plus rigoureuse.

Nous utiliserons tout d’abord ces processus gaussiens pour modéliser le champ physique d’intérêt pour l’application. Ainsi, le capteur localisé au point x_i mesurera $f(x_i)$ auquel s’ajoute un bruit, souvent considéré lui aussi gaussien.

Si on suppose que le champ mesuré par le réseau de capteurs ne concerne qu’une seule valeur, l’espace des valeurs de f sera simplement la droite réelle. Il est possible de prendre en compte des champs multiples en modifiant l’espace des valeurs de f .

Le champ aléatoire f est complètement spécifié par deux fonctions $m(x)$ et $k(x, y)$ qui permettent de calculer la loi de f : le vecteur aléatoire $[f(x_1), \dots, f(x_n)]^T$ suit une loi normale de moyenne $[m(x_1), \dots, m(x_n)]^T$ et de matrice de covariance de coefficients $k(x_i, x_j)$; c’est-à-dire, plus symboliquement,

$$\begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(x_1) \\ m(x_2) \\ \vdots \\ m(x_n) \end{bmatrix}, \begin{bmatrix} k(x_1, x_1) & \cdots & k(x_1, x_n) \\ k(x_2, x_1) & & k(x_2, x_n) \\ \vdots & & \vdots \\ k(x_n, x_1) & \cdots & k(x_n, x_n) \end{bmatrix} \right). \quad (3.16)$$

La fonction $m(x)$ représente la partie déterministe du champ ou encore l’*a priori* dans un contexte bayésien. Le noyau k peut prendre différentes formes [Abr97] selon l’application. Souvent la valeur de la covariance dépend de la distance entre les points de X . C’est à dire qu’il est souvent possible dans les applications de considérer que $k(x, y) \propto k(r)$ où r est la « distance » de X . Le coefficient de proportionnalité indique la variabilité du champs en un point.

Noyau gaussien Il est particulièrement bien adapté aux champs dont les corrélations

décroissent vite avec la distance.

$$k(r) \propto \exp\left(-\frac{r^2}{2\ell^2}\right) \quad (3.17)$$

Le noyau gaussien dépend d'un paramètre, la constante d'échelle spatiale ℓ qui mesure la distance minimale nécessaire pour une variation significative du champ. Les champs construits avec le noyau gaussien sont infiniment dérivables.

Noyau de Matérn Il est utilisé pour assurer la $\nu - 1$ -ème dérivabilité. Elle est définie par

$$k(r) \propto \frac{1}{\Gamma(\nu)2^{\nu-1}} \left(\sqrt{2\nu}\frac{r}{\ell}\right)^\nu K_\nu\left(\sqrt{2\nu}\frac{r}{\ell}\right) \quad (3.18)$$

où K_ν est la fonction de Bessel modifiée de seconde espèce. Comme pour le noyau gaussien, ℓ est une constante d'échelle spatiale. Lorsque ν devient grand, le noyau de Matérn tend vers le noyau gaussien.

Noyau exponentiel Ce noyau ne dépend que de la constante d'échelle ℓ ,

$$k(r) \propto \exp\left(-\frac{r}{\ell}\right) \quad (3.19)$$

qui indique une décroissance de la corrélation du champ moins rapide que pour le noyau gaussien.

Noyau rationnel Les processus gaussiens ayant ce noyau sont infiniment dérivables comme pour le noyau gaussien. Mais le paramètre α permet de limiter la vitesse de la décroissance de la corrélation en fonction de la distance.

$$k(r) \propto \left(1 + \frac{r^2}{2\alpha\ell^2}\right)^{-\alpha} \quad (3.20)$$

Comme pour le noyau de Matérn, lorsque α devient grand, le noyau tend vers le noyau gaussien.

Naturellement, la liste précédente n'est pas exhaustive. Il est possible de mélanger les noyaux entre eux pour prendre en compte différents phénomènes physiques. Ainsi, une combinaison linéaire de noyaux est encore un noyau.

De même la fonction de covariance $k(x, y)$ n'est pas nécessairement de la forme $k(r)$ ce qui permet d'introduire une certaine inhomogénéité et/ou anisotropie du champs. Ainsi, les champs présentant une certaine périodicité sont aussi modélisables par des processus gaussiens si le noyau est périodique.

La figure 3.11 montre trois réalisations de processus gaussiens unidimensionnels dont le noyau est gaussien. Comme attendu, si la constante spatiale est grande, les fluctuations sont moins rapides ; alors que de faibles constantes autorisent une plus grande variabilité du champ.

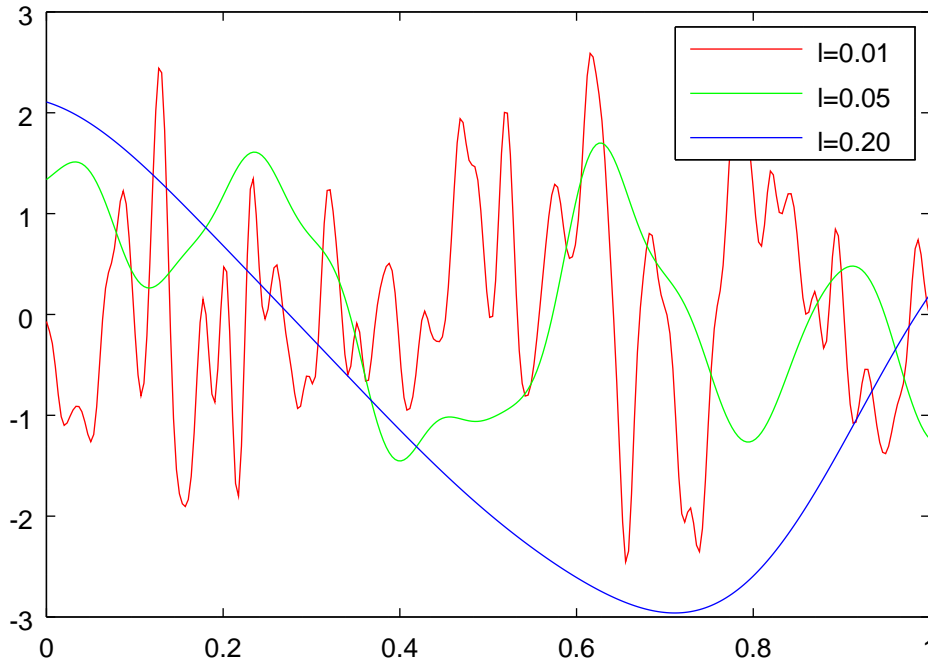


FIGURE 3.11 – Exemples de réalisation de processus gaussiens unidimensionnels

3.2.2 Régression

L'attrait principal de l'utilisation des processus gaussien est sa grande flexibilité de modélisation via les noyaux de covariance mais aussi grâce à la facilité d'effectuer des régressions bayésiennes [GH12].

Supposons que nous ayons une série de m observations $D = \{(x_i, y_i) \mid 1 \leq i \leq m\}$ qui suivent le modèle suivant

$$\begin{aligned} y_i &= f(x_i) + e_i \\ e_i &\sim \mathcal{N}(0, \sigma^2) \end{aligned} \quad (3.21)$$

où f est le champ d'intérêt et σ^2 la variance du bruit de mesure e_i . Naturellement, ces échantillons de bruits sont supposés indépendants les uns des autres.

L'objectif est de retrouver le champ f à partir du jeu de données. Plus précisément, nous recherchons la valeur du champ à n emplacements x_1^*, \dots, x_n^* .

Simplifions les notations par un formalisme vectoriel : $\mathbf{x}^* = [x_1^*, \dots, x_n^*]^T$, $\mathbf{f}^* = [f(x_1^*), \dots, f(x_n^*)]^T$, $\mathbf{x} = [x_1, \dots, x_m]^T$, $\mathbf{f} = [f(x_1), \dots, f(x_m)]^T$ et $\mathbf{y} = [y_1, \dots, y_m]^T = \mathbf{f} + \mathbf{e}$.

Le champ f suit *a priori* un processus gaussien. Par conséquent,

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(\mathbf{x}) \\ m(\mathbf{x}^*) \end{bmatrix}, \begin{bmatrix} k(\mathbf{x}, \mathbf{x}) & k(\mathbf{x}, \mathbf{x}^*) \\ k(\mathbf{x}^*, \mathbf{x}) & k(\mathbf{x}^*, \mathbf{x}^*) \end{bmatrix} \right), \quad (3.22)$$

où la fonction de covariance k est étendue naturellement aux valeurs vectorielles.

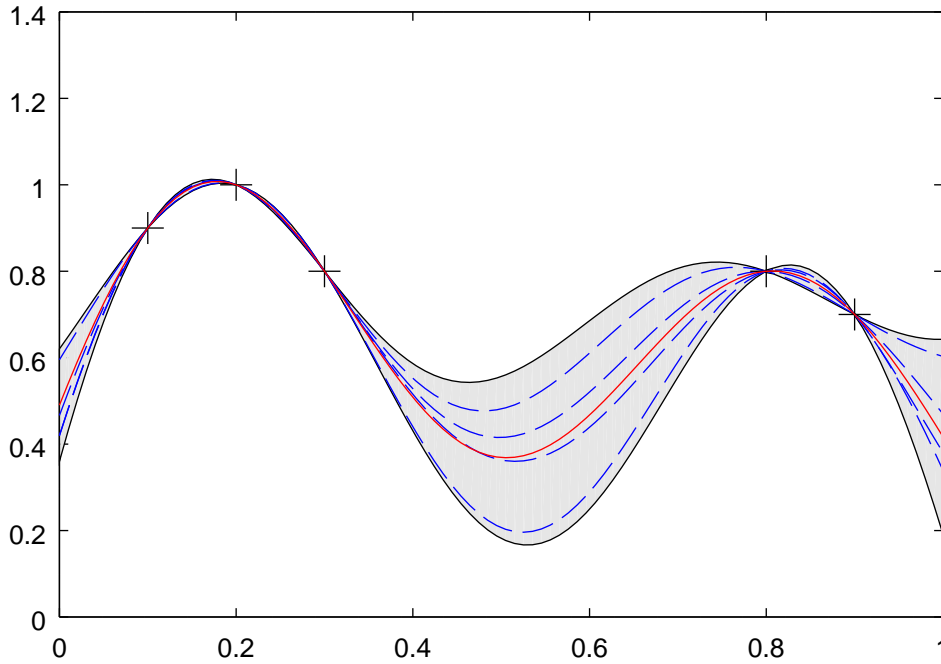


FIGURE 3.12 – Régression par les processus gaussiens unidimensionnels

La stabilité des processus gaussien par combinaison linéaire permet alors d'ajouter les erreurs de mesures au modèle précédent en modifiant la matrice de variance-covariance. Comme ici, les erreurs de mesure sont indépendantes des autres variables aléatoires,

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}^* \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} + \begin{bmatrix} \mathbf{e} \\ \mathbf{0} \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} m(\mathbf{x}) \\ m(\mathbf{x}^*) \end{bmatrix}, \begin{bmatrix} k(\mathbf{x}, \mathbf{x}) + \sigma^2 I & k(\mathbf{x}, \mathbf{x}^*) \\ k(\mathbf{x}^*, \mathbf{x}) & k(\mathbf{x}^*, \mathbf{x}^*) \end{bmatrix} \right). \quad (3.23)$$

Le conditionnement d'une partie d'un vecteur gaussien par sa partie complémentaire est encore gaussien avec une moyenne et une variance exprimées simplement. Par conséquent, les valeurs du champs aux points x_1^*, \dots, x_n^* étant données les observations $f(x_1), \dots, f(x_m)$ est donc un vecteur gaussien vérifiant la loi

$$\mathbf{f}^* | \mathbf{y} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad (3.24)$$

avec

$$\boldsymbol{\mu} = m(\mathbf{x}^*) + k(\mathbf{x}, \mathbf{x}^*) k(\mathbf{x}, \mathbf{x})^{-1} (\mathbf{y} - m(\mathbf{x})), \quad (3.25)$$

$$\boldsymbol{\Sigma} = k(\mathbf{x}^*, \mathbf{x}^*) - k(\mathbf{x}, \mathbf{x}^*) k(\mathbf{x}, \mathbf{x})^{-1} k(\mathbf{x}^*, \mathbf{x}). \quad (3.26)$$

Il est remarquable que dans l'expression ci-dessus, la valeur des mesures \mathbf{y} n'intervient que dans le calcul de la moyenne $\boldsymbol{\mu}$. Cette observation sera à la base de l'application décrite ci-dessous sur la détection des trous de couverture.

La figure 3.12 est un exemple de résultat de la régression présentée ci-dessus. À partir des cinq mesures non bruitées indiquées par des croix, nous avons reconstitué le champ

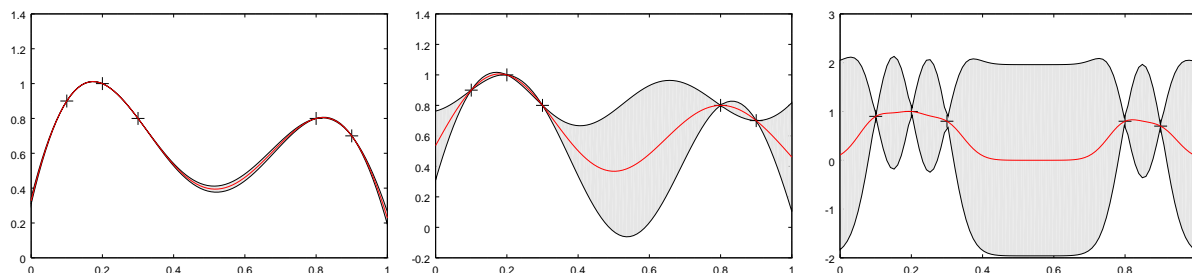


FIGURE 3.13 – Impact des hyper-paramètres sur la régression

moyen — en rouge sur la figure — issu de la régression en utilisant un noyau gaussien de constante spatiale de 0.3. La zone grisée sur la figure correspond à l'intervalle de confiance à 95% de la moyenne de $\mathbf{f}^* \mid \mathbf{y}$ obtenue grâce à l'expression de la matrice de variance Σ . Enfin quelques réalisations de $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ sont aussi tracées en bleu et en traits tiretés.

3.2.3 Calcul des hyper-paramètres

Un processus gaussien est spécifié par la fonction moyenne et la fonction de covariance. Mais ces dernières peuvent être elles-même paramétrées par des hyper-paramètres θ . Par exemple, sur la figure 3.13, les mêmes données ont servi à des régressions utilisant un noyau gaussien mais utilisant trois constantes spatiales différentes qui sont, de gauche à droite, 0.5, 0.25 et 0.05.

Nous remarquons que lorsque la constante spatiale est trop petite, à droite sur la figure 3.13, les mesures ne permettent pas d'inférer la valeur de la fonction autrement qu'au point de mesure. En effet, les valeurs aux points voisins des mesures deviennent très rapidement décorrélés des points de mesure. Par conséquent, seul le modèle *a priori* donne des information.

A contrario, une constante spatiale trop forte contraint fortement le modèle une fois les mesures connues. La partie gauche de la figure 3.13 montre le résultat de la régression avec ce type de constante spatiale. Les corrélations entre un point quelconque et les mesures sont si fortes que la valeur en ce point est quasiment imposée par le modèle.

Les deux cas précédents n'apportent plus d'information une fois les mesures effectuées car dans un cas l'*a priori* prend le dessus et dans l'autre cas ce sont les données qui fixent tout le modèle.

Pour éviter ces écueils, les hyper-paramètres θ du modèle comme la constante spatiale doivent être fixés par les données. Le cadre bayésien de la régression par processus gaussien suggère d'apprendre θ en maximisant $\Pr[\theta \mid \mathbf{x}, \mathbf{y}]$. Si l'*a priori* sur θ est uniforme, le critère d'optimisation est équivalent à la vraisemblance des données étant données les

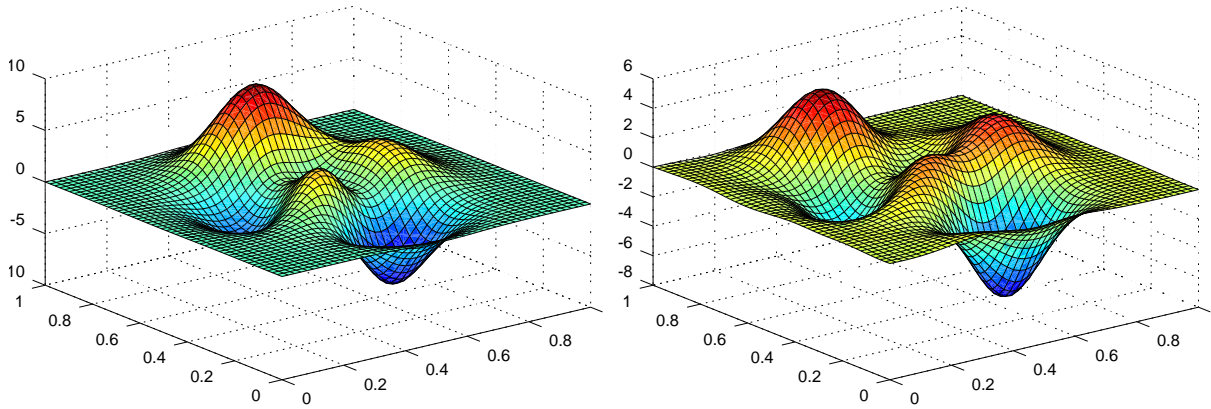


FIGURE 3.14 – Régression d'un champ déterministe par un processus gaussien

hyper-paramètres θ . Cette vraisemblance est donnée par

$$\begin{aligned} \log \Pr[\theta \mid \mathbf{x}, \mathbf{y}] &\propto \log \Pr[\mathbf{y} \mid \mathbf{x}, \theta] \\ &= -\frac{1}{2} \log \det(k(\mathbf{y}, \mathbf{y})) - \frac{1}{2} \mathbf{y}^T k(\mathbf{y}, \mathbf{y})^{-1} \mathbf{y} - \frac{n}{2} \log 2\pi, \end{aligned} \quad (3.27)$$

où $k(\mathbf{y}, \mathbf{y}) = k(\mathbf{x}, \mathbf{x}) + \sigma^2 I$ est la matrice de covariance des mesures, bruit inclus.

La régression en position centrale de la figure 3.13 utilise ce dernier critère pour obtenir la constante spatiale. Intuitivement, nous sentons que cette optimisation permet d'obtenir un équilibre entre l'*a priori* et le modèle.

3.2.4 Exemple de régression d'un champ déterministe

Nous avons utilisé le schéma de régression gaussienne précédent lorsque le champ est déterministe. À partir de 50 points de mesures aléatoirement et uniformément répartis, le champ de mesure est estimé à l'aide d'un processus gaussien, son noyau de corrélation est un noyau gaussien dont les hyper-paramètres sont optimisés selon le critère développé à la section précédente. Les mesures sont bruitées par une erreur additive gaussienne.

La figure 3.14 présente les surfaces du champ déterministe à gauche et son estimation à droite. Visuellement, la régression est performante : la forme générale est retrouvée. Les différences sont principalement dues à la position des mesures : l'estimation est mauvaise lorsque les points de mesures sont trop distants.

Afin d'apprécier les différences entre le champ et son estimation ainsi que leur relation avec la position des points de mesures, les deux champs sont affichés sur la figure 3.15 par des contours. Les points de mesure sont affichés par une croix. Par exemple, les maxima et leur valeur sont mal estimés si aucune mesure n'est prise aux alentours.

La figure 3.16 permet de repérer la précision de l'estimation au travers de la variance. Les courbes de niveau de $10 \log_{10} \tilde{\sigma}^2$ y sont dessinées, où $\tilde{\sigma}^2$ est la variance de l'estimation,

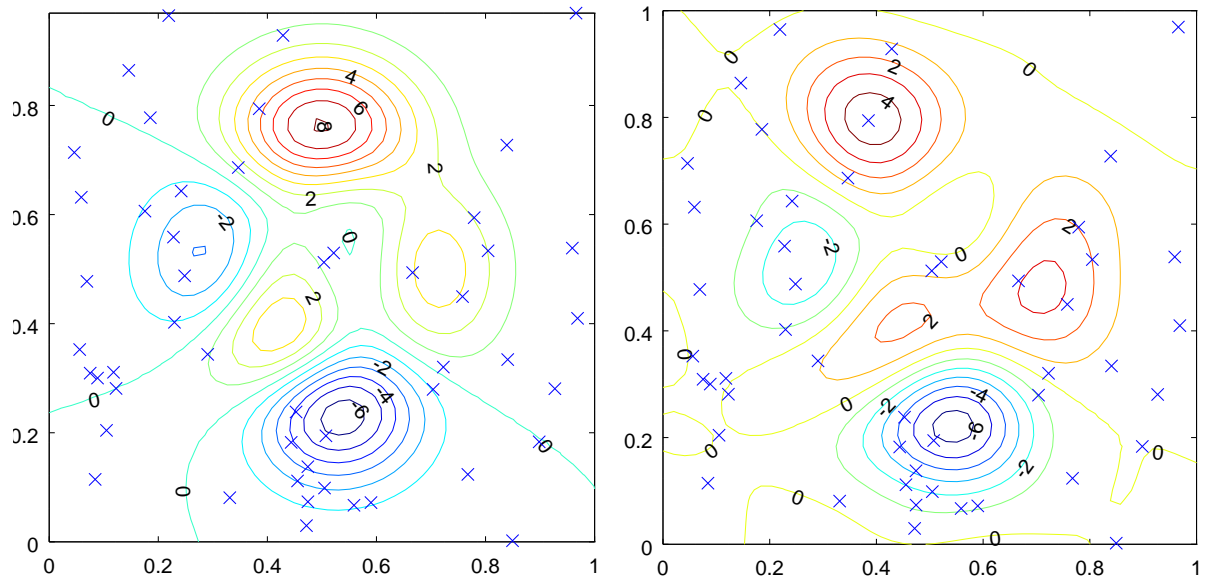


FIGURE 3.15 – Régression d'un champ déterministe par un processus gaussien (contour)

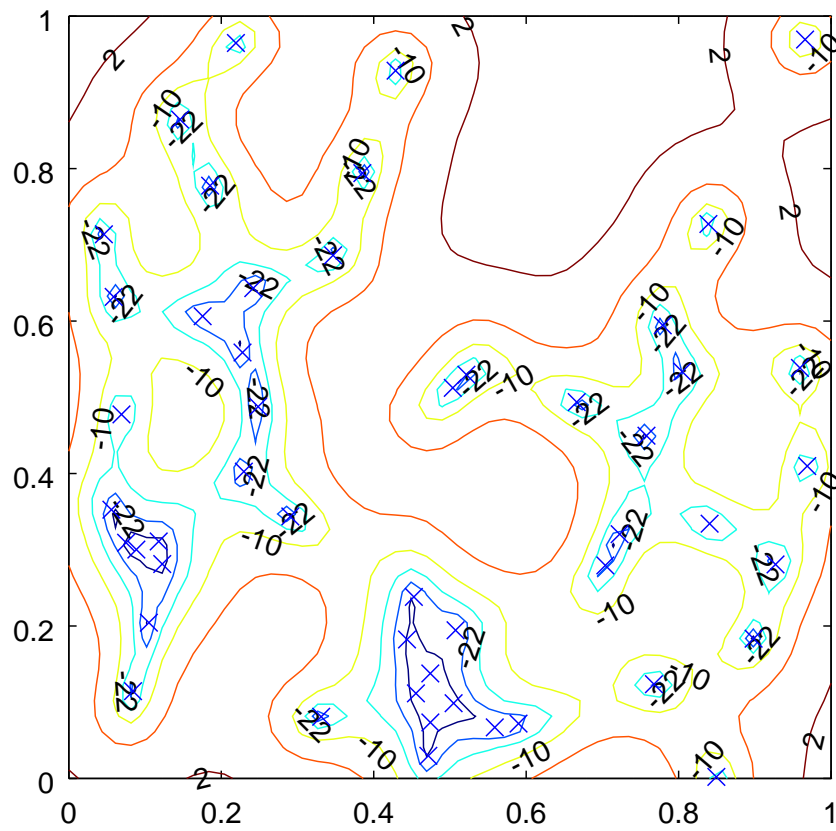


FIGURE 3.16 – Variance de l'estimation du champ

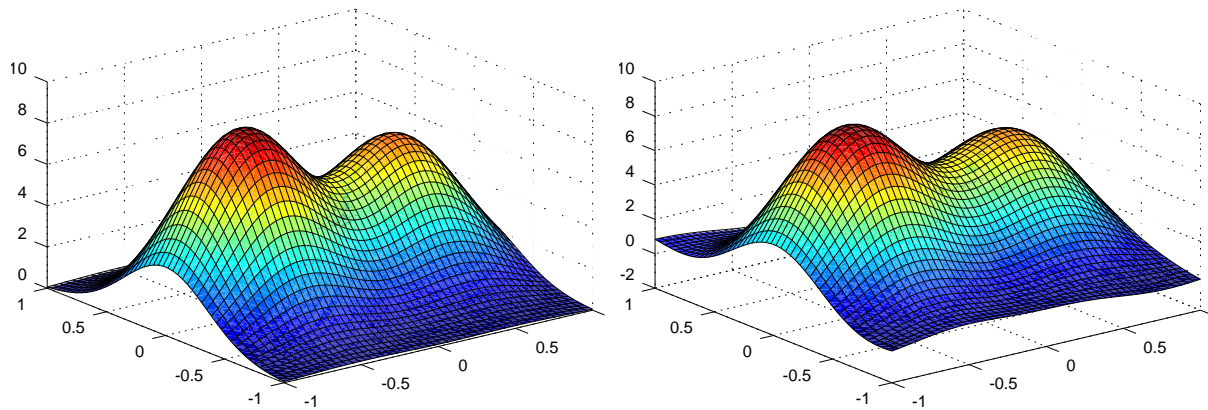


FIGURE 3.17 – Le champ et la régression de l'exemple

pour mesurer à quel point la position des mesures impacte la précision de l'estimation. Grâce à cette carte, il devient possible de parler de trous de couverture de mesure, avec en plus la possibilité de fixer un seuil de tolérance sur la précision de l'estimation via un seuil sur la variance.

3.3 Application à la couverture

Nos travaux présentés dans [ZGC13] ont appliqué l'approche de de Silva et Ghrist [dSG06, dSG07a] mais dans le cadre d'un trou dans la couverture sensorielle pour la mesure en utilisant les processus gaussiens décrits précédemment.

3.3.1 Description de l'application

L'idée est de construire dans un premier temps un complexe simplicial tel que les triangles entre trois capteurs qui appartiennent à ce complexe indiquent que la précision de la régression dans la surface comprise entre ces trois capteurs est suffisamment bonne. Un trou de couverture devient alors un endroit où la précision de l'estimation est insuffisante.

À l'instar de la détection de trous de couverture de [dSG06, dSG07a], une fois le complexe simplicial X construit, le calcul des groupes d'homologie $H_1(X)$ permet de repérer et d'encercler les trous de couverture. Si aucun trou n'est détecté, alors le calcul de $H_2(X, F)$, où F est le bord du domaine, permet de mettre en sommeil des capteurs surnuméraires tout en gardant une couverture respectant le degré de précision souhaité.

Un réseau de capteurs de 50 nœuds est déployé sur un terrain et y mesure un champ scalaire. Ce champ est représenté à gauche sur la figure 3.17.

À partir de ces mesures bruitées par une erreur gaussienne de moyenne nulle, le champ est estimé en utilisant la valeur moyenne obtenue par une régression utilisant un pro-

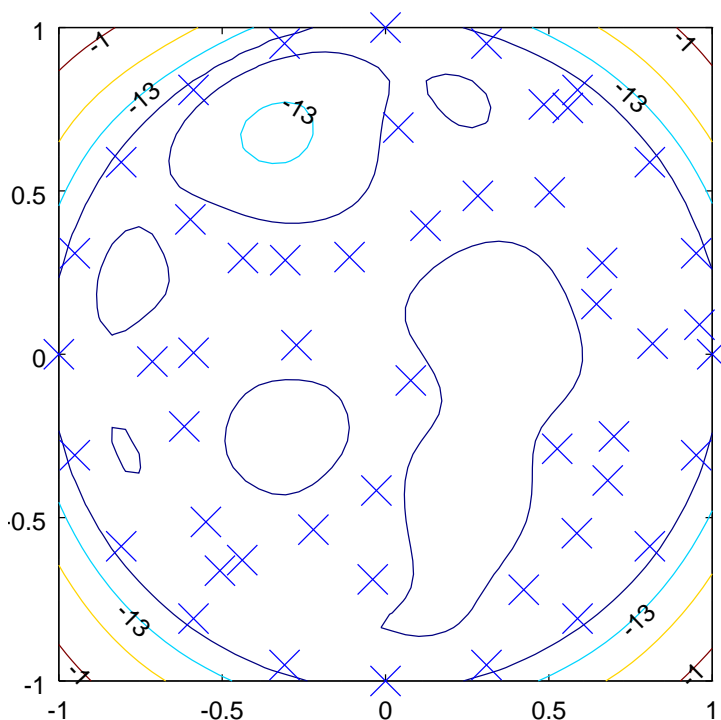


FIGURE 3.18 – Variance de l'estimation et position des capteurs

cessus gaussien de noyau gaussien comme vu précédemment. Naturellement, les hyperparamètres que sont la constante spatiale et l'auto-corrélation, sont obtenus par l'optimisation de (3.27).

Le résultat de la régression est donné à droite de la figure 3.17. Nous remarquons que la forme générale est retrouvée par la régression mais que la précision est parfois insuffisante.

Aussi sur la figure 3.18, nous avons tracé la variance obtenue par la régression précédente selon l'emplacement. En plus des contours, l'emplacement des capteurs est indiqué par des croix.

Nous remarquons que la présence des capteurs influence sur la précision du champ estimé et que celle-ci ne dépend que des corrélations entre les mesures des capteurs et non par leur position absolue comme l'indique (3.26). Cette remarque est à la base de notre méthode.

Comme la localisation des capteurs n'est pas nécessaire, notre méthode s'applique bien au déploiement en intérieur ou si les capteurs n'ont pas de GPS intégré par exemple. De plus, la robustesse de notre solution s'affranchit de la fragilité de la localisation si elle est présente.

Les contours de la figure 3.18 font ressortir six trous de couverture dont trois particulièrement sérieux

- en bas à droite pour des abscisses entre 0 et 0.5, nous observons une faible densité de capteurs et donc la présence d'un trou sur la précision de l'estimation. Ce trou

- de couverture est assez étendu ;
- à côté du point $(-0.3, -0.3)$ se situe un petit trou de couverture de la même sévérité que le précédent mais bien moins étendu ;
- le trou le plus sévère se situe autour du point $(-0.3, 0.7)$. Son étendue est pourtant limitée mais les capteurs sont trop éloignés du centre de ce trou pour espérer le combler.

Notre méthode tente de détecter ce type de trous. La condition est nécessaire mais non suffisante ; c'est-à-dire que si notre méthode détecte un trou alors il existe effectivement mais il se peut que certains trous ne soient pas détectés comme les trois derniers petits trous de la figure 3.18. Pour être plus précis, la méthode décrite ci-dessous se base sur un seuil qui indique le degré de précision souhaitée. Si la précision est grande, plus le nombre de trous détectés sera important.

3.3.2 Construction du complexe simplicial

L'objectif de notre méthode est de reconstruire une version simplifiée de la carte de précision telle que donnée sur la figure 3.18 sans toutefois connaître la position des capteurs.

La version simplifiée sera donnée par un complexe simplicial construit selon les règles suivantes

- tous les nœuds du réseau sont représentés par un sommet ;
- une arête entre deux sommets indique que la variance de l'estimation sur la ligne entre les deux nœuds associés est inférieure à un seuil. La valeur de la variance est calculée en ne prenant en compte que les deux nœuds ;
- un triangle entre trois sommets indique aussi que la variance sur le terrain entre les trois nœuds associés est inférieure au seuil. À l'instar des arêtes, la valeur de la variance est calculée à partir des trois nœuds du triangle seulement ;
- un tétraèdre est ajouté si ses triangles de sa bordure sont déjà dans le complexe.

Il reste à définir comment la variance de l'estimation est calculée. La figure 3.19 sera prise en référence pour les explications.

Considérons deux nœuds x et y . Les corrélations connues et calculables par le réseau de capteurs sont $k(x, x) + \sigma^2$, $k(y, y) + \sigma^2$ et $k(x, y)$. En effet elles correspondent aux auto-corrélations des mesures et à leur inter-corrélation. Par conséquent, l'estimation de ces valeurs ne nécessite pas la connaissance des positions des deux capteurs x et y . Un simple protocole d'échange de mesures permet le calcul distribué de ces corrélations.

Comme nous l'avons vu dans la section 3.2.2, la plupart des noyaux des processus gaussiens sont décroissants avec la distance. Grâce à cette seule hypothèse, les points t qui sont à une distance inférieure à la distance entre x et y vérifient

$$\begin{aligned} k(x, x) + \sigma^2 &\geq k(x, t) \geq k(x, y) \\ k(y, y) + \sigma^2 &\geq k(y, t) \geq k(x, y), \end{aligned} \tag{3.28}$$

où les bornes supérieures viennent de la même hypothèse de décroissance de la corrélation avec la distance.

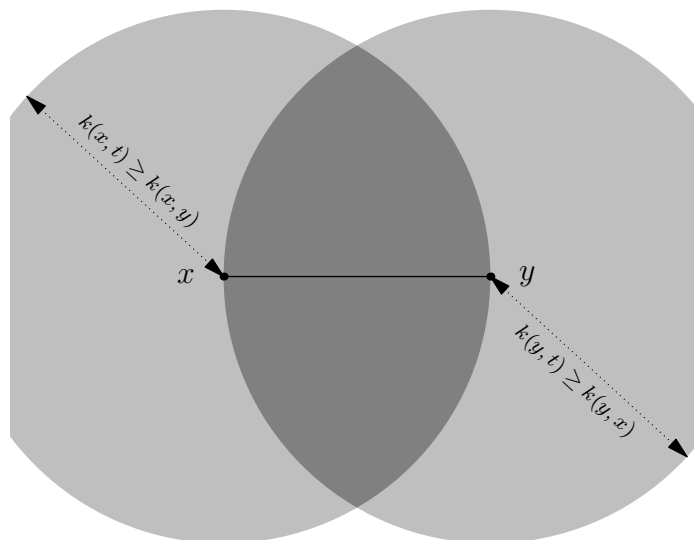


FIGURE 3.19 – Contraintes sur le domaine

Lorsque l'estimation du champ repose uniquement sur les deux points x et y , la variance de l'estimation au point t vaut selon l'équation (3.26)

$$\sigma_{xy}(t) = k(t, t) - \begin{bmatrix} k(x, t) & k(y, t) \end{bmatrix} \begin{bmatrix} k(x, x) + \sigma^2 & k(x, y) \\ k(x, y) & k(y, y) + \sigma^2 \end{bmatrix}^{-1} \begin{bmatrix} k(x, t) \\ k(y, t) \end{bmatrix}. \quad (3.29)$$

Naturellement, plus le nombre de mesures prises en compte est important plus l'estimation est fiable en un point donné. Cependant les mesures distantes ont un impact moindre. Par conséquent $\sigma_{xy}(t)$ est une borne supérieure sur la variance de l'estimation au point t . La borne sera d'autant meilleure que les capteurs x et y sont proches de t .

Les points t vérifiant les deux contraintes (3.28) sont fortement grisés sur la figure 3.19. Naturellement le segment $[x, y]$ est inclus dans ce domaine. Par conséquent, en maximisant $\sigma_{xy}(t)$ sur le domaine décrit par (3.28), il est possible d'obtenir la borne supérieure $\bar{\sigma}_{xy}$ sur la précision de l'estimation

$$\bar{\sigma}_{xy} = \max_{\mathbf{u}} \left(k(x, x) - \mathbf{u}^T \begin{bmatrix} k(x, x) + \sigma^2 & k(x, y) \\ k(x, y) & k(y, y) + \sigma^2 \end{bmatrix}^{-1} \mathbf{u} \right) \quad (3.30)$$

sous les contraintes $\begin{bmatrix} k(x, x) + \sigma^2 \\ k(y, y) + \sigma^2 \end{bmatrix} \geq \mathbf{u} \geq \begin{bmatrix} k(x, y) \\ k(y, x) \end{bmatrix}$,

où \mathbf{u} joue le rôle muet de $\begin{bmatrix} k(x, t) & k(y, t) \end{bmatrix}^T$ de l'équation (3.29).

Il est possible de définir $\bar{\sigma}_{xyz}$ pour des triplets de capteurs et autres k -uplets de la même façon avec la même intuition sous-jacente. Bien sûr, la borne $\bar{\sigma}_{xyz}$ sur le triangle xyz sera plus précise que les bornes $\bar{\sigma}_{xy}$, $\bar{\sigma}_{xz}$ et $\bar{\sigma}_{yz}$ de ses côtés.

Toutes les optimisations précédentes sont facilement calculables numériquement car elles sont de la forme quadratique ; la fonction objectif est convexe ce qui assure l'unicité de

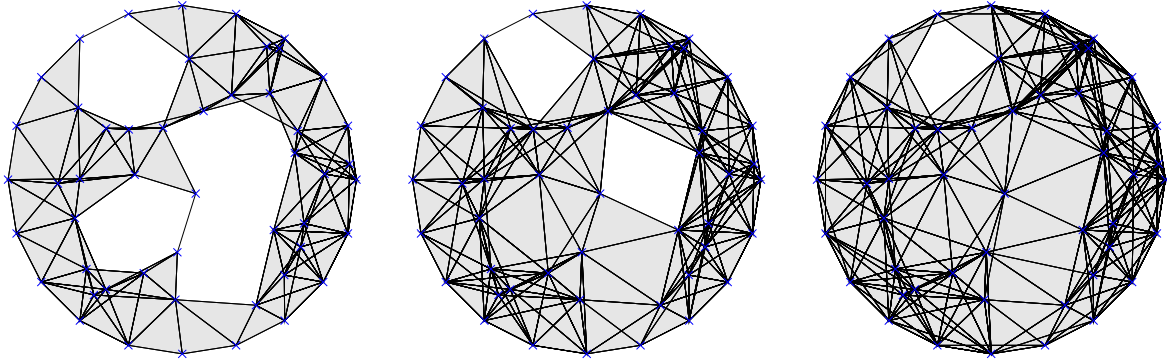


FIGURE 3.20 – Complexe simplicial pour les seuils 7.0, 7.5 et 7.6

la solution. La méthode du simplexe pour la programmation linéaire peut être détournée pour assurer la programmation quadratique qui nous concerne [Wol59]. Comme la dimension du problème est faible, le coût calculatoire reste très limité.

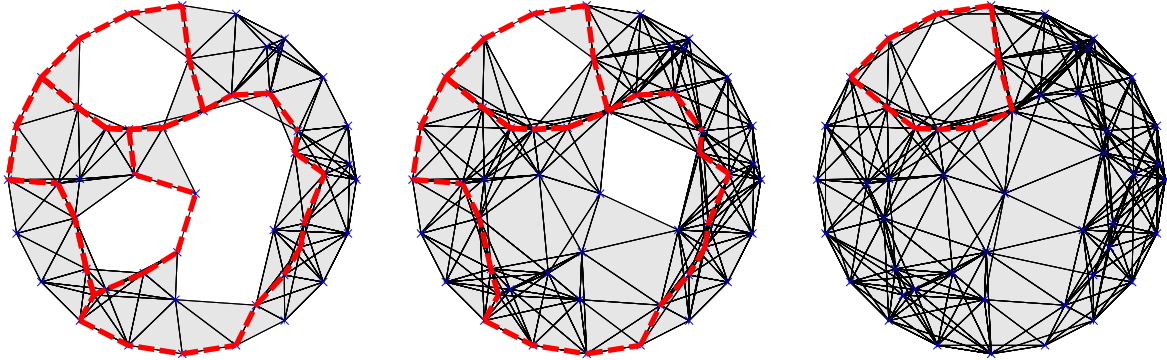
Le complexe simplicial découlant des observations précédentes se construit en utilisant comme sommets les capteurs, puis les arêtes sont formées des couples de capteurs (x, y) tel que $\bar{\sigma}_{xy}$ soit inférieure à un seuil prédéterminé. Les triangles xyz qui satisfont la même condition sur $\bar{\sigma}_{xyz}$ sont aussi inclus dans le complexe tout comme les tétraèdres du même type.

La figure 3.20 illustre le résultat sur l'exemple d'application précédent. Pour les besoins de la simulation, les hyper-paramètres du modèle ont été optimisés comme expliqué auparavant et les corrélations sont calculées en utilisant le noyaux gaussien correspondant. Toutefois, l'utilisation des corrélations estimées par des méthodes classiques serait naturellement préférée en situations réelles.

Les différents complexes sont construits en utilisant un seuil sur la variance (en dB) fixé de gauche à droite à 7 dB, 7.5 dB et 7.6 dB. Naturellement, plus le seuil est élevé moins les trous sont présents. Mais les trois trous principaux décrits ci-dessus sont bien présents et leur ordre de disparition dans le complexe reflète parfaitement les commentaires sur la sévérité des différents trous de couverture.

Le complexe ainsi construit assure la qualité de l'estimation en limitant la variance de l'estimation. Comme la construction fait appel à des optimisations par paire ou par triplet, l'écart entre la seuil est la variance véritable est parfois grand mais le complexe ainsi construit est toujours pessimiste. En effet, si un trou détecté indique que la couverture est insuffisante pour une certaine qualité de l'estimation, cette qualité est sûrement meilleure que celle prévu par notre méthode.

Il faut remarquer que la construction du complexe simplicial ne dépend pas directement de la localisation des capteurs mais que seules les corrélations paires à paires entre les mesures des capteurs sont nécessaires. Par conséquent, aucune méthode de placement n'est nécessaire, seul le lien entre le noyau de la régression gaussienne et la distance doit

FIGURE 3.21 – Cycles générateurs de H_1 pour les seuils 7.0, 7.5 et 7.6

être respecté.

3.3.3 Détection des trous de couverture

Une fois le complexe simplicial construit, il reste à détecter les trous de couverture, c'est-à-dire les trous dans le complexe. Cette détection est immédiatement obtenue par le calcul du groupe H_1 dont la dimension, le nombre de Betti β_1 indique le nombre de trous.

Les générateurs des trois complexes illustratifs de la figure 3.20 sont dessinés sur la figure 3.21 en traits tiretés gras rouges. Ils entourent effectivement les trous même s'ils ne sont pas optimaux sur leur longueur. Comme attendu, les nombres de Betti β_1 des trois complexes sont bien respectivement 3, 2 et 1.

3.3.4 Utilisation de la persistance

Les exemples d'application montrent l'intérêt de la méthode et son efficacité à retrouver les trous de couverture. Toutefois, il est difficile de fixer le niveau du seuil *a priori* car, comme expliqué plus haut, il y a un décalage entre ce seuil et la véritable précision.

Ce type de difficulté est classique dans les méthodes d'analyse de données basées sur une certaine géométrie où un paramètre d'échelle est souvent nécessaire pour connaître le degré de précision à utiliser. L'analyse topologique des données [Car09, Car14] fournit une solution très élégante grâce à l'utilisation de la persistance homologique.

L'idée est de construire le complexe comme précédemment tout en étiquetant les arêtes entre les capteurs x et y par $\bar{\sigma}_{xy}$, les triangles reposant sur les nœuds x , y et z par $\bar{\sigma}_{xyz}$ et ainsi de suite.

Les complexes précédents sont donc des sous-complexes des simplexes étiquetés par une valeur inférieure au seuil. Par conséquent, en faisant varier le seuil, nous pouvons construire

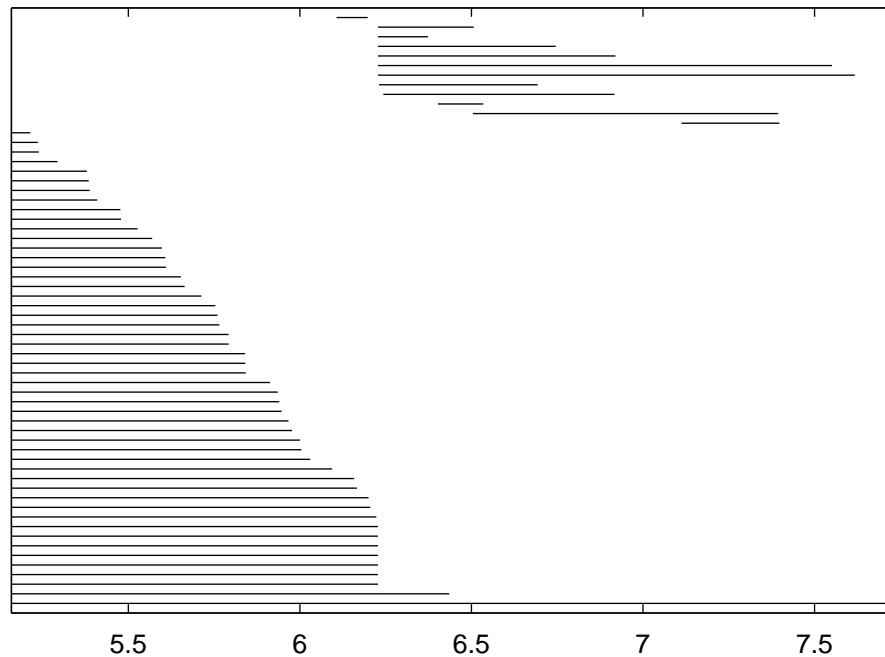


FIGURE 3.22 – Code-barres des groupes d’homologie H_0 en bas et H_1 en haut

une filtration sur le complexe simplicial complet. Cette filtration reste pertinente pour notre application [dSG07a].

Le complexe complet et la filtration permet de calculer la persistance des groupes d’homologie. Nous avons représenté cette persistance des groupes H_0 et H_1 par des code-barres [Ghr08a] sur la figure 3.22. Le groupe H_0 qui mesure le nombre de composante connexe est visible en bas de la figure. Nous remarquons que les groupes de capteurs se forment rapidement pour n’en former plus qu’un lorsque le seuil dépasse 6.4 dB.

La persistance du groupe H_1 , visible en haut de la figure, est plus intéressant car un trait représente un trou entre le seuil où il apparaît et le seuil où il disparaît. Plus le trait est long, plus le trou est significatif et plus le trait disparaît tard plus le trou de couverture est sévère.

Les trois seuils qui ont permis la construction des complexes de la figure 3.20, 7 dB, 7.5 dB et 7.6 dB, font bien intervenir 3, 2 et 1 trous sur le code-barre.

Grâce au code-barre, il devient plus aisé de sélectionner les seuils intéressants et surtout le type et la sévérité des trous de couverture sans avoir à fixer de seuil *a priori* : la persistance apporte une grande flexibilité à l’étude de la couverture de mesure.

De plus, les algorithmes de calcul de persistance [Zom09, EH10] permettent de récupérer les différents générateurs des groupes H_1 sans difficulté. Il n’est donc pas nécessaire de faire une première étude sur les seuils puis de refaire des calculs un fois les seuils fixés pour connaître plus précisément les groupes H_1 d’intérêt.

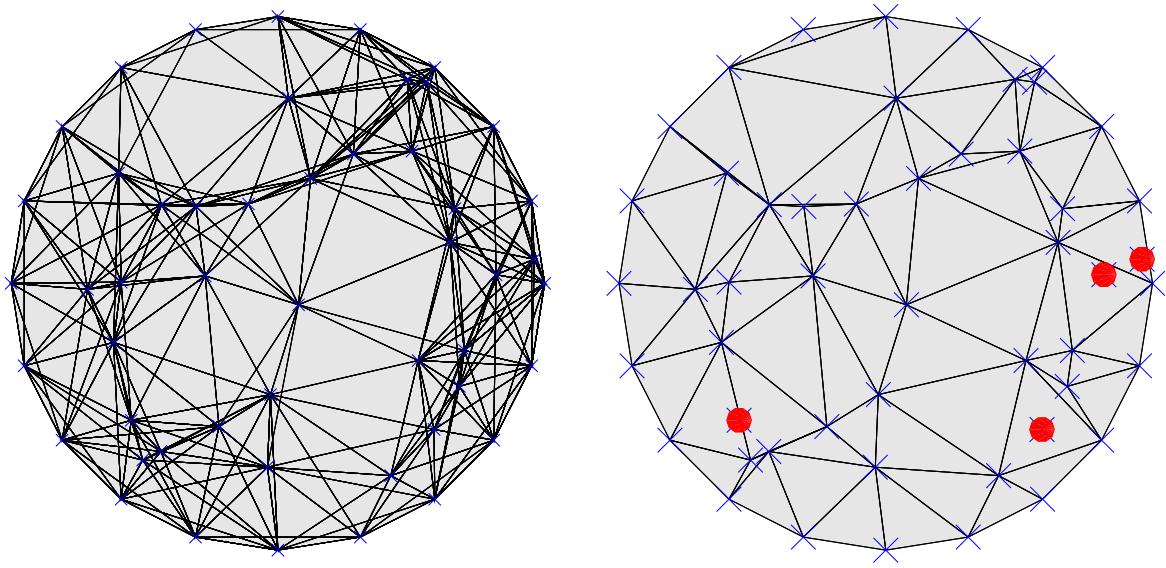


FIGURE 3.23 – Mise en veille de capteurs

3.3.5 Mise en veille de capteurs

Les articles de de Silva et Ghrist [dSG06, dSG07b] offrent une alternative à la détection des trous qui repose sur le calcul du groupe H_2 relativement à la frontière du domaine. Si ce groupe est vide alors il y a au moins un trou qui est récupérable par le calcul de H_1 comme auparavant. Par contre si le groupe H_2 relativement à la frontière est non vide alors le nombre de Betti associé est nécessairement 1, c'est-à-dire qu'il n'y a qu'un générateur. Naturellement, l'unicité du générateur est à comprendre au sens du groupe quotient : les représentants de la classe du générateur peuvent être beaucoup plus nombreux.

Il est possible de récupérer les sommets impliqués dans les simplexes d'un représentant de l'unique générateur du groupe H_2 , s'il existe. Ces sommets sont alors suffisants pour assurer la couverture, les autres capteurs peuvent être mis en veille sans créer de trous de mesure au niveau de précision fixé.

La figure 3.23 représente à gauche le complexe de l'exemple lorsque le seuil est suffisamment haut pour fermer le dernier trou, le plus sévère. En calculant le groupe H_2 relatif à la frontière qui est formée des capteurs placés sur le cercle unité et des arêtes les plus courtes entre ces capteurs, nous avons extraits un ensemble de nœuds pouvant être mis en veille. Ces capteurs sont indiqués par des points rouge sur la figure de droite. Le sous-complexe ne faisant pas intervenir ces nœuds en veille est aussi représenté sur la figure de droite pour indiquer qu'aucun trou de couverture n'est apparu suite à ces mises en veille.

Naturellement, le nombre de nœuds mis en veille n'est pas optimal car aucune recherche de représentant optimal du générateur de H_2 n'a été effectuée. Seul le principe est réellement testé ici. Toutefois, le nombre de capteurs pouvant être mis en veille sur la figure 3.23 ne semble pas être très éloignés de l'optimum.

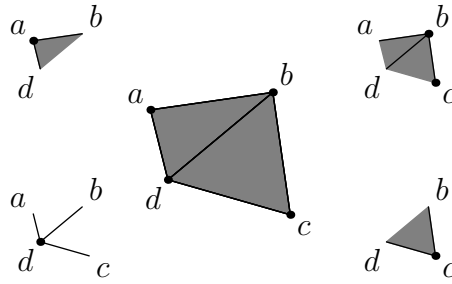


FIGURE 3.24 – Structure de donnée distribuée pour représenter un complexe

Le groupe $H_2(X, F)$ relativement à la frontière F peut se calculer en ajoutant au complexe une cellule temporaire σ de dimension 2 dont le bord $\partial\sigma$ est la frontière F . Les générateurs du groupe d'homologie non relatif $H_2(X \cup \sigma)$ qui sont de la forme $g = \sigma + g'$ fournissent directement les générateurs g' du groupe relatif $H_2(X, F)$.

3.3.6 Remarques sur l'algorithme

Notre méthode de détection des trous amènent quelques remarques car nous avons choisi de la présenter à travers un exemple et en explicitant uniquement les concepts topologiques associés. Toutefois, pour une mise en œuvre plus pratique dans le cadre des réseaux de capteurs, faut il l'adapter.

La distribution de l'algorithme est discutée ci-dessous pour montrer à quel point notre méthode satisfait les contraintes des RCSF, à savoir la localisation des données et des calculs pour permettre le passage à l'échelle. Enfin, nous aborderons rapidement la localisation des trous détectés en présentant l'optimisation des cycles puis nous envisagerons des améliorations de notre méthode basées sur une représentation plus fidèle de la précision de la régression par le complexe simplicial.

3.3.6.1 Distribution de l'algorithme

L'algorithme présenté ici ne semble pas distribué par la présentation très haut niveau qu'il en a été faite. Toutefois, la construction du complexe ne fait intervenir que des calculs locaux. En effet pour calculer (3.30), seuls les capteurs mis en jeu sont requis avec leur corrélation. Comme les faibles corrélations entre les mesures correspondent aux capteurs suffisamment proches, la construction du complexe est réalisable localement.

De même en utilisant l'algorithme distribué du calcul de l'homologie de l'article [DGJ12], l'ensemble de notre méthode peut être distribué simplement. Cet article présente aussi une structure de données distribuées [GBH00] pour représenter le complexe simplicial et un protocole pour le gérer.

Cette structure de données distribuées est illustrée sur la figure 3.24. Le complexe au

centre est distribué sur les nœuds a , b , c et d . Sur le pourtour de la figure centrale, nous avons représenté la vision du complexe par chacun des nœuds. Par exemple a gère a , ab , ad et abd comme illustré en haut à gauche.

La gestion des simplexes est distribuée comme suit : chaque simplexe est géré par les deux nœuds de sommets de plus faible identifiant. Par exemple le simplexe abd est géré par les nœuds a et b mais est inconnu du nœud c .

Nous remarquons la forte localisation de la structure de donnée ; les nœuds gérant un même simplexe sont nécessairement fortement reliés car ils sont voisins dans le complexe.

L'ajout et la suppression d'un simplexe dans la structure sont simplifiés car ces opérations se font au travers d'échanges entre seulement deux nœuds, ceux de plus faible identifiant. La cohérence de la représentation distribuée lors de ces échanges de messages peut se faire par le protocole Two-Phase Commit (2PC) [GBH00].

Si, malheureusement, les sommets du complexes ne sont pas des capteurs, il reste cependant possible de distribuer la structure de données. En effet, des techniques de distribution de structures de données ont été développés dans le domaine de l'informatique distribuée, voir par exemple [GBH00], ou [LNS96, ME01].

Il existe une seconde approche [MJ07b, MJ07a, CK13] pour le calcul distribué des générateurs de H_1 qui repose sur la propriété des fonctions harmoniques qui vérifient $\Delta_1 f = 0$ avec Δ_1 l'opérateur de Laplace-Beltrami associé au complexe. En effet, la version discrète de la décomposition de Hodge que permet le calcul différentiel discret [Hir03, DHL05, CdGD13] indique que H_1 est isomorphe à $\ker \Delta_1$. Le calcul distribué des générateurs de H_1 peut alors se faire par la résolution de l'équation différentielle

$$\frac{dy}{dt} = -\Delta_1 y. \quad (3.31)$$

Cette dernière pouvant être approché par un schéma d'Euler distribué. Nous n'avons pas choisi cette approche car le corps sur lequel H_1 doit être défini pour pouvoir l'utiliser est \mathbb{R} et alors les problèmes de convergence et d'initialisation seront à gérer.

3.3.6.2 Optimisation des cycles

Comme les groupes d'homologie sont des groupes quotients, les générateurs calculés et affichés sur la figure 3.21 ne sont bien définis qu'à un bord près. C'est à dire que tous les cycles entourant les trous de couverture équivalent à un générateur \mathbf{c} sont de la forme $\mathbf{x} = \mathbf{c} + \partial\mathbf{y}$.

Il est parfois intéressant de trouver le cycle équivalent à un générateur \mathbf{c} qui minimise sa taille. Cela permet par exemple de situer un trou de couverture en connaissant les nœuds qui sont sur son bord. Cette recherche s'écrit sous la forme

$$\begin{aligned} & \min_x \text{taille}(\mathbf{x}) \\ & \text{tel que } \mathbf{x} = \mathbf{c} + \partial\mathbf{y}. \end{aligned} \quad (3.32)$$

La taille d'un cycle α n'est pas nécessairement bien défini car cela peut être le nombre de simplexes qui forment sa base ou encore la somme de poids de ces simplexes définis selon l'application. Ce genre d'approche est étudié dans [DHK11] et [Eri11] qui propose de transformer ces problèmes d'optimisation en programmation linéaire.

Ces problèmes d'optimisation de cycles pour l'application à la couverture sont aussi abordés dans [TSJ10]. Leur approche n'est pas combinatoire comme ici car les auteurs utilisent l'isomorphisme entre le noyau de l'opérateur de Laplace-Beltrami Δ_1 et le groupe $H_1(X)$ par la théorie de Hodge discret [ME06, Hir03, DHL05]. Ainsi, il devient possible de distribuer le calcul de générateur du groupe d'homologie et d'en optimiser la longueur par des méthodes de descente de gradient.

L'article [DSW10] poursuit le même objectif mais en considérant une base complète du groupe d'homologie et la notion de taille découlent directement de la notion de distance sous-jacente à l'espace pour lequel le complexe simplicial est une approximation.

Enfin l'article [CF10] définit la notion de taille par une approche très générale et typiquement topologique. Aucune notion géométrique n'y est requise. L'idée est d'avoir un ensemble de sous-complexes qui servent d'étalons pour la mesure de taille. Un algorithme permettant le calcul d'une base minimale est alors fourni.

Tout ces algorithmes sont intéressants mais dépendent fortement de l'application. De plus la mise sous forme distribuée semble très complexe car ils nécessitent une connaissance globale des groupes Z_1 et B_2 .

3.4 Conclusion

La modélisation du champ par un processus gaussien nous a permis de construire dans ce chapitre un algorithme de détection de trou de couverture sans connaître la position des capteurs grâce notamment à une application des groupes d'homologie. De plus, la persistance homologique fournit un outil flexible qui limite l'introduction d'*a priori* sur le champ à mesurer en fixant *a posteriori* le seuil de détection.

Enfin, pouvoir en plus sélectionner un ensemble suffisant de capteurs actifs constitue un à-côté particulièrement intéressant de la méthode qui assure une gestion intelligente du réseau pour augmenter sa durée de vie.

Toutefois, nos solutions reposent sur un modèle pour lequel des hyper-paramètres doivent être déterminés par exemple en maximisant leur vraisemblance étant données les mesures des capteurs. De part ces aspects calculatoire, il est difficile de rendre simplement notre algorithme complètement distribuée tout en le modifiant pour qu'il s'adapte aux variations du réseau ou des mesures.

C'est pourquoi, nous présenterons notre algorithme de calcul des groupes d'homologie dans le prochain chapitre qui s'adapte aux variations des complexes représentant le réseau. Cet algorithme corrige les défauts précédents et offre, grâce à sa grande généralité, de nouvelles

applications notamment la gestion adaptative des capteurs actifs pour une gestion plus fine de la consommation d'énergie.

Chapitre 4

Théorie de Morse et RCSF

4.1	La théorie de Morse et sa version discrète	68
4.1.1	Version différentielle de la théorie de Morse	68
4.1.2	Historique et applications de la théorie de Morse discrète	71
4.1.3	Champ discret de vecteurs combinatoire	71
4.1.4	Le flot discret d'un champ de vecteur	73
4.1.5	Construction du complexe de Morse	74
4.1.6	Exemple de l'anneau	75
4.1.7	La fonction potentielle	76
4.1.8	Simplification du complexe de Morse	77
4.1.9	Construction aléatoire du champ discret de vecteur	80
4.2	Algorithme proposé pour le suivi des groupes d'homologie	81
4.2.1	Principe de l'algorithme	81
4.2.2	Mise à jour incrémentale	82
4.2.3	Comparaison avec d'autres algorithmes	87
4.3	Application à la couverture de mesure	88
4.3.1	Description de l'application	88
4.3.2	Résultats des simulations	89
4.4	Application à la gestion de l'énergie	92
4.4.1	Description de l'application	92
4.4.2	Cadre général des simulations	93
4.4.3	Sélection des capteurs éveillés	93
4.4.4	Suivi de la couverture et activation automatique des capteurs	102
4.4.5	Améliorations de l'algorithme	105
4.4.6	Gestion de la couverture sans garantie	107
4.5	Conclusion	107

La méthode du chapitre précédent que nous avons développée pour l'application de détection de trous de couverture suppose que les covariances entre les mesures des différents

nœuds soient connues et surtout qu'elles ne varient pas. Par conséquent, une fois le complexe simplicial approchant la couverture du champ il ne doit pas être modifié par ajout ou suppression de simplexes.

En situations réelles, les covariances seront estimées au fil de l'eau parce que les nœuds seront mobiles ou encore parce que la durée de vie des nœuds est parfois limitée. Ainsi, certains d'entre eux disparaîtront alors que d'autres pourront être redéployés. Dans tout ces cas, le complexe simplicial sous-jacent devra s'adapter aux modifications structurelles.

Répondre à ce problème est donc primordial. Une première solution sera de recalculer le complexe simplicial et ses groupes d'homologie périodiquement mais cette solution n'est pas satisfaisante car le coût calculatoire est trop élevé. Nous proposons dans ce chapitre d'utiliser la théorie de Morse pour suivre simplement les variations du complexe simplicial et pour éviter la plupart des calculs pour le suivi des groupes d'homologie.

La section suivante présente la théorie de Morse principalement dans sa version discrète. À partir de cette dernière nous proposerons à la section 4.2 son application au problème de poursuite de l'évolution d'un complexe simplicial. Cette capacité sera mise à profit pour le problème de couverture dans la section 4.3, la gestion de l'énergie sera également étudiée à la section 4.4. Enfin, la section 4.5 conclura ce chapitre.

4.1 La théorie de Morse et sa version discrète

4.1.1 Version différentielle de la théorie de Morse

Les groupes d'homologie ne sont pas modifiés si l'espace de base subit des déformations de rétractions. Plus précisément, un espace X et une rétraction A de X ont des groupes d'homologie isomorphes.

Une rétraction A est ici un sous espace de X obtenu par une fonction continue $F : X \times [0, 1] \rightarrow X$ telle que $F(0, \cdot)$ soit l'identité sur X et $F(1, x) \in A$. Comme F est continue, aucun « trous » n'apparaît ni ne disparaît pendant la déformation.

Si la déformation est judicieusement choisie, le calcul des groupes d'homologie de A sera plus simple que celui de l'espace initial X . La méthode la plus simple pour décrire cette déformation consiste à utiliser un champ de vecteur sur la surface X . La déformation suit le flot correspondant. Par exemple, l'anneau de la figure 4.1 est homologiquement équivalent à un cercle. En effet, sous l'effet du champ de vecteurs, l'épaisseur de l'anneau diminue jusqu'à devenir nul. La figure 4.1 montre la déformation à deux instants différents : proche du démarrage du flot et vers la fin.

La théorie de Morse développe cette idée en passant par un champ de vecteurs créé par une fonction potentielle f suffisamment régulière ; le champ est donnée par l'opposé du gradient de f . À chaque point critique p de la fonction potentielle est associé une cellule σ_p dont la dimension est donnée par son indice qui est la dimension de la variété instable

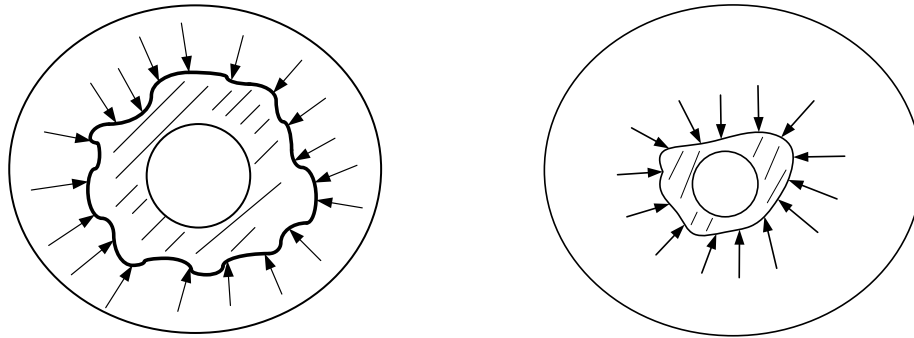


FIGURE 4.1 – Déformation de l'anneau

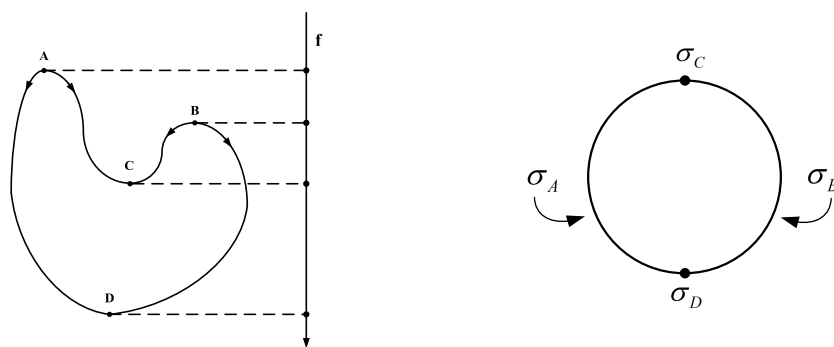


FIGURE 4.2 – Fonction de Morse et son complexe unidimensionnel

de p et qui est aussi donnée par le nombre de valeurs propres négatives du Hessien de f au point p . La référence sur la théorie de Morse est le livre de Milnor [Mil63], voir Bott [Bot88] pour un article plus historique ou [For04] pour une introduction basée sur les points d'équilibre de systèmes dynamiques.

Un complexe de Morse est alors associé à f qui comprend l'ensemble des cellules σ_p associés aux points critiques de f . L'opérateur de bord est donné en suivant le flot du gradient.

La figure 4.2 montre l'utilisation de la fonction de Morse qui correspond à l'altitude pour construire un complexe cellulaire équivalent à la courbe de gauche qui est topologiquement équivalente à un cercle. La fonction f a quatre points critiques, A à D . Comme A et B sont des maxima, leur indice vaut 1 alors que C et D sont des minima donc d'indice 0.

À chaque point critique est associé une cellule de dimension donnée par l'indice du point. À A et B sont associés les arêtes σ_A et σ_B à droite de la figure 4.2 et à C et D sont associés les sommets σ_C et σ_D . En suivant l'opposé du gradient de f , nous voyons que le bord de la variété instable de A associé à l'altitude est d'un côté C et de l'autre D . Il en est de même pour le bord de la variété instable de B . Par conséquent, en choisissant une orientation, nous trouvons que $\partial\sigma_A = \sigma_C - \sigma_D$ et $\partial\sigma_B = \sigma_D - \sigma_C$. Le complexe de Morse associé, dessiné à droite de la figure 4.2 est donc complètement défini par l'espace initial et la fonction potentielle.

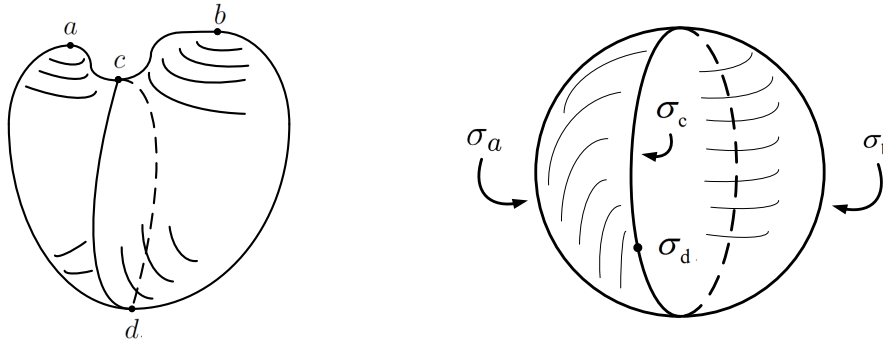


FIGURE 4.3 – Fonction de Morse et son complexe bidimensionnel

Pour un second exemple de dimension 2, la fonction potentielle qui donne l'altitude de la surface en forme de cœur représentée sur la figure 4.3 a quatre points critiques, a , b , c et d . Comme a et b sont des maxima, la dimension des cellules σ_a et σ_b est 2. Le point c est un point « col, » donc, une seule direction tend vers le bas et la cellule σ_c est de dimension 1. Enfin le point d est un minimum et la cellule σ_d est un point.

Descendre le point c selon le gradient de l'altitude — selon la gravité — se termine nécessairement au point d quelque soit la direction descendante pris à partir de c . Par conséquent le bord de σ_c est $2\sigma_d$. De même le bord de σ_a est σ_c car le flot partant de a rejoint asymptotiquement le flot partant de c . Le même raisonnement tient pour b .

Au final, le complexe de Morse associé à la fonction altitude pour la forme de cœur est composé de quatre cellules, σ_a à σ_d avec $\partial\sigma_a = \sigma_c$, $\partial\sigma_b = \sigma_c$ et $\partial\sigma_c = 2\sigma_d = 0$ si nous travaillons sur le corps binaire pour éviter les difficultés liées à l'orientation des cellules.

Il y a un lien fort entre l'espace topologique X et le complexe de Morse M_f associé à la fonction potentielle f . Notons β_k les nombres de Betti de X et m_p le nombre de points critiques de f d'indice p . Si le champ de vecteurs satisfait une condition précisée par la suite, les deux séries de nombres β_k et m_p sont reliés par la relation de Morse

$$m_k - m_{k-1} + \cdots \pm m_0 \geq \beta_k - \beta_{k-1} + \cdots \pm \beta_0 \quad \text{pour tout } k \geq 0. \quad (4.1)$$

De cette forme forte d'inégalités, nous pouvons déduire la forme faible déjà intéressante,

$$m_k \geq \beta_k \quad \text{pour tout } k \geq 0. \quad (4.2)$$

Ainsi un simple comptage des points critiques de la fonction potentielle permet d'approcher les nombres de Betti de la surface. La relation (4.1) devient une égalité si k vaut la dimension de X , nous retrouvons en effet la caractéristique d'Euler-Poincaré

$$\chi(X) = \beta_{\dim X} - \beta_{\dim X-1} + \cdots \pm \beta_0 = m_{\dim X} - m_{\dim X-1} + \cdots \pm m_0 \quad (4.3)$$

En résumé la théorie de Morse associe à un espace et une fonction potentielle un complexe, le complexe de Morse, qui a la même homologie que l'espace initial. Souvent la

décomposition de l'espace initial en son complexe de Morse rend le calcul plus simple et la représentation plus compacte. De même la notion de filtration de la persistance homologique est proche et donc à relier à la notion de fonction potentielle de la théorie de Morse.

Ces avantages rendent la théorie de Morse très attractive pour bon nombre d'applications, par exemple pour l'analyse de donnée [AAC11]. De même la fonction potentielle décompose les espaces ce qui est l'approche utilisée dans [AC07] pour l'analyse de forme ou [EH09] pour la segmentation d'images médicales. Enfin [GBH08] utilise l'approximation d'un espace par le complexe de Morse pour faciliter le calcul des groupes d'homologie.

4.1.2 Historique et applications de la théorie de Morse discrète

Une version combinatoire de la théorie de Morse n'est malheureusement pas immédiate. Forman qui l'a développée autour de l'année 2000 a dû redéfinir plusieurs concepts de la géométrie différentielle pour y parvenir. La première difficulté est de transposer [For99, For98a] les concepts de topologie différentielle — surtout la notion de champ de vecteur — de la théorie de Morse aux espaces combinatoires comme les complexes cellulaires.

La référence du travail de Forman est [For98b] mais il a écrit en 2002, un article plus accessible [For02b]. Ce travail est ensuite étendu à la cohomologie et au transport des opérations cohomologiques aux espaces combinatoires dans [For02a]. Kozlov reprend la théorie de Forman [Koz08] pour l'adapter à son approche des espaces combinatoires.

Grâce à la puissance de la théorie de Morse et à la possibilité de calcul par ordinateur donné par l'approche combinatoire, le champ d'applications basées sur la théorie de Morse discrète devient de plus en plus vaste. Parmi ces applications, nous trouvons par exemple

- la simplification topologique pour supprimer les caractéristiques topologiques considérées comme insignifiantes [Bau11, BLW12];
- l'aide à la visualisation et à la compression des espaces triangulés [LLT04];
- faciliter le calcul des groupes d'homologie par réduction [HMM14, MN13] ou pour le calcul des groupes de cohomologie des faisceaux [CGN13] ou encore de la persistance homologique [AKL13];
- pour approcher et classer rapidement les espaces grâce à la forme faible des inégalités de Morse (4.2) par une approche de Monte-Carlo [BL14];
- pour le traitement d'image [RWS11] ou l'analyse d'un nuage de points [KKM05];
- et bien d'autres utilisations.

4.1.3 Champ discret de vecteurs combinatoire

L'idée de base de la version discrète de la théorie de Morse est de construire l'équivalent d'un champ de vecteur qui indique comment réduire l'espace initial ou encore comment associé la cellule d'un point critique à son bord. Le point fondamental de Forman est que seule la direction des vecteurs est important et non leur taille. Ainsi, il n'est pas

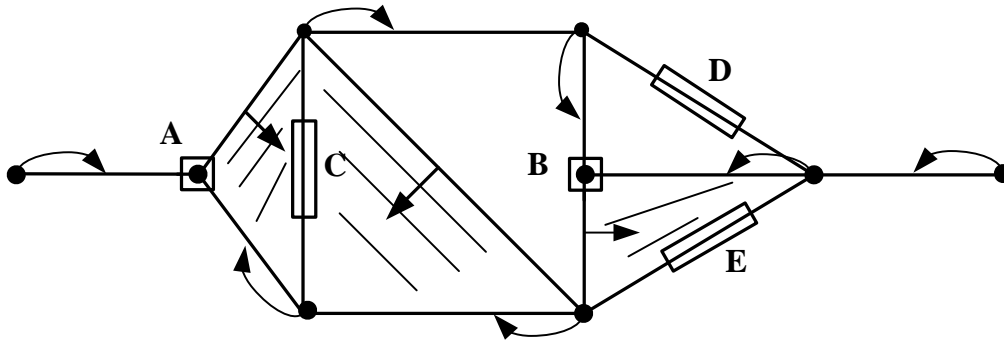


FIGURE 4.4 – Champ de vecteur discret pour un complexe simplicial

nécessaire de construire une géométrie différentielle discrète complète comme dans [Hir03] ou [DHL05] mais rester au niveau combinatoire.

Un champ discret de vecteurs — Discrete Vector Field en anglais — défini sur un espace cellulaire X est un ensemble de vecteurs qui est une paire de cellules (σ, τ) telles que σ soit sur le bord de τ . De plus, une cellule ne doit être associée qu'à une seule autre cellule au maximum.

Par conséquent, un champ discret de vecteurs est un simple appariement ou un couplage entre cellules incidentes ; l'aspect différentielle du champ de vecteur disparaît complètement. Les cellules que ne sont associées avec aucune autres sont qualifiées de critique. Elles joueront le même rôle que les points critiques de la théorie de Morse différentielle.

Il existe toujours un champ discret de vecteur ; par exemple, toutes les cellules peuvent être considérées comme critique.

La figure 4.4 représente un complexe simplicial et un champ discret de vecteur. Chaque vecteur (σ, τ) est représenté par une flèche allant de σ vers τ . Ainsi les flèches vont des sommets vers les arêtes ou des arêtes vers des triangles. Nous avons aussi encadré les cellules critiques, non associées par le champ, les deux sommets A et B et les trois arêtes C , D et E .

La version discrète des inégalités de Morse permet déjà d'obtenir une bonne approximation des invariants homologiques. En effet, nous avons deux sommets critiques et trois arêtes critiques, par conséquent,

$$\beta_0 \leq 2, \quad \beta_1 \leq 3, \quad \beta_1 - \beta_0 \leq 3 - 2 = 1, \quad \text{et } \chi(X) = 3 - 2 = 1. \quad (4.4)$$

Ces inégalités sont fortes car elles approchent fortement les valeurs théoriques $\beta_0 = 1$ et $\beta_1 = 2$. Dans un soucis de simplification, il est par conséquent préférable d'avoir le maximum de cellules appariées par le champs de vecteur car alors les inégalités précédentes seront encore plus strictes.

Cette remarque est à la base du système de classification présentée dans [BL14] qui construit aléatoirement un champ de vecteur discret par un algorithme simple et rapide

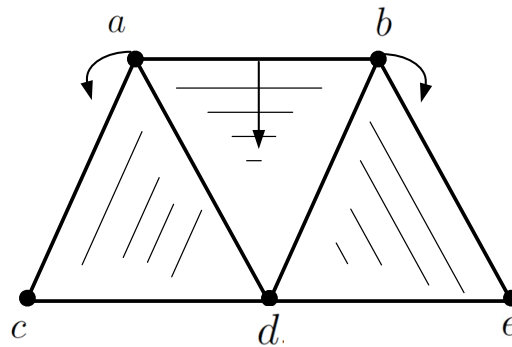


FIGURE 4.5 – Flot d'un champ discret de vecteurs

de type glouton qui fournit alors des bornes sur les nombres de Betti. Ces bornes sont très souvent fines et valent les nombres de Betti. Pour affiner encore plus ces bornes il est possible de simplifier encore le complexe de Morse comme nous le verrons ultérieurement.

4.1.4 Le flot discret d'un champ de vecteur

La donnée d'un champ discret de vecteur permet de construire un flot qui permettra de construire le complexe de Morse à l'instar de la version différentielle qui utilise les variétés instables des points critiques.

Dans un premier temps, un champ de vecteur permet de construire une application linéaire V sur le complexe de chaînes C_* . Soit une cellule σ associée à une cellule τ et sur la face de cette dernière. Le champ linéarisé $V(\sigma) = \pm\tau$. Le signe est choisi tel que le signe de σ dans $\partial V(\sigma)$ soit -1 . Naturellement, travailler sur le corps binaire évite de considérer ces problèmes d'orientation.

Le flot Φ est un opérateur linéaire défini à partir de V et de ∂ par $\Phi = I + V\partial + \partial V$ avec I l'identité, c'est-à-dire,

$$\Phi(\sigma) = \sigma + V(\partial\sigma) + \partial V(\sigma). \quad (4.5)$$

Cet opérateur linéaire $\Phi : C_k \rightarrow C_k$ est bien défini et les chaînes images sont de la même dimension que les chaînes de départ. L'opérateur Φ est décomposé en deux parties : l'une suit les vecteurs de l'argument et l'autre suit les vecteurs de son bord.

La figure 4.5, montre un exemple issu de [For98b], d'un calcul de flot. Intuitivement le bord de l'arête (ab) doit s'étendre vers l'arête (ac) pour suivre le vecteur $(a) \rightarrow (ac)$ ainsi que vers l'arête (be) pour suivre le vecteur $(b) \rightarrow (be)$. L'arête (ab) , quant à elle, doit se déformer pour suivre la bordure $(ad) + (db)$. Finalement, nous souhaitons avoir intuitivement $\Phi(ab) = (ca) + (ad) + (db) + (be) = (ac) + (ad) + (bd) + (be)$ en travaillant sur le corps binaire pour éviter les subtilités liées à l'orientation.

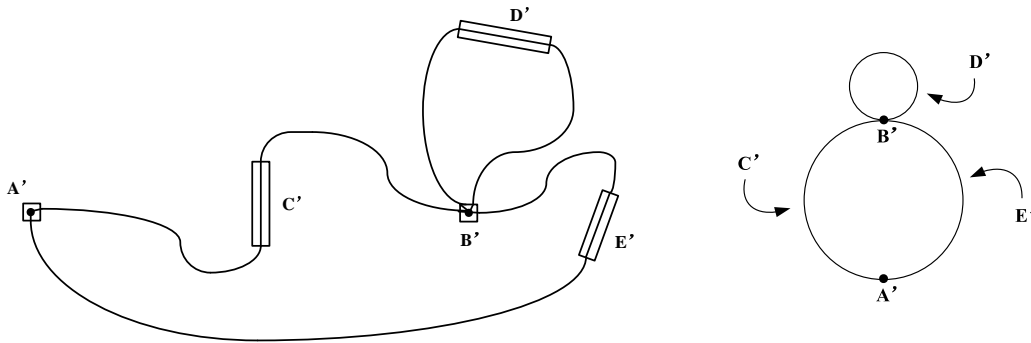


FIGURE 4.6 – Complexe de Morse discret associé et sa version simplifiée

La formule (4.5) donne sur le corps binaire

$$\begin{aligned}
 \Phi(ab) &= (ab) + \partial V(ab) + V \partial(ab) \\
 &= (ab) + \partial(abd) + V((b) + (a)) \\
 &= (ab) + ((ab) + (bd) + (ad)) + V(b) + V(a) \\
 &= (ab) + (ab) + (bd) + (ad) + (be) + (ac) \\
 \Phi(ab) &= (bd) + (ad) + (be) + (ac),
 \end{aligned} \tag{4.6}$$

ce qui est le résultat attendu.

4.1.5 Construction du complexe de Morse

Le flot peut ensuite être itéré k fois pour obtenir l'application Φ^k . Si de plus, ce flot est acyclique, condition que nous développerons par la suite, alors par la finitude du complexe X , l'application Φ^∞ est bien définie.

Les chaînes qui sont invariantes par Φ^∞ sont primordiales car elles permettent la construction du complexe de Morse associé au champ discret de vecteurs. Or chaque chaîne invariante contient une cellule critique.

Le complexe de Morse est donc construit en associant à chaque cellule critique σ de X une cellule σ' de même dimension. Le bord de $\Phi^\infty \sigma$ contient aussi des cellules critiques qui seront associées au bord de σ' .

Ce processus a été utilisé pour calculer le complexe de Morse associé au complexe de la figure 4.4. Le résultat est le complexe dessiné à gauche de la figure 4.6. Ce complexe ne contient que cinq cellules : trois arêtes et deux sommets.

En itérant le flot Φ sur l'arête C nous obtenons finalement un chaîne dont le bord est la chaîne $A + B$. Par conséquent la cellule C' du complexe de Morse a pour bord $A' + B'$. Le même calcul permet d'obtenir le bord de D' et de E' comme le montre la figure 4.6. Une version plus lisible se retrouve à droite. On remarque alors que le complexe de Morse

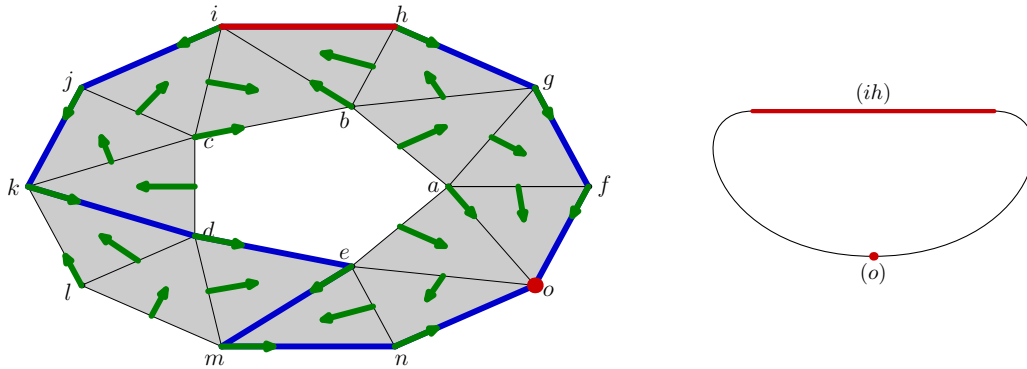


FIGURE 4.7 – Exemple d'un champ de vecteur sur l'anneau et son complexe de Morse associé

a $\beta_0 = 1$ et $\beta_1 = 2$ tout comme le complexe initial mais ce complexe est une version très simplifiée.

La raison pour laquelle la construction précédente est correcte est la suivante. En démarrant par un complexe X_0 , on calcule le complexe équivalent $X_1 = \Phi(X_0)$, puis par itérations successives $X_k = \Phi(X_{k-1}) = \Phi^k(X_0)$. Si Φ n'est pas dégénéré ce processus atteint un point fixe $X_\infty = \Phi^\infty(X_0)$. Forman a montré que Φ ne modifie pas l'homologie et donc $H_*(X_\infty) = H_*(X_0)$. Par contre, Φ simplifie le complexe d'où le gain de cette approche.

La construction nécessite la possibilité de calculer Φ^∞ . Par conséquent, il est nécessaire qu'aucune orbite du système dynamique défini par Φ ne soit périodique. Il est possible de se passer de cette condition nécessaire en complexifiant la construction du complexe de Morse et en considérant des cellules autres que critiques [For98c].

4.1.6 Exemple de l'anneau

La figure 4.7 présente un nouvel exemple sur la construction du complexe de Morse associé à un anneau.

Un champ discret de vecteurs, indiqués par des flèches vertes, a été construit sur le complexe issu de la triangulation de l'anneau. Les cellules critiques sont l'arête (ih) et le sommet (o) . Par conséquent le complexe de Morse comporte l'équivalent de ces deux cellules. En suivant le flot $\Phi^\infty(ih)$ nous aboutissons finalement au cycle indiqué en bleu $(ij) + (jk) + (kd) + (de) + (em) + (mn) + (no) + (of) + (fg) + (gh) + (ij)$. Ce cycle a pour bord $2(o)$ qui fournit le bord de la cellule (ih) dans le complexe de Morse comme indiqué sur la droite de la figure 4.7.

Si nous suivons le flot donné par le champ de vecteur, à la manière du cas continu de la figure 4.1, nous remarquons que l'anneau se contracte pour finalement donner le cycle bleu qui entoure le trou de l'anneau.

Ainsi le champ discret de vecteur, qui est une donnée combinatoire locale par essence, permet de simplifier fortement le complexe initial. Par exemple, l'anneau comporte 15 sommets, 30 arêtes et 15 triangles alors que le complexe de Morse associé ne comporte qu'un sommet et qu'une arêtes, nombre minimal de cellules critiques selon la version faible des inégalités de Morse (4.2).

De plus, cette forte simplification est faite sans perdre l'information concernant les générateurs des groupes d'homologie qui sont récupérables directement en suivant le flot à partir des cellules critiques, comme le montre l'exemple qui reconstruit un cycle autour du trou à partir de (ih) .

4.1.7 La fonction potentielle

Il peut sembler étrange que la version différentielle de la théorie de Morse considère en premier lieu une fonction potentielle pour en déduire un champ de vecteur alors que la théorie de Morse discrète commence par l'équivalent du champ de vecteur sans prendre en compte une fonction potentielle. Cette approche est un choix de présentation de notre part.

En effet, Forman commence pour définir une fonction potentielle f , encore appelée une fonction de Morse discrète, qui vérifie les deux conditions

$$\begin{aligned} \#\{\sigma > \tau \mid f(\sigma) \leq f(\tau)\} &\leq 1, \\ \#\{\sigma < \tau \mid f(\sigma) \geq f(\tau)\} &\leq 1 \end{aligned} \tag{4.7}$$

où $\#\{A\}$ est le nombre de paires vérifiant la proposition A et $\sigma < \tau$ dénomme deux cellules dont σ est sur le bord de τ .

Une telle fonction est une fonction potentielle qui permet de définir un champ discret de vecteurs

$$V\sigma = \tau \iff f(\sigma) > f(\tau). \tag{4.8}$$

Alors que les cellules critiques de ces fonctions sont définies par

$$\#\{\sigma > \tau \mid f(\sigma) \leq f(\tau)\} = 0 \tag{4.9}$$

ou

$$\#\{\sigma < \tau \mid f(\sigma) \geq f(\tau)\} = 0, \tag{4.10}$$

les deux conditions étant exclusives entre elles.

Les deux concepts de champs discret de vecteur et de fonction de Morse sont fortement reliées car il est possible de passer de l'un à l'autre s'il n'existe pas de séquence de cellules cycliques de la forme

$$\sigma_0 < \tau_1 > \sigma_1 < \tau_1 > \cdots < \tau_{n-1} > \sigma_n = \sigma_0 \tag{4.11}$$

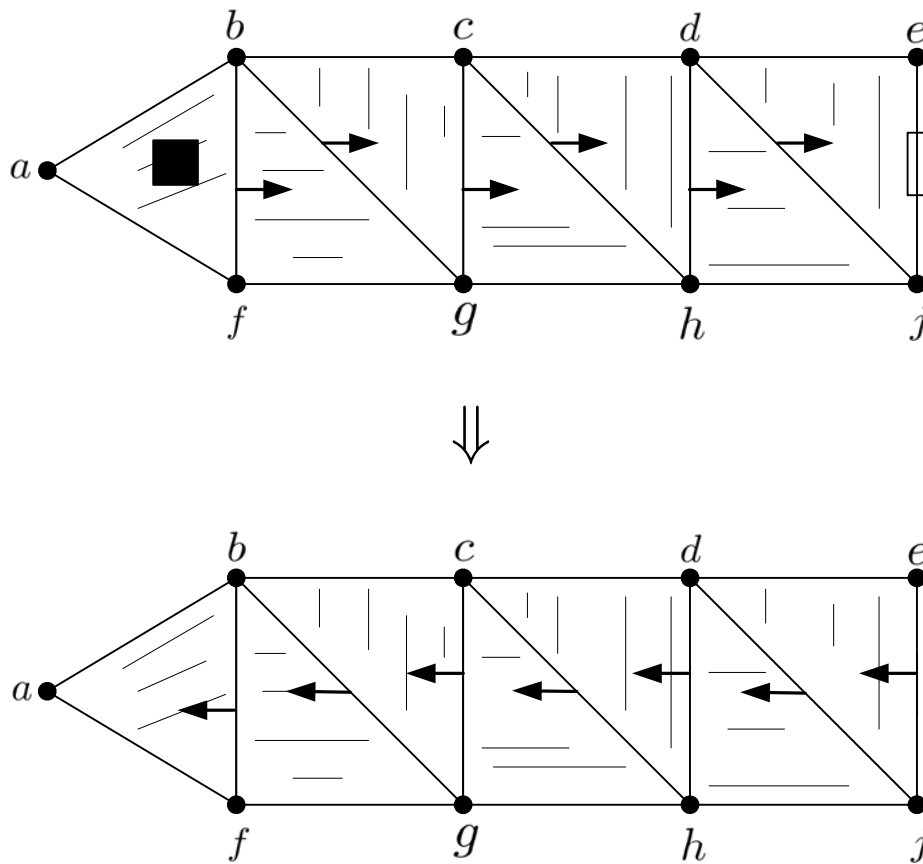


FIGURE 4.8 – Exemple de Forman de simplification de Morse

tel que

$$f(\sigma_0) \geq f(\tau_1) \geq f(\sigma_1) \geq f(\tau_1) \geq \cdots \geq f(\tau_{n-1}) \geq f(\sigma_n) = f(\sigma_0). \quad (4.12)$$

Cette dernière condition équivaut à un gradient qui ne peut pas être rotationnel en analyse vectorielle pour la version différentielle de la théorie de Morse. En effet, il est possible sous cette condition de construire à partir d'un champ de vecteur un fonction potentielle de proche en proche à l'instar de la démonstration du lemme de Poincaré.

4.1.8 Simplification du complexe de Morse

À l'instar de la version différentielle, la théorie de Morse discrète permet de simplifier le complexe de Morse en modifiant le champ de vecteur discret.

En effet, considérons le complexe de la figure 4.8 issu de l'article de Forman [For02b]. Sur la figure du haut, les cellules (abf) et (ej) sont critiques. Les flèches indiquent le champ de vecteurs discrets telles que $(cg) \rightarrow (cgh)$. Les vecteurs pour les arêtes extérieures ne sont pas dessinés pour alléger la figure mais nous supposons qu'ils existent.

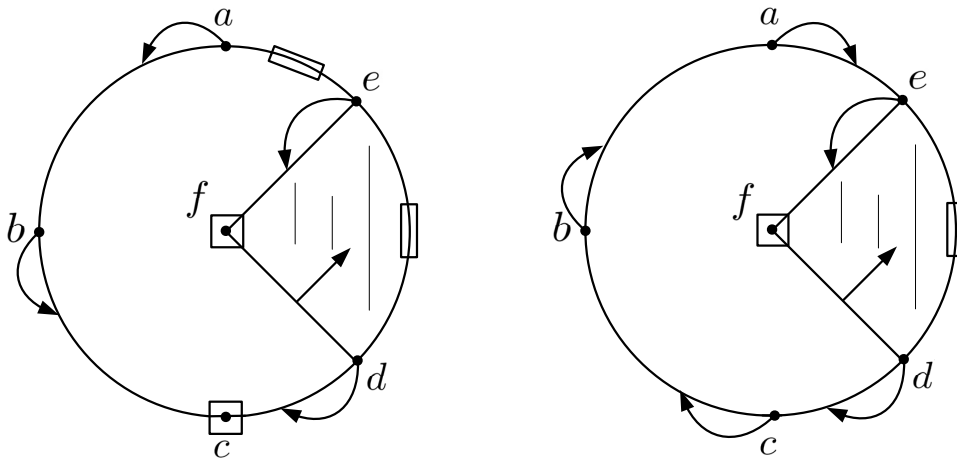


FIGURE 4.9 – Exemple de simplification de Morse d'un complexe homologue à un cerclce

Le champ de vecteurs définit un chemin de la cellule critique (abf) vers la cellule critique (ej) donné par la série de cellules

$$(abf), (bf) \rightarrow (bfg), (bg) \rightarrow (bcg), (cg) \rightarrow (cgh), (ch) \rightarrow (cdh), (dh) \rightarrow (dhj), (dj) \rightarrow (dej), (ej). \quad (4.13)$$

Ce chemin alterne les cellules de dimension 2 et des cellules de dimension 1. Il y a entre deux cellules adjacentes dans le chemin une relation de face/coface ou un vecteur discret.

Si nous reprenons ce chemin dans le sens inverse et que nous renversons les vecteurs pour suivre cette nouvelle direction, alors les deux cellules critiques extrêmes deviennent appariées et ne sont alors plus critiques, comme l'indique le nouveau champ de vecteurs discrets dessiné au bas de la figure 4.8.

Ce type de modifications du champ de vecteurs discrets permettent de simplifier le complexe de Morse. Les invariants homologique ne sont pas par contre pas modifiés par cette procédure de simplification. Par exemple, la caractéristique d'Euler-Poincaré n'est pas modifiée car les dimensions des deux cellules critiques qui disparaissent sont l'une paire et l'autre impaire.

La méthode de simplification est à rapprocher des algorithmes de couplage ou d'appariement dans un graphe qui utilisent des chemins augmentant tels que l'algorithme hongrois d'affectation [Kuh55, Mun57], l'algorithme d'Edmonds [Edm65] ou l'algorithme de Hopcroft-Karp [HK73].

La figure 4.9 montre un exemple sur un complexe homologue à un cercle. Sur la figure de gauche, en suivant, le flot partant du bord de la cellule critique (ae) nous arrivons d'un côté à la cellule critique (c) et de l'autre au second point critique (f) . Considérons le chemin $(ae), (a) \rightarrow (ab), (b) \rightarrow (bc), c$ qui contient deux vecteurs, nous pouvons le renverser pour obtenir les trois nouveaux vecteurs $(c) \rightarrow (bc), (b) \rightarrow (ab)$ et $(a) \rightarrow (ae)$ tels que le montre la droite de la figure 4.9. Cette modification a donc échangé deux cellules critiques pour un vecteur discret.

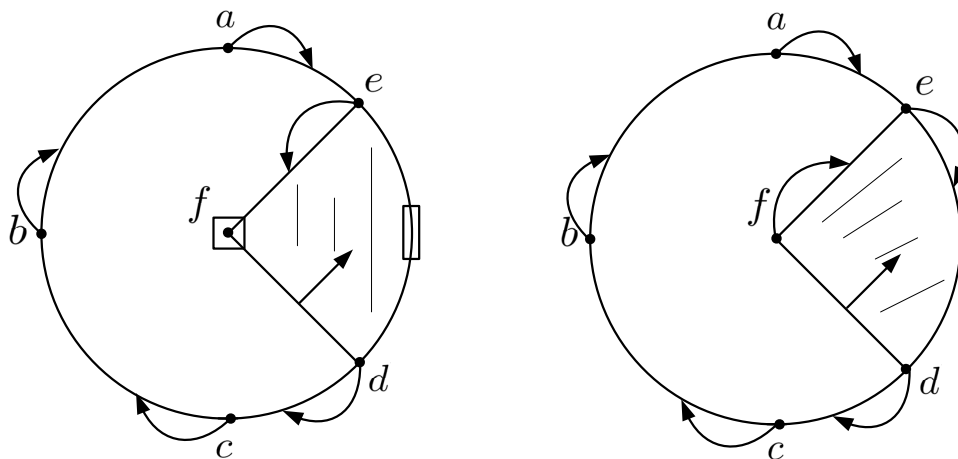


FIGURE 4.10 – Exemple de l'échec d'une simplification de Morse abusive d'un complexe homologue à un cercle

Le complexe de Morse devient alors particulièrement simple : une arête associée à la cellule critique (ed) dont le bord est un point associé au sommet critique (f) . Nous avons donc bien un espace équivalent à un cercle.

Il faut toutefois éviter de faire certaines simplifications excessives. En effet, considérons le résultat précédent redonné à gauche de la figure 4.10.

Il est possible de construire deux chemins entre l'arête critique (ed) et le sommet critique (f) . En effet, d'un côté nous avons le chemin court $(e), (f)$ et de l'autre le chemin plus long $(d), (c), (b), (a), (e), (f)$ — ces deux chemins sont donnés pour plus de lisibilité.

Il est tentant de renverser les vecteurs d'un de ces chemins par exemple le premier pour obtenir le complexe de droite. Mais alors il n'y a plus de cellule critique et le complexe de Morse est vide. Cette simplification est en effet abusive car elle indiquerait que le cercle est homologue à un espace vide !

Le problème précédent est issu de la création d'un chemin cyclique $(e), (d), (c), (b), (a), (e), \dots$ qui ne doit pas apparaître sous peine de casser l'équivalence entre le complexe de Morse et le complexe initial. En effet, le flot implique un orbite périodique comme nous l'avons vu dans la section 4.1.5 et il n'y a plus d'existence de la fonction potentielle. Naturellement, comme nous l'avons évoqué, l'utilisation de la forme plus développée de la théorie de Morse discrète de Forman décrite dans [For98c] reste possible mais à un prix calculatoire trop grand pour nos applications.

Il faut donc restreindre l'application des simplifications du champ de vecteurs discrets. Forman donne dans le théorème 9.1 de [For02b] une condition simple pour la procédure décrite ci-dessus soit toujours théoriquement correcte et que le complexe de Morse reflète bien le complexe de base. Il suffit en effet qu'il n'existe qu'un chemin unique entre les deux cellules critiques. Dans l'exemple précédent, il y avait bien deux chemins de l'arête

critique (ed) vers le sommet critique (f), l'un direct, l'autre faisant le tour du cercle.

4.1.9 Construction aléatoire du champ discret de vecteur

L'article [BL14] décrit un algorithme rapide de construction d'un champ discret de vecteurs. Cet algorithme est aléatoire dans le sens où le champ de vecteur ne sera pas nécessairement identique à chaque réalisation. Les expériences montrent que cette méthode de construction du champ de vecteur et du complexe de Morse associé sont souvent optimaux. Cette constatation est à la base du développement de notre algorithme de suivi présenté à la section 4.2.

Algorithme 4.1 : Construction aléatoire d'un champ discret de vecteurs

Input : Un complexe cellulaire X
Output : Un champ discret de vecteurs

```

1 indéfinies  $\leftarrow$  toutes les cellules de  $X$ 
2 while indéfinies est non vide do
3   if il existe une cellule dans indéfinies libre et de codimension 1 then
4     Choisir  $\sigma$  une cellule libre et de codimension 1
5     Enlever  $\sigma$  de indéfinies
6      $\tau \leftarrow \text{CoFace}(\sigma)$ 
7     Enlever  $\tau$  de indéfinies
8     Ajouter le vecteur  $\sigma \rightarrow \tau$  à vecteurs
9   else
10    Choisir une cellule  $\sigma$  de dimension maximale
11    Enlever  $\sigma$  de indéfinies
12    Ajouter  $\sigma$  à critiques
13  end
14 end

```

L'algorithme de [BL14] peut être réécrit sous la forme de pseudo-code de l'encadré 4.1. Initialement toutes les cellules sont non appairées et sont par conséquent dans la collection indéfinies. Tant qu'un couple de cellules (σ, τ) existe tel que σ soit une face de τ , que ces deux cellules soient dans indéfinies et que τ n'ait pas de coface dans cette même collection alors le couple forme un nouveau vecteur qui est ajouté au champ de vecteurs vecteurs. Si aucune paire n'existe alors une cellule de plus grande dimension de indéfinies est déclarée critiques en la plaçant dans la collection critiques.

L'aspect aléatoire de cet algorithme repose sur la façon de faire le choix des cellules à mettre dans critiques ou pour construire un nouveau vecteur de vecteurs.

Le choix des structure de données est primordial pour la rapidité d'exécution de cet algorithme. Par exemple la collection indéfinies peut être mise en œuvre par une queue de priorité dont le contenu serait des quadruplets dont les entrées sont la codimension, la

dimension, la cellule et un un nombre aléatoire. Cela permet d'ordonner les éléments par codimension croissante puis par dimension décroissante. Le nombre aléatoire casse les cas d'égalité et permet au choix de se faire au hasard.

De même la collection **critiques** des cellules critiques peut être un ensemble mis en œuvre sous forme d'arbre et un dictionnaire ou une table d'association est un bon choix pour le champ vecteurs.

4.2 Algorithme proposé pour le suivi des groupes d'homologie

Comme nous avons vu ci-dessus, la théorie de Morse discrète permet de simplifier un espace combinatoire X sans dénaturer sa structure topologique. Cette simplification est rendue possible par l'utilisation d'un champ de vecteur discret V .

Comme les groupes d'homologie $H_*(X)$ sont isomorphes aux groupes $H_*(M_V)$ du complexe de Morse associé à V , leur calcul est bien plus simple. De plus, le champ de vecteur V est une donnée combinatoire, représentable par une liste d'association, qui est foncièrement localisée car un vecteur ne fait intervenir qu'une paire face/coface. La seule difficulté provient de la non-dégénérescence du champ de vecteurs.

Supposons maintenant que l'espace X puisse être modifié dans le temps par ajout ou retrait successif de cellules. Il est intéressant de pouvoir suivre l'évolution des groupes $H_*(X)$ sans toutefois devoir recalculer ces groupes *ex nihilo*. Nous proposons dans cette section d'utiliser la théorie de Morse discrète pour mettre à jour $H_*(X)$.

Cette approche et l'algorithme que nous développons dans la suite a été l'objet de notre article [ZGC14] qui répond en partie aux interrogations posées et aux perspectives données par l'article précédent [ZGC13].

4.2.1 Principe de l'algorithme

L'évolution d'un complexe en fonction du temps peut être décomposé en deux opérations élémentaires qui transforment le complexe $X = X_t$ en un nouveau complexe $Y = X_{t+1}$ à l'instant suivant :

Ajout d'une cellule qui agrandit le complexe X . Il faut toutefois que le résultat Y de cet ajout reste un complexe bien défini. C'est pourquoi nous imposons que les cellules aux bords de la cellule ajoutée soient déjà dans le complexe à l'instant précédent X ;

Suppression d'une cellule qui enlève une cellule du complexe. Pour la cohérence de l'opération, il faut que la cellule supprimée ne soit pas sur le bord d'une autre cellule. La condition que la cellule supprimée ne soit pas une coface de X doit donc être satisfaite.

Les contraintes imposées ci-dessus ne nuit pas à la généralité de notre approche mais elles imposent parfois un ordre des opérations élémentaires lorsque plusieurs insertions et plusieurs suppressions de cellules ont lieu au même moment dans l'application.

L'idée de notre algorithme est de construire un champ discret de vecteurs V sur le complexe initial puis de remettre à jour localement le champ V pour chacune des opérations élémentaires de modification du complexe. L'homologie peut alors être calculée à partir des cellules critiques du champ en suivant le flot comme expliqué plus haut.

La localisation des cellules telle qu'elle est expliquée à la section 3.3.6.1 permet aussi de rendre locale la représentation du champ de vecteur car un vecteur est une paire face/coface gérée par le même nœud. Comme nous le verrons, les mises à jour du champ V restent aussi localisées.

Pour calculer le groupe d'homologie, plusieurs solutions existent :

- si une forme approchée est suffisante, alors le comptage des cellules critiques peut se révéler suffisant ;
- si des cycles générateurs de la forme approchée sont souhaités, il est simple de suivre le bord du flot des cellules critiques grâce à un algorithme de type routage qui calcule le résultat en sautant de capteurs en capteurs ;
- si le groupe d'homologie exact et complet est désiré, la forme approché avec les flots calculés par l'algorithme du point précédent peut être centralisé pour que l'homologie du complexe de Morse soit finie. Cette opération ne demande ni un grand investissement en calcul ni en communication car le complexe de Morse associé est en pratique bien plus petit que le complexe initial.

L'initialisation du champ V dépend de l'application. Par exemple, seuls les sommets sont présents dans le complexe et dans ce cas il n'y a aucun vecteur. Si le complexe existe *a priori*, une initialisation aléatoire du champ V telle que la construction de [BL14] peut se révéler profitable.

4.2.2 Mise à jour incrémentale

L'algorithme que nous proposons remet à jour le champ discret de vecteur V . Lors de cette mise à jour l'état des cellules peuvent prendre trois valeurs :

- indéfini** qui indique que la cellule est étudiée par la méthode. Cet état est temporaire ;
- critique** qui indique que la cellule est une cellule critique du champ de vecteur ;
- appairée** qui signifie que la cellule est déjà associée avec une autre cellule dans un vecteur du champ.

Les cellules susceptibles d'être appairées avec la modification du complexe passent à l'état indéfini. Ensuite, un appariement glouton [BL14] est effectué sur les cellules dont l'état est indéfini. Un fois fait, les cellules à l'état encore indéfini passent à l'état critique.

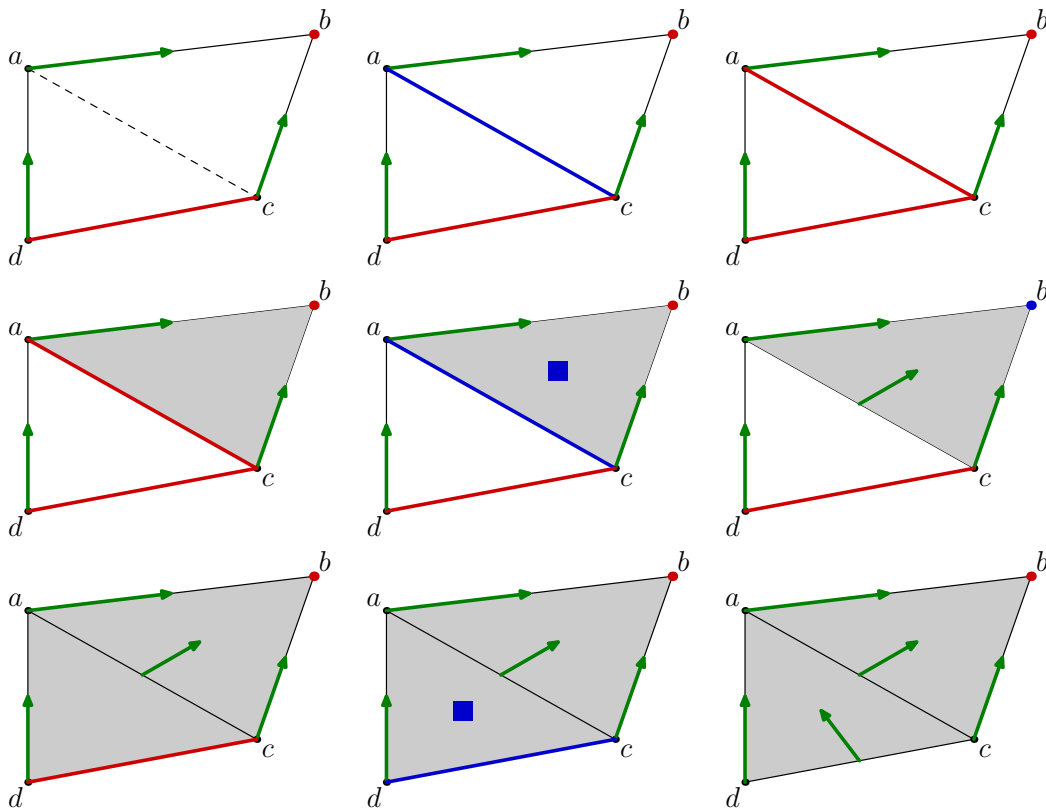


FIGURE 4.11 – Étapes de la mise à jour lors d'ajout de cellules

4.2.2.1 Mise à jour en cas d'insertion d'une cellule

Supposons qu'une nouvelle cellule σ apparaisse dans le complexe et que ses bords y sont déjà présents. Le champ de vecteur doit être modifié pour prendre en compte cet ajout. La cellule σ et ses faces qui sont critiques passent à l'état indéfini. L'appariement glouton est alors effectué et les cellules non appariées deviennent critiques.

La figure 4.11 montre les différentes étapes lors de l'ajout de la diagonale et des triangles au quadrilatère initial. Les opérations de bases sont dans l'ordre : 1. ajout de la diagonale ac ; 2. ajout du triangle abc ; 3. ajout du triangle acd .

L'ajout de la diagonale se déroule selon :

1. l'état de la cellule (ac) de la diagonale est indéfini. Comme les sommets (a) et (b) ne sont pas critiques, seule (ac) est dans cet état ;
2. aucun appariement entre cellules indéfinies n'est possible ;
3. (ac) devient une cellule critique.

Puis l'ajout de la cellule (abc) s'accompagne des opérations suivantes :

1. (abc) est classé indéfini tout comme la diagonale (ab) et le sommet b qui sont sur son bord et qui sont critiques ;

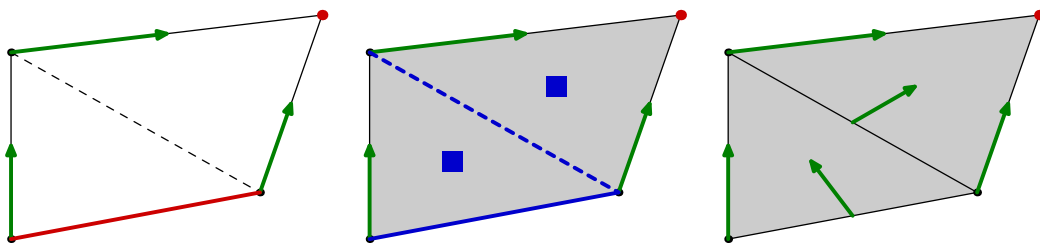


FIGURE 4.12 – Mise à jour lors de l'ajout d'un ensemble de cellules

2. l'appariement entre les cellules (ac) et (abc) se produit car ces deux cellules sont indéfinies et leur état passe à apparié;
3. il ne reste pas de cellule indéfinie, donc aucune ne devient critique.

Enfin la dernière opération, l'ajout du triangle acd :

1. (acd) est classé indéfini tout comme le bord (cd) qui était critique;
2. l'appariement entre les deux cellules précédentes, (acd) et (cd) est ajouté au champ discret de vecteurs;
3. il ne reste plus de cellules critiques.

Nous remarquons dans cet exemple que le complexe de Morse associé est toujours optimal au sens où les inégalités (4.2) sont des égalités. Dans ce cas la méthode peut être qualifiée d'optimale. Malheureusement nous n'aurons pas cette garantie pour d'autres exemples ou dans les simulations.

Les opérations élémentaires peuvent ici s'effectuer ensemble. Dans ce cas, la méthode est encore correcte. Les trois cellules (ac) , (abc) et (acd) sont ajoutées en même temps, voir la figure 4.12. Les cellules ajoutées ainsi que leur faces qui sont critiques passent à l'état indéfini. L'appariement se fait alors aussi de façon gloutonne. Les cellules restant indéfinies deviennent critiques. En fait, ces deux méthodes, par opérations élémentaires et par groupement pour l'ajout sont équivalentes car l'appariement glouton de la seconde méthode revient à faire un ordonnancement des opérations élémentaires d'insertion.

4.2.2.2 Mise à jour en cas de suppression d'une cellule

Considérons maintenant le cas d'une cellule σ qui disparaît du complexe et qui n'est pas sur le bord d'une autre cellule.

La figure 4.13 illustre la mise à jour du champ discret de vecteurs lorsque des cellules du complexe sont supprimées. L'ordre des suppression est ici acd , abc et ac .

La suppression du triangle acd s'accompagne de la suite d'opérations suivantes

1. le statut des cellules (c) et (cd) devient indéfini car (c) est une cellule critique sur le bord du triangle au sens large. L'arête (cd) est aussi classée comme indéfinie car elle participait à un appariement avec la cellule supprimée (acd) qui n'est plus cohérent avec le complexe;

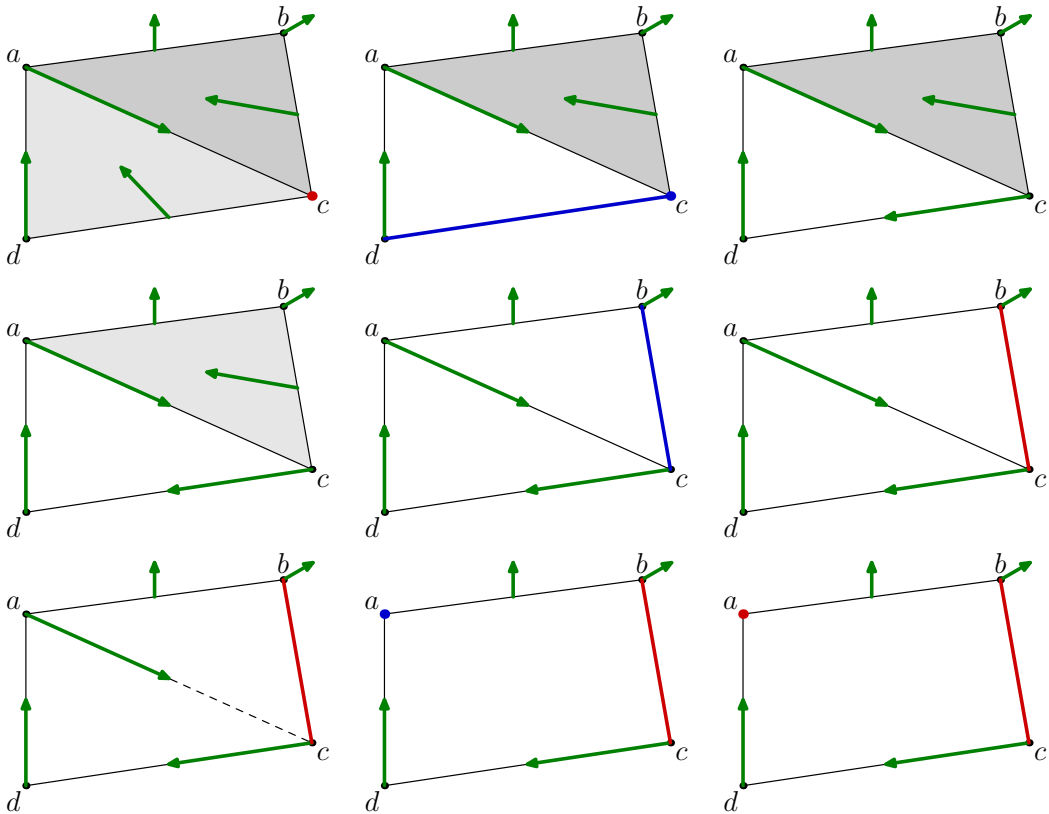


FIGURE 4.13 – Étapes de la mise à jour lors de retraits de cellules

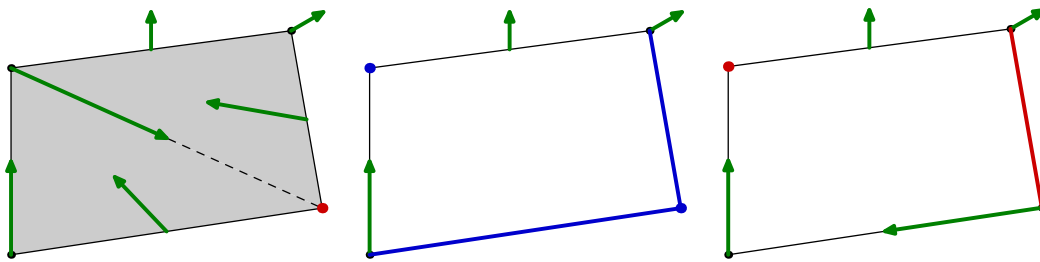


FIGURE 4.14 – Mise à jour lors du retrait d'un ensemble de cellules

2. l'appariement entre (c) et (cd) est ajouté au champ de vecteur ;
3. il ne reste aucune cellule pouvant devenir critique.

Nous insistons sur le fait que toutes les cellules sur le bord des cellules enlevées ou ajoutées doivent être examinées pour savoir si elles deviennent indéfinies ou non. En effet la suppression ou l'ajout d'une cellule de dimension d peut affecter le champ de vecteur par leur appariement avec les cellules de dimension $d - 1$ mais aussi en faisant apparaître des appariements entre les cellules du bord comme entre les cellules (c) et (cd) de l'exemple.

Le triangle abc est maintenant supprimé entraînant les étapes

1. le statut de (bc) passe de apparié à indéfini car l'arête participe avec (abc) à un vecteur qui n'a plus lieu d'être ;
2. aucun appariement n'est possible ;
3. l'arête (bc) devient critique.

Enfin, le retrait de ac provoque

1. le statut de (a) qui était apparié avec (ac) par le champ de vecteurs devient maintenant indéfini ;
2. aucun appariement n'est possible ;
3. la cellule (a) devient une cellule critique du champ de vecteur.

Ici encore, nous pouvons constater que le complexe de Morse associé reste optimal tout au long des étapes et que le résultat final correspond à un seul trou dans le complexe car il y a une seule nouvelle arête critique dans le champ final.

Nous pouvons faire la même remarque que pour l'ajout d'une cellule : les opérations de suppression peuvent être regroupées mais alors l'appariement glouton ordonnancera l'ordre des opérations élémentaires selon la mise en œuvre qui en sera fait. La méthode avec les opérations de suppression regroupée reste la même : toutes les cellules sur le bord des cellules supprimées deviennent indéfinies tout comme les cellules qui étaient appariées avec une cellule supprimée. L'appariement glouton construit les nouveaux vecteurs dès que possible et les cellules deviennent critiques si ce n'est plus le cas.

4.2.2.3 Dégénérescence du champ de vecteurs

Les opérations précédentes n'interviennent que très localement car les vecteurs correspondent à des associations entre des cellules et leurs faces. La distribution de cet algorithme est donc simple.

Il est toutefois possible que le complexe de Morse associé au champ de vecteur ainsi construit ne deviennent qu'approximatif car le champ de vecteur devient parfois dégénéré. C'est-à-dire que les fonctions potentielles associées au champ ne soit plus irrotationnel dans le sens de (4.11) et (4.12).

Soit l'approximation est gérable par l'application qui ne souhaite qu'une vision grossière des résultats soit notre méthode doit être adaptée pour interdire la dégénérescence.

Trois approches sont possibles pour corriger notre algorithme :

1. nous pouvons par exemple étendre notre algorithme en utilisant une forme plus générale de la théorie de Morse de Forman telle que celle présentée dans [For98a]. Le champ de vecteur peut devenir dégénéré mais la construction du complexe de Morse devient plus compliquée car en plus des cellules critiques, il faut aussi prendre en compte les orbites périodiques sous le flot Φ . Cela peut se faire grâce à un algorithme de détection de cycle tel que l'algorithme du lièvre et de la tortue de Floyd [Flo67] ou la version améliorée que Brent a introduit pour la factorisation d'entiers [Bre80]. Nous n'avons pas étudié la distribution d'un tel algorithme ;
2. maintenir une fonction potentielle en même temps que le champ de vecteur semble être une méthode simple car alors l'appariement ne sera possible que si la fonction potentielle le permet. Toutefois, la localisation du champ de vecteur est perdue au profit d'une construction globale : celle de la fonction potentielle. C'est pourquoi nous n'en avons fait aucune mise en œuvre ;
3. l'approche que nous avons choisi dans [ZGC14] est plus pragmatique : nous avons introduit la date de naissance des cellules et un vecteur n'est autorisé que si l'âge des cellules va en grandissant si nous suivons sur 1 itérations le flot Φ . L'optimalité du complexe de Morse se perd mais la simplicité de l'approche en fait son point fort. La date de naissance des cellules n'est bien sûr qu'un prétexte à ordonner les cellules ; d'autres choix sont possibles comme un identifiant ou une fonction d'altitude, etc.

4.2.3 Comparaison avec d'autres algorithmes

Dans [DGJ12], les auteurs décrivent un algorithme distribué qui calcule les groupes d'homologie pour lequel le complexe de l'espace de données est aussi représentée localement. Cependant, leur algorithme effectue des simplifications locales par l'intermédiaire du complexe co/réduction afin d'obtenir une représentation plus mince du complexe qui est ensuite envoyé à un nœud central pour terminer le calcul de l'homologie, comme avec [EH10].

L'adaptabilité de la méthode de [DGJ12] est toutefois limitée par l'utilisation des co/réductions qui doivent être recalculées complètement à chaque modification du complexe. Notre solution, grâce aux champs de vecteurs, ne souffre pas de ce défaut.

L'approche de [GCK14b], ainsi que l'article associé [GCK14a] est différente de la notre car elle utilise la persistance homologique en zigzag [CS10]. Cette approche demande une connaissance complète de l'historique de la construction du complexe avant d'en faire l'analyse. Elle n'est donc pas adaptée à la situation d'un algorithme de suivi. Ensuite la filtration en zigzag demande des réunions du complexe entre deux instants qui demandent une gestion fine si la distribution de l'algorithme est envisagé. Enfin, il n'existe pas encore à notre connaissance d'algorithme distribué pour le calcul de la persistance en zigzag.

4.3 Application à la couverture de mesure

La suite de ce chapitre concerne l'application de notre algorithme à deux situations déjà rencontrées : la détection des trous de couverture à partir des corrélations entre les mesures des capteurs et à la sélection de capteurs actifs pour limiter les dépenses énergétiques superflues.

Nous avons repris la construction du complexe simplicial qui approche la couverture d'un réseau de capteur de notre article [ZGC13]. Toutefois ce complexe est amené à être modifié au cours du temps et l'algorithme de la section précédente est appliqué à son suivi.

Cette application utilise uniquement la détection des trous basée sur le nombre de Betti β_1 et les générateurs de $H_1(X)$. L'intérêt ici est de suivre l'évolution tout en maintenant la stabilité des générateurs pour ne pas avoir un un représentant d'un trou qui varie à chaque instant.

4.3.1 Description de l'application

Tout comme la section 3.3, un champ est construit qui est capté par un réseau de capteurs sans fil placés aléatoirement. Ces capteurs mesure avec une période dt le champ. À chaque instant t , ils en obtiennent $F_x(t)$ une version bruitée par un bruit blanc gaussien. Ces bruits sont indépendants les uns des autres et des mesures.

Chaque capteur estime le champ moyen à leur position $\bar{F}_x(t)$ par un filtrage récursif du premier ordre,

$$\bar{F}_x(t) = (1 - \alpha) \bar{F}_x(t - dt) + \alpha F_x(t), \quad (4.14)$$

où α est le coefficient d'oubli du filtre.

Ces moyennes servent ensuite à estimer des covariances σ_{xy} entre les capteurs x et y par le même type de filtrage :

$$\hat{\sigma}_{xy}(t) = (1 - \beta) \hat{\sigma}_{xy}(t - dt) + \beta \left(F_x(t) - \bar{F}_x(t) \right) \left(F_y(t) - \bar{F}_y(t) \right), \quad (4.15)$$

où β est un second coefficient d'oubli du même ordre de grandeur que α .

Dans nos simulations, le réseau de capteur est statique. Toutefois le complexe représentant la précision du champ se modifie au fur et à mesure de l'estimation des corrélations du champ. Bien sûr, ces estimations se calculent alors que le champ se modifie avec le temps. Par contre, les covariances du modèle restent constantes le long de la simulation et il en est de même pour la couverture de mesure.

À l'instar de la section 3.3.2, les capteurs x et y sont connectés si un seuil est plus grand que l'optimum

$$\min_{\mathbf{u}} \mathbf{u}^T \begin{bmatrix} \hat{\sigma}_{xx}(t) & \hat{\sigma}_{xy}(t) \\ \hat{\sigma}_{yx}(t) & \hat{\sigma}_{yy}(t) \end{bmatrix}^{-1} \mathbf{u} \quad (4.16)$$

sous les contraintes $\begin{bmatrix} \hat{\sigma}_{xx}(t) \\ \hat{\sigma}_{yy}(t) \end{bmatrix} \geq \mathbf{u} \geq \begin{bmatrix} \hat{\sigma}_{xy}(t) \\ \hat{\sigma}_{yx}(t) \end{bmatrix}$.

L'équation (4.16) n'est pas exactement la version estimée de l'équation (3.30) car cette dernière utilise la variance du champ en un point sans prendre le bruit en compte. Comme cela n'est pas fourni par les données, nous ne l'utilisons pas. Toutefois, cette constante est intégrée au seuil.

Ainsi le complexe se construit petit à petit jusqu'à la convergence. Des arêtes peuvent aussi disparaître dans le processus. Même si le champ varie, cette convergence n'est pas fortement affectée car le complexe dépend de la corrélation entre les mesures et non les mesures en elles-même.

4.3.2 Résultats des simulations

Le réseau de capteurs reste statique tout au long de la simulation. Les seules modifications sont celles dues à une mauvaise estimation, initialement, des covariances entre les mesures des capteurs.

Bien sûr, notre méthode est applicable pour des réseaux de capteurs mobiles ou pour des champs dont la précision varie avec le temps, etc. Mais nous voulons éviter la confusion des causes dans nos simulations et nous nous restreignons au RCSF statique.

La situation du réseau de capteurs à sept instants différents est illustrée sur les figures 4.15 et 4.16. L'algorithme fonctionne par itération. À chacune d'elle les capteurs affinent les estimations de la moyenne et des covariances entre elles. À partir de ces estimations, le complexe est remis à jour. Ce complexe est le complexe de cliques des arêtes ; c'est-à-dire que si trois arêtes d'un triangle sont dans le complexe alors le triangle l'est aussi. De même pour les tétraèdres. À partir des cellules qui apparaissent et qui disparaissent, le champ discret de vecteurs est remis à jour selon la méthode précédente.

Dans la simulation, les coefficients α et β des filtres valent $5 \cdot 10^{-3}$. La variance du bruit de mesure est de 10^{-2} . Une cinquantaine de capteurs sont positionnés à l'intérieur d'un

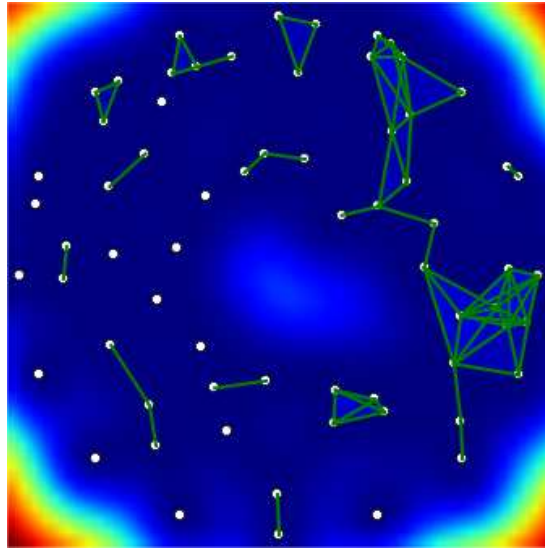


FIGURE 4.15 – Début de la construction du complexe

cercle délimité par seize capteurs. Les positions des capteurs intérieurs sont choisis aléatoirement et uniformément. Certains d’entre eux ont été repositionnés pour créer un trou de couverture au centre.

Le champ généré est un processus gaussien tridimensionnel — deux dimensions spatiales et une dimension pour le temps — de moyenne nulle et dont la fonction de covariance est basé sur un noyau gaussien dont l’échelle spatiale ℓ est fixée à 0.4 à mettre en perspective avec le diamètre de 2 du réseau de capteur et l’échelle temporelle τ est fixée à 1. La covariance entre le champ à l’instant t à la position x et le champ à l’instant t' à la position x' est donc donnée par

$$k((x, t), (x', t')) \propto \exp\left(-\frac{\|x - x'\|^2}{2\ell^2} - \frac{|t - t'|^2}{2\tau^2}\right). \quad (4.17)$$

Naturellement, ce modèle est inconnu des capteurs qui ne récupèrent que des mesures bruitées du champ à leur position.

Comme le nombre d’itérations de la simulation est initialement inconnue, nous n’avons pas généré un champ complet mais nous avons simulé le champ par un filtrage d’ordre 1 d’un processus gaussien bidimensionnel. Ce filtrage est simple à réaliser car le champ à un instant sachant sa réalisation à l’instant précédent est une variable aléatoire gaussienne. La séparabilité de l’espace et du temps dans le noyau (4.17) rend ce calcul plus efficace encore. La précision du modèle reste suffisant pour notre propos. De plus comme notre méthode ne repose pas spécifiquement sur ce modèle, elle se montre robuste.

Les instantanés de la figure 4.16 représentent le réseau aux itérations 670, 675, 686, 940, 960 et 1050 (a–f) qui est la convergence. La première remarque concerne la stabilité du cycle qui entoure le trou de couverture central. Cela s’explique par notre méthode qui ne

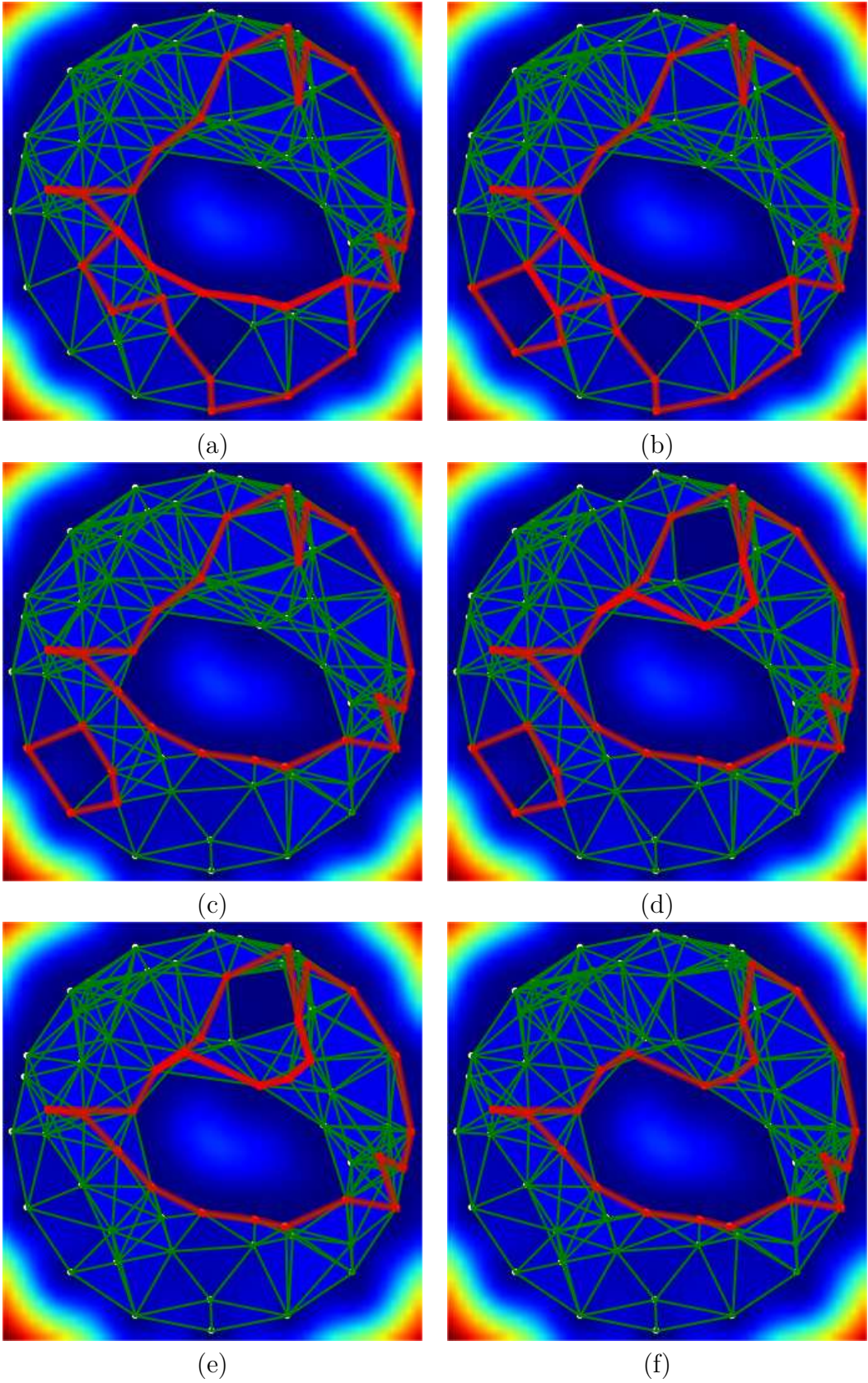


FIGURE 4.16 – Instantanés de la construction du complexe représentant la couverture

modifie pas le champ de vecteur sauf en cas de nécessité. Comme ce cycle apparaît assez tôt dans la construction, le champ de vecteur n'est que faiblement modifié aux alentours de ce dernier. Par conséquent, le générateur obtenu en suivant le flot de la cellule critique correspondante ne sera pas fondamentalement modifié. Ensuite les trous de couverture supposés dus à la mauvaise estimation des covariances entre les capteurs se rebouchent rapidement et, semblerait-il, dans l'ordre de leur sévérité.

4.4 Application à la gestion de l'énergie

Cette application utilise le critère de de Silva et Ghrist [dSG07b] pour la couverture qui repose sur le calcul de $H_2(X, F)$ où F est la bordure du réseau. Ce critère permet d'obtenir un ensemble suffisant de capteurs qui assurent à eux seuls la couverture.

La section 3.1.5 a déjà rapidement présenté cette application mais pas dans un cadre distribué. En effet, tous les calculs sont centralisés pour faire le calcul des groupes $H_2(X, F)$ à chaque instant et d'un représentant de son générateur.

Il s'agit ici, d'une part d'éviter de recalculer complètement ces groupes d'homologie à chaque instant et d'autre part de rendre les calculs locaux pour en permettre sa distribution. Comme nous l'avons vu, la théorie de Morse discrète et nos algorithmes de mise à jour du champ de vecteurs discrets se prêtent particulièrement bien à cet exercice.

Ces résultats ont fait l'objet d'une soumission à la conférence GretsI de 2015 [ZGC15].

4.4.1 Description de l'application

Un réseau de capteurs sans fil est déployé dans une zone et certains de ces capteurs forment la frontière F de la zone. Nous utilisons un modèle simple de couverture : chaque capteur couvre au moins un disque de rayon r_c . De plus les capteurs peuvent communiquer entre eux si leur distance est inférieure à r_b .

Le complexe représentant la couverture est alors construit comme étant le complexe de clique du graphe de communication. Il s'agit donc du complexe de Rips de rayon r_c .

Comme nous l'avons déjà vu, si le rayon de captage de chaque capteur est plus grand que le $r_c/\sqrt{3}$, alors le complexe de Rips approche suffisamment bien la couverture du réseau pour que le critère de couverture soit correct.

Un champ de vecteurs V est initialement construit sur le complexe de couverture. Ensuite, l'homologie $H_2(X, F)$ est calculée grâce au complexe de Morse associé au champ V . À partir du représentant de l'unique générateur de $H_2(X, F)$, un ensemble de sommets — donc de capteurs — sont maintenus éveillés alors que les autres passent dans un état endormis.

Chaque capteur a un stock d'énergie initial qui ne peut que s'épuiser en fonction du temps. Les capteurs endormis ne piochent pas dans leur réserve alors que les nœuds éveillés consomment de l'énergie. À chaque fois qu'un capteur a un stock d'énergie insuffisant, il disparaît du réseau. Ce faisant, le complexe de la couverture se remet à jour et par la même occasion le champ discret de vecteurs V s'adapte à cette modification en suivant l'algorithme de la section 4.2.

Le complexe de Morse est *de facto* aussi remis à jour et le représentant unique de $H_2(X, F)$, s'il existe, permet de réveiller certains capteurs et de mettre en sommeil d'autres capteurs. Si β_2 , la dimension de $H_2(X, F)$ est nul, alors le réseau de capteur n'est plus apte à assurer la couverture par manque de capteurs encore présents.

4.4.2 Cadre général des simulations

Dans la prochaine simulation, nous avons positionné les capteurs du réseau de façon plus homogène en utilisant un échantillonneur de Poisson rapide [Bri07]. La distance minimum entre deux capteurs est fixée dans la mise en œuvre de l'échantillonneur à 0.1 alors que le nombre de points de test est de 30.

À ces capteurs intérieurs sont ajoutés des capteurs à la bordure du champ qui est un disque de rayon 1. Ces capteurs forment la bordure F nécessaire au calcul de $H_2(X, F)$.

Initialement, deux capteurs sont connectés si leur distance est inférieure à un rayon fixé r_c qui est le rayon de communication. Dans nos simulations r_c est fixé à 0.3 alors que le diamètre du réseau est 2.

Une fois que les capteurs connaissent leurs voisins, un complexe simplicial de clique X est construit comme à la section 3.1.5. Il devient alors possible d'appliquer le critère de de Silva et Ghrist qui consiste à calculer $H_2(X, F)$.

4.4.3 Sélection des capteurs éveillés

La figure 4.17 indique le placement des capteurs avec leur identifiants. Le sous complexe F qui représente la bordure est formé des arêtes entre les nœuds du bord d'identifiant inférieur à 20.

Les arêtes du complexe sont représentées par des lignes vertes et chaque triangle du complexe est aussi dessiné en bleu avec transparence pour indiquer le niveau de couverture du domaine. Nous remarquons que la couverture est complète, sans trou.

Le niveau de bleu de la figure 4.17 indique que chaque point du domaine est même couvert plusieurs fois par plusieurs capteurs. Il est donc possible de limiter le nombre de capteurs en éveil pour accroître la durée de vie de ce réseau de capteurs.

À partir de ce complexe simplicial, nous avons utilisé notre variante de l'algorithme de [BL14] présentée à la section 4.1.9 afin d'y construire un champ discret de vecteurs.

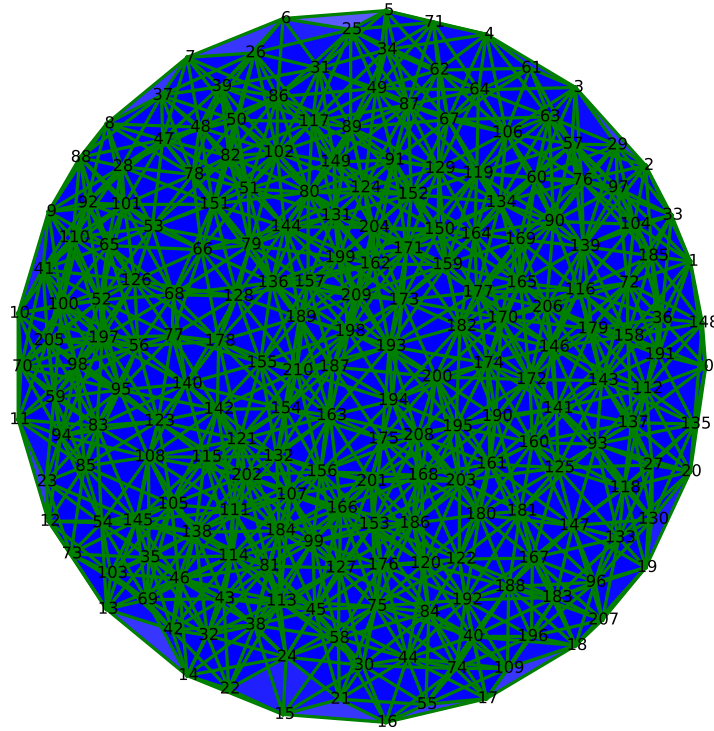


FIGURE 4.17 – Couverture initiale

Ce champs de vecteurs permet de calculer simplement comme nous l'avons vu lors de la présentation de la théorie de Morse discrète, un complexe homologiquement équivalent et plus simple. La difficulté est cependant de calculer un groupe d'homologie relative et non absolue. Malheureusement, la théorie de Morse présentée ci-dessus permet une représentation simplifiée d'un complexe et non d'un complexe à un sous-complexe près.

Toutefois, la théorie de l'homologie permet de transformer le calcul de l'homologie relative en un calcul d'homologie absolue. En effet, nous avons l'équivalence [Sat99],

$$H_*(X, A) \cong H_*(X/A, \star) \quad (4.18)$$

où $A \subset X$ est un sous-complexe et X/A est l'espace quotient du complexe X sur A . Ensuite, $H_*(X/A, \star)$ se calcule en prenant le cône de A .

Plus précisément, cette équation permet de calculer la classe d'homologie $H_2(X, F)$. En effet, ajoutons une cellule π à X dont le bord est donnée par F , c'est-à-dire $\partial\pi = F$, alors

$$H_2(X, F) \cong H_2(X \cup \pi). \quad (4.19)$$

Le calcul $H_2(X \cup \pi)$ par le champ de vecteurs discrets construit pour X reste simple en supposant que la cellule π est toujours critique. Si la couverture est garantie, ce sera toujours le cas. Par contre si la couverture n'est pas garantie alors il se peut que le complexe de Morse ne soit pas minimal — ce qui est sans gravité pour notre application.

Considérer π comme étant toujours critique dans le champ de vecteurs discrets permet aussi une simplification de la structure de donnée représentant le complexe. En effet, il n'est pas utile de l'inclure dans le complexe car aucun vecteur ne pourra y être associé.

Si $H_2(X, F)$ est non nul alors l'unique générateur sera nécessairement la chaîne invariante par le flot associée à π . Ainsi, le champ de vecteurs discrets permet le calcul d'un représentant du générateur de $H_2(X, F)$ car il suffit de savoir que $\partial\pi = F$,

$$H_2(X, F) \cong \langle \Phi^\infty(\pi) \rangle = \langle \Phi^\infty(\pi + V\partial\pi + \partial V\pi) \rangle = \langle \Phi^\infty(\pi + VF) \rangle, \quad (4.20)$$

en notant $\langle g \rangle$ l'espace vectoriel généré par g et en utilisant le fait que π soit critique, que le bord de π soit F et l'expression du flot (4.5).

Nous proposons ci-dessous deux solutions simples pour calculer un représentant de l'unique générateur du groupe $H_2(C_*, F)$. La première méthode repose directement sur la correspondance entre les cellules critiques du complexe C_* et les cellules du complexe de Morse via les chaînes invariantes par le flot créé par le champ de vecteur. La seconde méthode n'est pas aussi précise car elle suit plutôt le champs de vecteur directement mais elle reste particulièrement intéressante.

4.4.3.1 Première méthode

Une première solution consiste à suivre le flot de vecteur à partir des cellules classées critique tout en rejetant les simplexes associés à la bordure F . C'est à dire que la mise en œuvre correspond à l'équation (4.20).

L'algorithme 4.2 résume la procédure pour obtenir les capteurs à activés une fois le champ de vecteurs construits. Pour construire la chaîne invariante, il faut calculer $\Phi^k(\pi)$ jusqu'à obtenir un point fixe.

Algorithme 4.2 : Première méthode pour calculer les capteurs activés

Input : Un complexe cellulaire X et sa bordure F
Input : Un champ de vecteurs discrets V
Output : Liste des sommets participant à la couverture
Data : σ et τ , deux chaînes de $C_2(X)$

- 1 $\sigma \leftarrow VF$
- 2 **repeat**
- 3 $\tau \leftarrow \sigma$
- 4 $\sigma \leftarrow \tau + V\partial\tau + \partial V\tau$
- 5 $\sigma \leftarrow \sigma + VF$
- 6 **until** $\sigma = \tau$
- 7 **return** liste des sommets des simplexes de σ

Dans cet algorithme, à l'itération k , σ est la chaîne $\Phi^k(\pi) - \pi$ et τ est la chaîne $\Phi^{k-1}(\pi) - \pi$. La cellule π est soustraite dans σ et τ pour ne pas avoir à représenter cette cellule dans le

réseau. Les lignes 4 et 5 permet de prendre en compte π sans y faire directement référence. En effet, à l'itération k nous avons après la ligne 3 $\tau = \Phi^k(\pi) - \pi$ et nous cherchons à calculer $\sigma = \Phi^{k+1}\pi - \pi$, mais comme $\Phi\pi = \pi + V\partial\pi + \partial V\pi = \pi + VF$,

$$\begin{aligned}
\sigma &= \Phi^{k+1}\pi - \pi = \Phi(\Phi^k\pi) - \pi \\
&= \Phi(\tau + \pi) - \pi \\
&= \Phi\tau + \Phi\pi - \pi \\
&= \tau + V\partial\tau + \partial V\tau + \pi + VF - \pi \\
&= \tau + V\partial\tau + \partial V\tau + VF.
\end{aligned} \tag{4.21}$$

La ligne 4 calcule le flot de τ et la ligne suivante le corrige en y ajoutant VF . De cette façon, la cellule π n'intervient jamais dans cet algorithme et la couverture est garantie par la liste de sommets retournée.

La figure 4.18 montre l'évolution de la chaîne σ en utilisant l'algorithme 4.2 aux itérations 2, 3, 6, 7, 9 et 10 (a-f). Il en faut 17 pour que le point fixe soit atteint. Les nœuds en rouge sont ceux qui doivent être activés pour participer à la couverture. Les triangles rouges donne la couverture progressivement construite qui n'utilise que les sommets activés alors que les triangles bleus indique les cellules surnuméraires mises en sommeil pour éviter le gaspillage énergétique.

Comme attendu, nous remarquons que la couverture se construit de l'extérieur vers l'intérieur du domaine à couvrir. Toutefois la vitesse de cette construction dépend fortement de la structure du champ de vecteurs discrets comme nous pouvons le constater dans le quatrième cadran où la couverture est plus longue à se mettre en place.

La correction de l'algorithme précédent repose sur la non existence de chemin cyclique dans le flot Φ associé à V comme cela a été discuté à la section 4.2.2.3. Toutefois nous pouvons étendre notre cette procédure au cas où le champ est dégénéré en identifiant les chaînes périodiques sous l'action du flot qui contiennent π .

L'algorithme 4.3 présente la version de Floyd [Flo67] et l'algorithme 4.4 celle de Brent [Bre80] moins coûteuse en complexité car repérant une période plus rapidement.

L'idée de l'algorithme de Floyd 4.3 est de comparer $\Phi^k(\pi) - \pi$ contenu dans τ et $\Phi^{2k}(\pi) - \pi$ contenu dans σ . Après un nombre d'itération assez grand, si le flot devient périodique alors $\Phi^k = \Phi^{2k}$ et donc $\sigma = \tau$. Nous savons de plus que la période est un diviseur de k . Cette algorithme est aussi connu sous le nom de l'algorithme du lièvre, rôle joué par σ , et de la tortue, joué par τ .

L'inconvénient de cet algorithme est de devoir parfois calculer le flot pour un grand nombre d'itérations avant que la condition d'arrêt soit vérifier car il faut que le nombre d'itérations soit supérieur à la partie de construction et qu'il soit aussi un multiple de la période.

L'algorithme de Brent 4.4 est plus fin en comparant $\Phi^i(\pi) - \pi$, contenu dans τ et $\Phi^{i+r}(\pi) - \pi$, contenu dans σ , pour des couples (i, r) valant itérativement $(0, 1)$, $(1, 1)$, $(1, 2)$, $(3, 1)$,

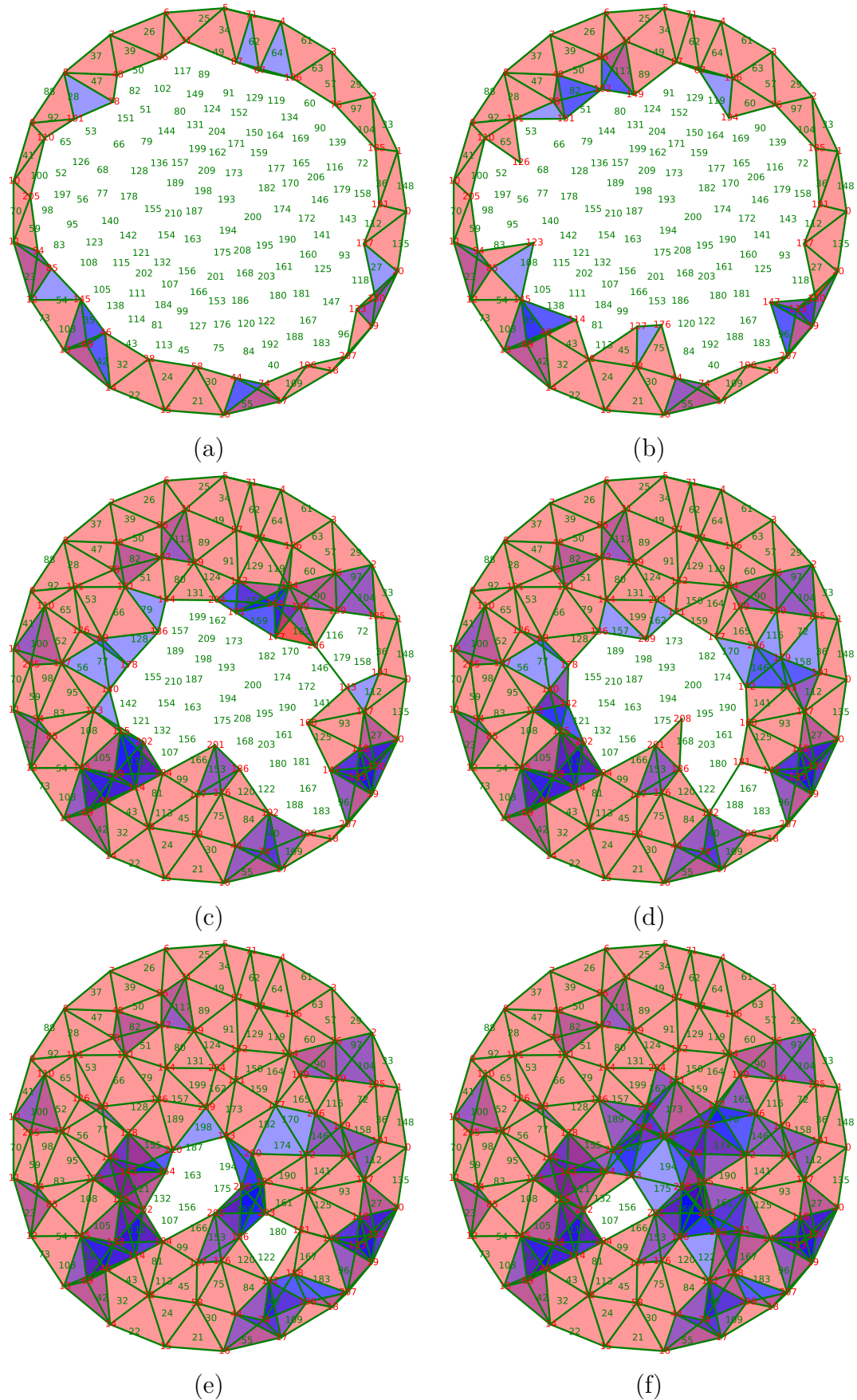


FIGURE 4.18 – Construction de la couverture par la première méthode

Algorithme 4.3 : Algorithme de Floyd pour obtenir l'orbite périodique de Φ

Input : Un complexe cellulaire X et sa bordure F
Input : Un champ de vecteurs discrets V
Output : Liste des sommets participant à la couverture
Data : σ et τ , deux chaînes de $C_2(X)$

```

1  $\sigma \leftarrow VF$ 
2  $\tau \leftarrow VF$ 
3 repeat
4    $\tau \leftarrow \tau + V\partial\tau + \partial V\tau + VF$ 
5    $\sigma \leftarrow \sigma + V\partial\sigma + \partial V\sigma + VF$ 
6    $\sigma \leftarrow \sigma + V\partial\sigma + \partial V\sigma + VF$ 
7 until  $\sigma = \tau$ 
8 return liste des sommets des simplexes de  $\sigma$ 

```

Algorithme 4.4 : Algorithme de Brent pour obtenir l'orbite périodique de Φ

Input : Un complexe cellulaire X et sa bordure F
Input : Un champ de vecteurs discrets V
Output : Liste des sommets participant à la couverture
Data : σ et τ , deux chaînes de $C_2(X)$
Data : i, r , deux entiers positifs

```

1  $(i, r) \leftarrow (0, 1)$ 
2  $\tau \leftarrow VF$ 
3  $\sigma \leftarrow \tau + V\partial\tau + \partial V\tau + VF$ 
4 while  $\sigma \neq \tau$  do
5   if  $r \leq i$  then
6      $r \leftarrow r + 1$ 
7      $\sigma \leftarrow \sigma + V\partial\sigma + \partial V\sigma + VF$ 
8   else
9      $(i, r) \leftarrow (i + 1, 0)$ 
10     $\tau \leftarrow \sigma$ 
11     $\sigma \leftarrow \tau + V\partial\tau + \partial V\tau + VF$ 
12  end
13 end
14 return liste des sommets des simplexes de  $\sigma$ 

```

(3, 2), (3, 3), (3, 4), etc. Le nombre de calcul de Φx est réduit par cet algorithme par rapport à celui de Floyd.

L'algorithme 4.2 pose de sérieux problèmes pour la mise en œuvre distribuée car il demande une grande synchronisation des capteurs. En effet, les lignes 4 et 5 sont simples à calculer localement car elles ne font intervenir que des données distribuées localement : le champ V et les faces des simplexes. Mais la condition de la boucle ou encore la synchronisation des itérations demandent une approche centralisée.

4.4.3.2 Seconde méthode

Suivre le flot se justifie par l'application directe de la théorie de Morse discrète au calcul des générateurs des groupes d'homologie. Toutefois, la particularité de notre application permet de s'en détacher.

En effet, nous ne cherchons pas une chaîne qui représente le générateur de $H_2(C_\star, F)$ mais les sommets d'une telle chaîne. Ce relâchement de contrainte permet de simplifier le calcul des capteurs à activer tout en garantissant la couverture.

Par exemple, nous remarquons sur la figure 4.18 que la couverture se construit petit à petit et que le bord de la chaîne σ passe par tout les capteurs que seront sélectionnés pour assurer la couverture. Il est par conséquent envisageable de suivre le flot comme pour la première méthode mais dans lequel seul le bord est réellement suivi.

Pour que cette solution soit viable, il reste à montrer que prendre le flot puis le bord est équivalent à prendre le flot du bord. Mathématiquement, nous avons d'une part

$$\partial\Phi\sigma = \partial(\sigma + \partial V\sigma + V\partial\sigma) = \partial\sigma + \partial^2 V\sigma + \partial V\partial\sigma = \partial\sigma + \partial V\partial\sigma, \quad (4.22)$$

car $\partial^2 = 0$ par construction des complexes et d'autre part, pour la même raison,

$$\Phi\partial\sigma = \partial\sigma + \partial V(\partial\sigma) + V\partial(\partial\sigma) = \partial\sigma + \partial V\partial\sigma + \partial^2 V\sigma = \partial\sigma + \partial V\partial\sigma = \partial\Phi\sigma. \quad (4.23)$$

Par conséquent l'opérateur de bord et l'opérateur de flot commutent, $\partial\Phi = \Phi\partial$, et donc leur itérés aussi $\partial\Phi^k = \Phi^k\partial$. La seconde méthode suit donc le bord de $\Phi^k\pi$ c'est-à-dire $\partial\Phi^k\pi$ qui est exactement équivalent à suivre le bord de $\Phi^k\partial\pi = \Phi^k F$.

L'algorithme 4.5 reprend donc cette dernière relation sous forme de pseudo-code. La condition d'arrêt de l'algorithme est elle aussi modifiée. Comme l'invariant $\Phi^\infty\pi$ est un générateur de $H_2(C_\star, F)$, c'est-à-dire qu'il est au minimum un cycle dont le bord est par définition vide. Il suffit par conséquent de s'arrêter dès que le bord est vide, d'où la condition d'arrêt de la ligne 6.

Calculer le bord de $\Phi^k\pi$ n'est pas utile si l'ensemble des sommets qui y participent n'est pas mémorisé. Cette mémorisation est l'objet de la variable `actifs`.

La figure 4.19 illustre le balayage du réseau de capteurs par la seconde méthode à différentes itérations. Tout comme la figure 4.18, les capteurs rouges sont activés, ceux en

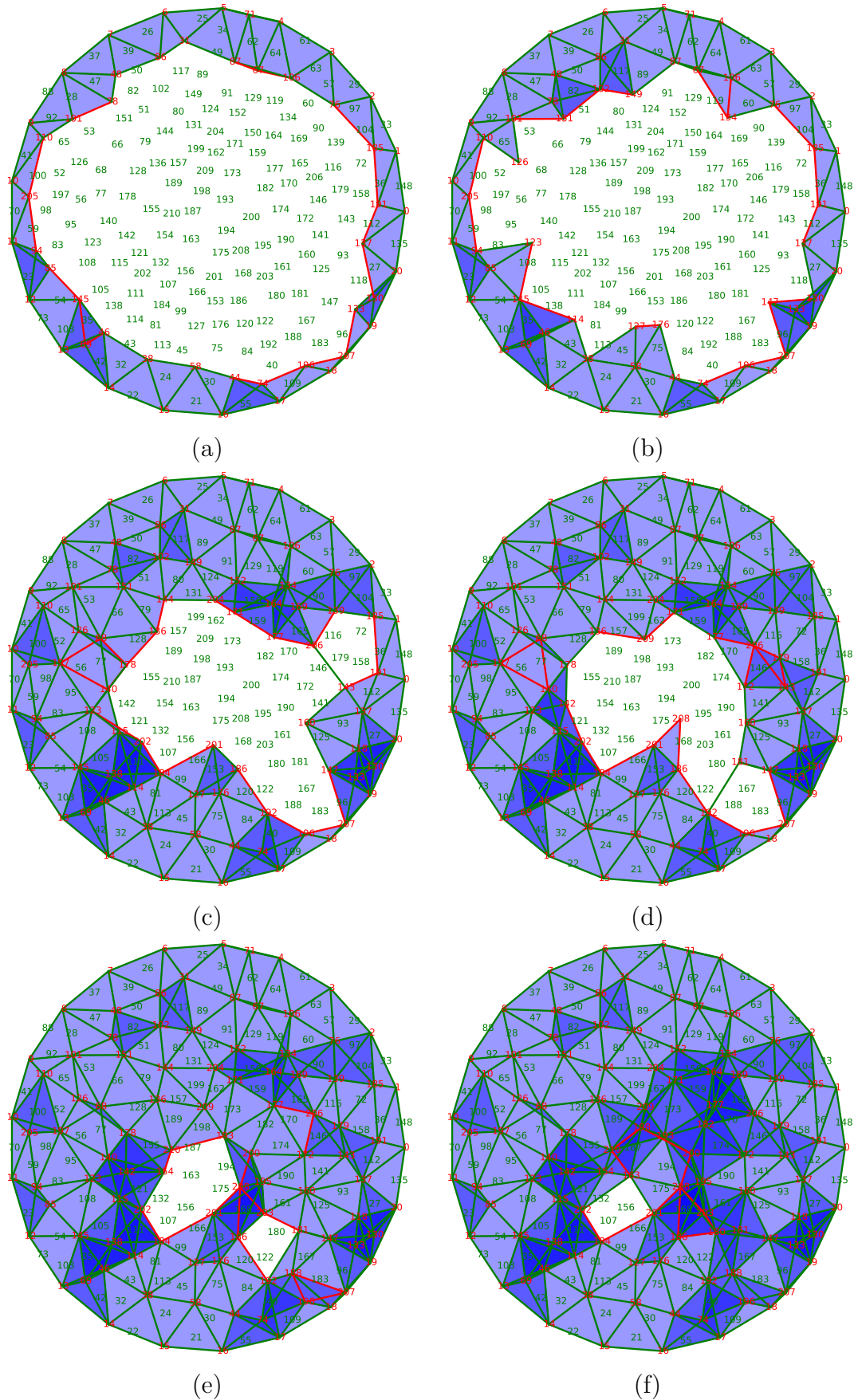


FIGURE 4.19 – Construction de la couverture par la seconde méthode

Algorithme 4.5 : Seconde méthode pour calculer les capteurs activés

Input : Un complexe cellulaire X et sa bordure F
Input : Un champ de vecteurs discrets V
Output : Liste des sommets participant à la couverture
Data : σ , une chaîne de $C_2(X)$
Data : **actifs** un ensemble de sommets

```

1  $\sigma \leftarrow F$ 
2 actifs  $\leftarrow$  sommets de  $F$ 
3 repeat
4    $\sigma \leftarrow \sigma + \partial V \sigma$ 
5   actifs  $\leftarrow$  actifs  $\cup$  sommets de  $\sigma$ 
6 until  $\sigma \neq \mathbf{0}$ 
7 return liste des sommets actifs

```

vert sont mis en sommeil, les triangles bleus représentent la couverture assurée à chaque itération. Les arêtes rouges donne le contenu de la variable σ à ces instants.

Les itérations — numéros 2, 3, 6, 7, 9 et 11 (a–f) — sont choisis pour correspondre à ceux de la figure 4.18. Nous remarquons bien que la seconde méthode suit la bordure de $\Phi^k \pi$ calculé par la première méthode. L'objectif est donc bien atteint.

Le grand avantage de cette seconde méthode est qu'elle est très simplement distribuable. En effet le calcul de la ligne 4 est purement local. D'autre part l'ensemble **actifs** peut être mis en œuvre avec un simple booléen par capteur qui indique s'il est activé ou non. L'algorithme commence donc avec les arêtes du bord F qui se propage de capteur en capteur pour calculer 4. Les capteurs qui « voient passer » ce calcul deviennent *de facto* actifs. Il s'agit bien d'un balayage du réseau qui peut, de plus, être calculé de façon complètement asynchrone. Par conséquent, la partie protocolaire de cet algorithme est très réduit.

4.4.3.3 Comparaison des deux méthodes

Les deux méthodes précédentes pour choisir l'activation ou non de chaque capteurs sont intuitivement différentes car elles reposent pour l'une sur la construction d'une chaîne de dimension 2 invariante par le flot et pour l'autre sur le suivi d'une chaîne de dimension 1 modifiée directement par les vecteurs du champ de vecteurs discrets.

Pour autant, dans notre simulation, les deux méthodes donnent quasiment le même résultat. La figure 4.20 affiche le résultat final des deux méthodes. Les capteurs activés sont en rouge et les autres en vert.

Les deux complexes sont identiques sauf au niveau du capteur numéro 164. En effet la seconde méthode l'active alors que la première le met en sommeil. Ainsi la première méthode ne conserve que 101 capteurs sur les 211 pour assurer la couverture du domaine

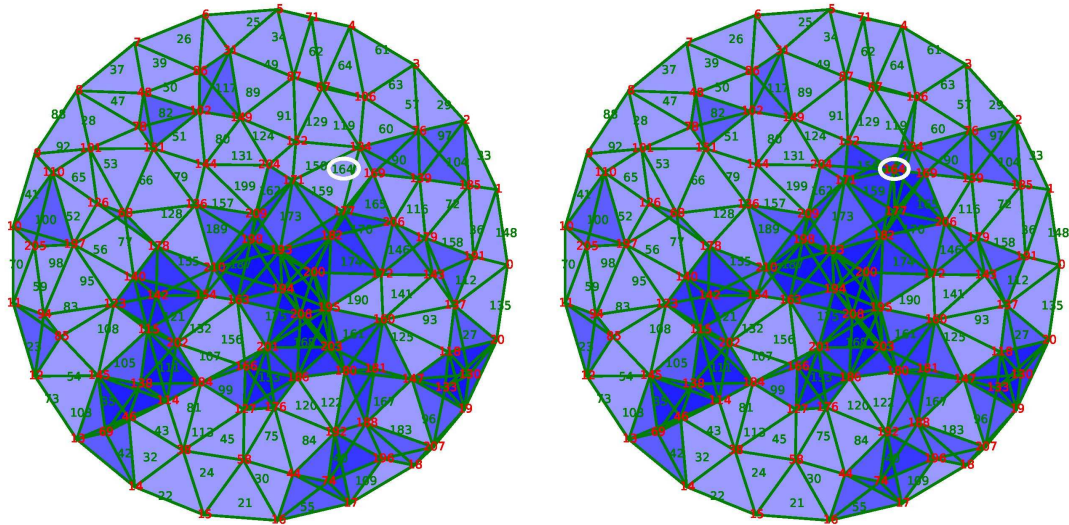


FIGURE 4.20 – Configuration finale des capteurs selon les deux méthodes

alors que la seconde méthode garde les mêmes capteurs plus le numéro 164, situé au premier quartier du réseau.

La différence est minime mais significative de la différence de traitement entre ces deux méthodes. En effet la première calcule sur \mathbb{F}_2 les chaînes pour récupérer finalement les capteurs, ce qui correspond à des opérations de différences symétriques au niveau ensembliste alors que la seconde méthode utilise des unions ; une fois activé par cette méthode, un capteur ne sera pas désactivé.

Cependant, la seconde méthode passe beaucoup mieux à l'échelle car elle correspond simplement à un balayage du réseau de capteurs. Cette opération est par conséquent purement locale et aucune mémoire supplémentaire est nécessaire à sa réalisation.

La première méthode, quant à elle, doit chercher une chaîne invariante du flot. Tant que celle-ci n'est pas trouvée, le réseau doit retenir l'ensemble de la chaîne. Un protocole particulier doit être mis en place pour s'assurer de la convergence de ce calcul qui permet ensuite de fixer l'état d'activation des capteurs et la libération de la mémoire des capteurs.

Les inconvénients de la première méthode nous feront préférer la seconde dans la suite de cette section et dans les simulations.

4.4.4 Suivi de la couverture et activation automatique des capteurs

Les capteurs intérieurs ont un stock d'énergie initial aléatoirement et indépendamment distribués selon une loi exponentielle de moyenne 100. Cette distribution est choisie pour modéliser une certaine hétérogénéité des batteries et de l'utilisation des capteurs. Les

capteurs sur la frontière F ont par contre un stock infini. Cette spécificité permet de garder les capteurs de la frontière toujours présents dans le réseau de capteurs.

La simulation est gérée par événement : d'une itération à l'autre, un capteur a complètement épuisé son stock d'énergie. Ainsi une itération peut représenter une durée plus ou moins longue dans la vie du réseau. Ce type de simulation est naturel car tant qu'un capteur éveillé n'a pas épuisé son stock d'énergie, aucune modification n'a lieu sur le complexe de la couverture et donc le champ de vecteur reste aussi constant.

La figure 4.21 représente l'état du réseau entre différentes étapes de mise à jour de l'algorithme : entre les étapes 0 et 1 (a–b), 4 et 5 (c–d), 20 et 21 (e–f). Les capteurs actifs sont numérotés en rouge, ceux mis en veille en vert et le noir correspond aux capteurs « morts ». Les arêtes représentent les connections qui sont entre deux nœuds éveillés et les triangles bleus indique la zone couverte qui doit être tout le disque de rayon 1.

Cet exemple montre la grande stabilité de notre système. Seuls les nœuds nécessaires à la couverture sont réveillés et ceci de façon très localisé. En effet, notre algorithme active 102 capteurs sur les 211 possibles pour garantir la couverture soit moins de la moitié. Mais surtout lorsqu'un capteur disparaît par manque d'énergie, l'algorithme n'en réveille qu'un autre la plupart du temps.

La couverture n'est pas très redondante comme le montre la vignette (a). Lorsqu'un capteur disparaît par manque d'énergie, l'algorithme n'en réveille qu'un autre la plupart du temps.

Par exemple, comme nous le voyons sur les vignettes (a–b) de la figure 4.21 la disparition du premier capteur, le numéro 186, entouré sur la vignette (a), ne réveille que le capteur numéro 120, lui aussi entouré sur la seconde vignette (b), qui suffit pour reprendre en charge localement la couverture. Il en est de même lors de la disparition du capteur numéro 113 (c–d), qui n'engendre que le réveil du capteur 81. Par contre le réveil de ce dernier densifie la couverture autour du capteur 184 comme l'indique la teinte bleue plus foncée. Dans cet exemple les capteurs 184 et 81 sont suffisamment proches pour être redondants.

Les vignettes (e–f) de la figure 4.21 illustrent la rupture de la couverture causée par la disparition du capteur 123, à gauche de la zone découverte, qui n'est pas résorbée par le réveil d'autres capteurs. La disparition du capteur 123 provoque la création d'un trou de couverture qui ne peut être résorbé avec notre algorithme simple par le réveil d'un autre capteur. Il semble toutefois que le réveil des nœuds 115, 121 et 142 puissent réparer le trou de couverture créé par la suppression du nœud 123. Notre algorithme peut donc être amélioré pour résoudre ce problème. La section suivante présentera deux directions de développement pour corriger cela.

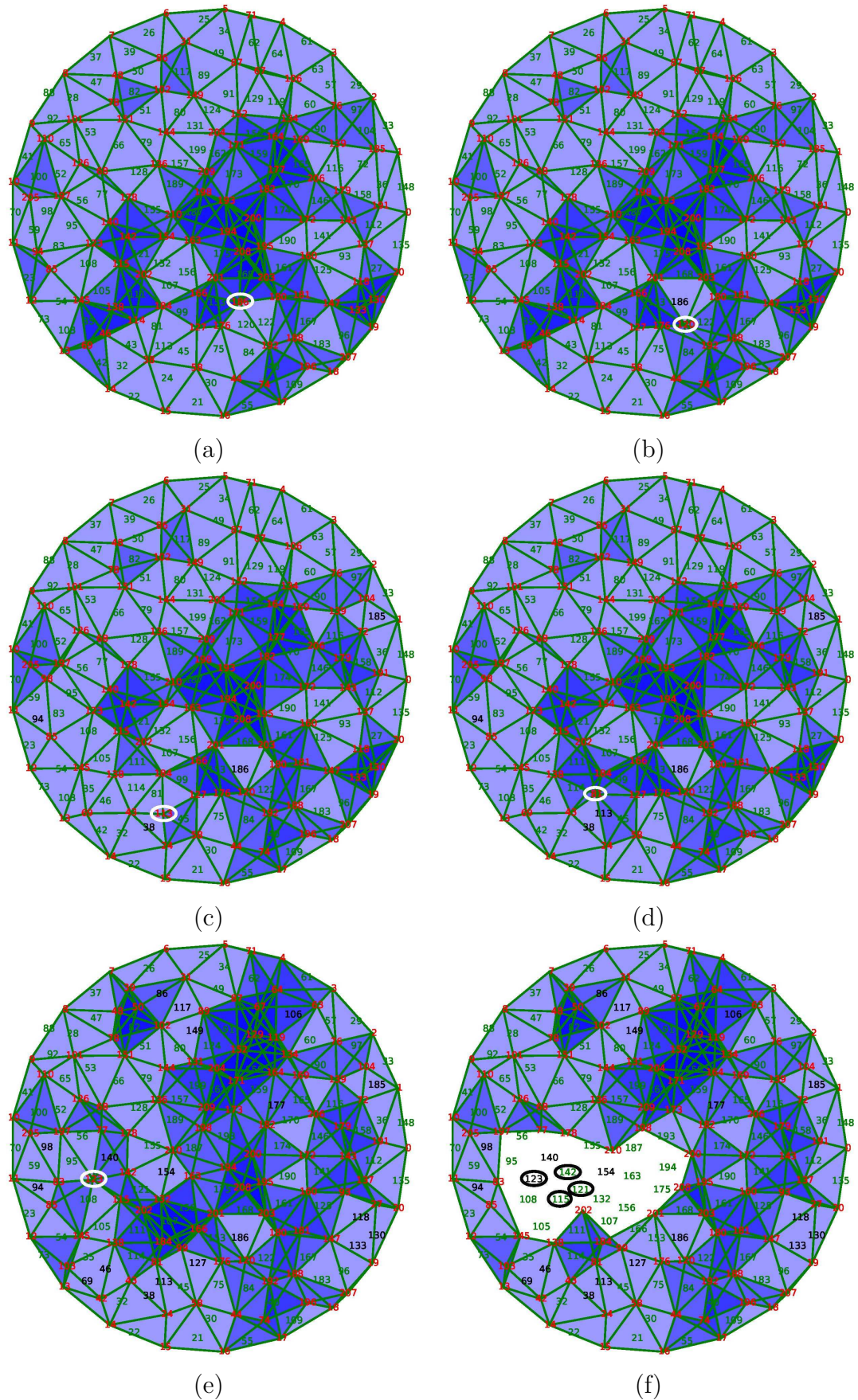


FIGURE 4.21 – Instantanés des capteurs éteints et éveillés

4.4.5 Améliorations de l'algorithme

La difficulté entrevue à la fin de vie du réseau de l'exemple précédent est due à la présence des simplexes $(83, 145)$, $(77, 128)$ et $(83, 95, 108)$ devenus critiques suite à la disparition du nœud 123.

En effet, à cause de la non optimalité du complexe de Morse provenant du champ de vecteurs discrets, notre méthode utilisée pour calculer le générateur du groupe $H_2(C_*, F)$ qui suit le flot de la bordure selon le champ de vecteurs stoppe sa progression lorsqu'elle rencontre une cellule critique.

Il est possible de remédier à cela de deux façons différentes décrites ci-dessous.

1. La première utilise le champ de vecteur pour construire le complexe de Morse associé M pour en calculer le groupe d'homologie $H_2(M, F)$. Le générateur désiré de $H_2(C_*, F)$ est alors construit en ramenant le générateur de $H_2(M, F)$ dans le complexe C_* .

Cette méthode n'est pas spécialement coûteuse car le complexe de Morse est très simple relativement au complexe initial C_* . Dans notre simulation, par exemple, il contient moins d'une dizaine de cellules critiques.

Cependant, le calcul du groupe d'homologie nécessite le rapatriement du complexe de Morse en un nœud central pour le calcul de son homologie. Cet aspect rend la méthode rétrograde, et nous n'en ferons pas usage.

2. La seconde méthode est complètement distribuée et utilise l'annulation de paire de cellules critiques dans le complexe de Morse telle qu'expliquée à la section 4.1.8. En effet, il est souvent possible de simplifier le complexe de Morse en retournant les vecteurs d'un chemin entre deux cellules critiques.

Plus précisément, pour chaque cellule critique σ , tous les chemins débutant par une face de σ et qui suivent le champs de vecteurs discret doivent déboucher sur une autre cellule critique τ . S'il n'y a qu'un unique chemin entre σ et τ alors en renversant le sens des vecteurs sur le parcours de ce chemin, les deux cellules sont appariées et ne sont donc plus critique. Le complexe de Morse est donc simplifié car le nombre de cellules critiques est diminuées de 2 sans modifier les groupes d'homologie.

L'unicité du parcours est important car elle assure que le champ de vecteurs discrets modifié ne comporte pas de cycle, c'est à dire d'un chemin selon le champ qui puisse revenir au point de départ. Ce type de chemin modifie l'homologie si la construction du complexe de Morse reste simple comme expliqué à la section 4.1.5.

Même si cette seconde méthode peut paraître plus compliquée que la première, elle est très facile à mettre en œuvre. En effet la recherche des chemins entre les cellules critiques peut se faire par un parcours ou une exploration dans un graphe [CLR09]. Ce type d'algorithme est à la base de beaucoup d'algorithme de routage ; il faut simplement les adapter pour qu'ils utilisent le complexe C_* à la place des liens de connexion. Par conséquent, ces algorithmes sont facilement distribuables dans le réseau de capteurs sans fils car le complexe simplicial et le champs de vecteur

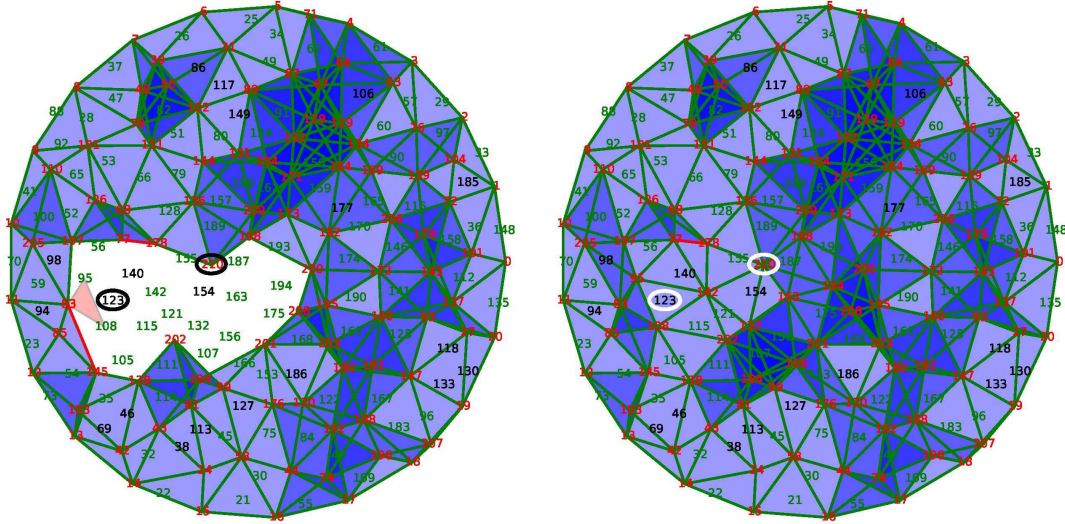


FIGURE 4.22 – Comparaison entre l’algorithme de base, à gauche, et la version corrigée par la simplification, à droite

discret sont gérés localement et le suivi des chemins selon champ de vecteurs se fait en faisant transiter un message entre les capteurs qui appartiennent au chemin du complexe de Morse.

Nous avons mis en place la seconde méthode qui simplifie le complexe de Morse. La différence entre l’algorithme avec cette amélioration et l’algorithme sans cette dernière. Le résultat est visible sur la figure 4.23 à l’instant où l’algorithme sans modification ne peut plus garantir la couverture.

Sans la simplification du complexe de Morse, la suppression du capteur 123 implique la mise à jour du champ de vecteurs discrets. Toutefois cette dernière n’est pas optimale : le nœud 210, les liens (83, 145), (77, 128) ainsi que le triangle (83, 95, 108) deviennent critiques. Ces cellules sont indiquées en rouge à gauche de la figure 4.22.

Avec la méthode de la simplification, la disparition du capteur 123 ne rompt pas la couverture car le nouveau champ de vecteurs discrets est modifié pour supprimer ces cellules critiques. En effet un unique chemin entre les cellules (83, 95, 108) et (83, 145) est trouvé par le réseau de capteur et le champ le long de ce chemin est inversé. Il en est de même entre le cellule (77, 128) et le nœud 210.

Une fois la simplification faite, le calcul du générateur de $H_2(C_*, F)$ reprend pour fournir la nouvelle configuration donnée à droite de la figure 4.22.

On remarque encore une fois la stabilité du processus car les modifications restent très localisées. Cette amélioration de l’algorithme permet de rallonger la durée de vie du réseau de capteurs sans fil en garantissant la couverture.

Naturellement, d’autres améliorations sont envisageables notamment pour la construction du champ de vecteur discrets initial qui permettrait d’optimiser au maximum le nombre

de capteurs endormis.

Par exemple, dans la simulation, 102 capteurs sont initialement réveillés sur un total de 211 capteurs. Mais certains capteurs pourraient être mis en sommeil sans difficulté comme le capteur 142 par exemple. Le générateur trouvé par le calcul direct de l'homologie présenté à la section 3.1.5.2 demande quant à lui 106 capteurs réveillés pour le même réseau.

Il paraît difficile toutefois de faire cette optimisation de manière décentralisé et sans recourir aux solutions lourdes présentées à la section 3.3.6.2. Il semble toutefois possible de gagner quelques unités à la marge par des algorithmes gloutons locaux.

4.4.6 Gestion de la couverture sans garantie

Que se passe-t-il lorsque notre algorithme de gestion de l'énergie continue de fonctionner même si la couverture n'est plus garantie. Dans ce cas, le groupe $H_2(C_\star, F)$ devient nul. Par contre, notre méthode pour calculer son générateur peut encore être fonctionnel. Elle cherche en effet à réduire en suivant le champ de vecteurs discrets la chaîne qui correspond à la frontière. Donc même s'il y a un trou de couverture, l'algorithme cherche à mettre en veille les capteurs non nécessaires en partant de la frontière.

La figure 4.23 montre le déroulement de l'algorithme après la rupture de la couverture juste après la création de trous. Ces trous sont bien détectés et correspondent à de vrais trous et non à un artefact de notre méthode tel que celui avant la simplification du complexe de Morse.

Les trous de couverture peuvent être entourés en suivant le flot issu du bord des cellules critiques de dimension 1. Naturellement les cycles ainsi construit ne sont pas optimaux mais donne une indication de la localisation des trous.

Pour finalement savoir si des trous sont présents ou non, il suffit de calculer la caractéristique d'Euler-Poincaré du complexe de Morse ce qui se fait très simplement en comptant le nombre de cellules critiques,

$$\chi(C_\star) = \chi(M) = \sum_{\sigma \in \text{cellules critiques}} (-1)^{\dim \sigma}, \quad (4.24)$$

le nombre de trous étant alors donné par $1 - \chi(M)$.

4.5 Conclusion

La théorie de Morse nous a permis dans ce chapitre de développer un algorithme de calcul de groupes d'homologie efficace et totalement distribué. Sa forme est suffisamment souple et générique pour qu'il soit utilisable dans la détection des trous de couverture uniquement grâce aux calculs des corrélations entre les mesures de capteurs.

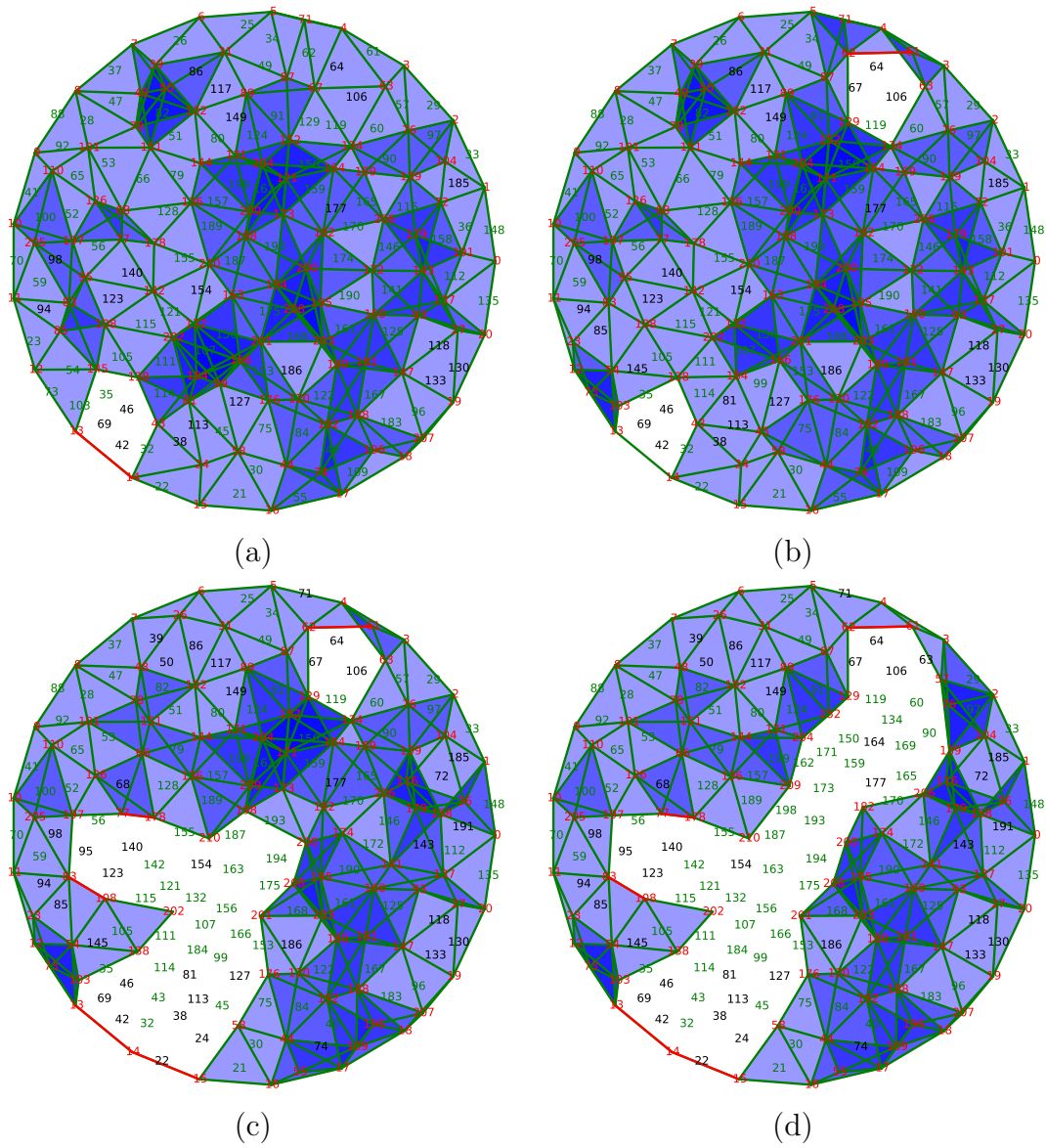


FIGURE 4.23 – Instantanés des capteurs éteints et éveillés sans couverture garantie

Mais notre algorithme peut aussi être mis en œuvre dans le cadre de la sélection de capteurs actifs. Pour cette application, une version asynchrone est proposée qui s'affranchit de la nécessité d'initier simultanément et périodiquement une phase de mise à jour.

Bien sûr, notre algorithme est utilisable dans d'autres contextes comme pour la sélection des fréquences porteuses pour éviter les interférences [VDM13b] ou pour tout autres problèmes solvables par les groupes d'homologie [Rob13a].

Chapitre 5

Conclusion et perspectives

5.1 Conclusion générale	111
5.2 Perspectives	112

5.1 Conclusion générale

L'application de la topologie et notamment de la topologie algébrique à des problèmes d'ingénierie est actuellement en plein développement. Cette théorie mathématique est particulièrement vaste et son applicabilité tient surtout à la possibilité de rendre une partie calculable et représentable sur un système réel. Par exemple, dans nos travaux sur les problèmes de couverture dans les réseaux de capteurs, l'homologie simpliciale et cellulaire et la théorie de Morse discrète se sont montrées particulièrement efficaces.

En effet grâce à ces deux outils mathématiques, nous avons proposé un algorithme qui détecte et localise les trous de couverture sur la régression d'un champ de mesure scalaire puis la distribution de cet algorithme permet ensuite de suivre les évolutions du réseau de capteurs. Un à-côté particulièrement intéressant de ces algorithmes et qu'ils sélectionnent une partie des capteurs nécessaire à la couverture. les autres capteurs non sélectionnés peuvent alors être mis en veille pour sauvegarder leur énergie et ainsi augmenter la durée de vie du réseau ; leur réveil sera dicté par l'évolution des unités du réseau.

La réussite de ces approches topologiques tient à la souplesse même des espaces manipulés. En effet, une simple notion de voisinage est requise. Celle-ci se transforme en une liste de cellules et de relation de collage entre elles grâce à l'opérateur de bord. Un fois ces données fournies, il est possible d'en calculer plusieurs invariants mesurant diverses quantités qui ne sont pas nécessairement scalaires comme les groupes d'homologie.

Les quantités issues de la topologie décrivent la forme de l'espace et en donnent une représentation qualitative. Si cet aspect qualitatif semble très faible par rapport à une

série quantitative de valeurs, pour permettre de déduire des faits, il n'en est rien. Ce qui semble perdu est largement compensé par la flexibilité de l'approche. Grâce à celle-ci, l'homologie permet de détecter des trous de couverture d'un réseau de capteurs sans en connaître la position des unités et de sélectionner un nombre limité tout en garantissant la couverture.

La difficulté de l'utilisation de la topologie algébrique se concentre maintenant non sur les outils mais sur la spatialisation du problème et l'interprétation des invariants. Par exemple, dans le chapitre 3, l'utilisation des corrélations entre les mesures des capteurs a permis la construction d'un espace, plus précisément un complexe simplicial, dont le premier groupe d'homologie détecte la présence de trou de couverture ou plutôt d'un point de la zone de couverture dont l'estimation du champ manque de précision. La mise sous forme d'un complexe simplicial n'est pas immédiate et relier le groupe d'homologie à la couverture demande un certain travail. Mais le gain est important, la seule hypothèse est que le champ de mesure suit un processus gaussien dont le noyau décroît avec la distance. Aucune hypothèse sur la position des capteurs ou autre n'est nécessaire.

La topologie n'est pas tant « géométrie du caoutchouc, » comme elle est souvent appelée, que l'outil mathématique permettant le passage du local au global, c'est-à-dire le passage d'une notion simple et locale de proximité à une notion complexe et plus générale de forme globale de l'espace. Les outils qui permettent cette intégration du local vers le global sont performants et peuvent parfois donner naissance à des algorithmes de calculs distribués. Par exemple, la théorie de Morse nous a permis au chapitre 4 de construire un algorithme distribué de calcul d'homologie et de pouvoir l'utiliser aussi bien pour des problèmes de couverture que pour la gestion adaptative de l'énergie par sélection active des capteurs éveillés.

5.2 Perspectives

Les perspectives que nos travaux ouvrent sont nombreuses, tant les applications de la topologie algébrique commencent à peine à apparaître. Nous pouvons faire rapidement le tour de celles-ci en les séparant entre les perspectives à court terme et celles à long terme.

Dans un premier temps, il serait intéressant d'améliorer la spatialisation du problème de couverture du chapitre 3 par les corrélations. Par exemple, au lieu de considérer une borne sur la précision de l'estimation, nous pourrions envisager d'utiliser des mesures issues de la théorie de l'information comme l'information mutuelle entre la valeur du champ en certain point et les mesures issues des capteurs voisins. Une autre voie serait d'éviter de considérer un champ suivant un processus gaussien mais d'être plus générique et choisir le type de champ selon le principe d'entropie maximale cher à Jaynes [Jay82].

Plus simplement, le complexe construit sur la corrélation peut être affiné si plusieurs capteurs sont ajoutés dans les contraintes de la programmation quadratique. Nous avons toutefois tenté cette approche sans réel succès immédiat mais une approche plus construite

sera sûrement bénéfique. De même, une optimisation des cycles selon leur longueur n'a pas été mise en œuvre, ce qui et serait profitable pour localiser précisément les trous ou alors pour optimiser le nombre de capteurs actifs sélectionnés lors de la gestion de l'énergie.

La topologie algébrique ne se cantonne pas à l'homologie. Ainsi, à plus long terme, il serait intéressant de développer d'autres parties. Par exemple la cohomologie, bien que duale de l'homologie, étudie un espace par l'étude de la cohérence locale des fonctions définies sur ce dernier. Les incohérences sont donc très informatives. Par exemple, les singularités d'un champ mesuré par des capteurs peuvent certainement être détectées grâce à la cohomologie. Afin, grâce à la cohomologie, il est possible de plaquer des coordonnées sur des espaces sans notion de métrique mais uniquement par celle de proximité [dSMVJ11b].

Plus généraux encore, les faisceaux représentent les données définies localement sur un espace et le collage entre les données à proximité. La théorie des faisceaux est donc tout indiquée pour les réseaux de capteurs car les données locales qui ne sont pas nécessairement de la même forme doivent être agrégées de proche en proche afin de construire une valeur globale pour l'opérateur. Les faisceaux, bien que d'un abord difficile, sont très puissants pour le traitement du signal [Rob14b], pour le codage de réseaux [GH11], pour l'échantillonnage [Rob14a] ou autre [Ghr14].

Enfin, la topologie différentielle extérieure et surtout le calcul extérieur discret [DHL05, CdGD13] sont attrayants pour les réseaux de capteurs car ils autorisent le calcul de l'homologie par des voies détournées [ME06, TSJ10] mais ils permettent aussi d'autres applications comme le suivi de cible ou l'agrégation de requêtes dans un réseau de capteurs [SG13].

Nous avons par exemple travaillé sur une utilisation du calcul extérieur discret pour un réseau de capteurs qui est uniquement basé sur des jetons voyageant dans le réseau et accumulant la valeur associée aux arêtes du graphe de connexion. Lorsque deux jetons se rejoignent et dont les trajectoires concaténées forment une boucle, alors la comparaison de leur valeur accumulée donne des indications sur l'intérieur de la boucle à l'instar du théorème de Green que relie une intégrale sur la frontière d'un domaine et l'intégration sur la surface. Par exemple, comme le savent les arpenteurs, l'aire d'une surface peut se calculer à partir de son pourtour [BLL05]. Ces travaux n'ont pas encore aboutis mais cette approche reste très intéressante et générale grâce à leurs généralisation par le théorème de Stokes dont le théorème de Green n'est qu'un corollaire.

Bibliographie

- [AAC11] Henry Adams, Atanas Atanasov, and Gunnar Carlsson. Morse theory in topological data analysis. arxiv, December 2011, 1112.1993.
- [AB12] A. Abdelgawad and M. Bayoumi. *Resource-Aware Data Fusion Algorithms for Wireless Sensor Networks*. Springer, 2012.
- [Abr97] Petter Abrahamsen. A review of gaussian random fields and correlation functions. Technical report, Norwegian Computing Center, 1997.
- [AC07] M. Allili and D. Corriveau. Topological analysis of shapes using morse theory. *Computer Vision and Image Understanding*, 105(3) :188–199, 2007.
- [Ada13] Henry Adams. *Evasion Paths in Mobile Sensor Networks*. PhD thesis, Stanford University, August 2013.
- [AF07] Colin Adams and Robert Franzosa. *Introduction to Topology : Pure and Applied*. Prentice Hall, 2007.
- [AKL13] Madjid Allili, Tomasz Kaczynski, and Claudia Landi. Reducing complexes in multidimensional persistent homology theory. arxiv, October 2013, 1310.8089.
- [Arm83] M. A. Armstrong. *Basic topology*. Springer, 1983.
- [ASS02] I.F. Akyildiz, Weilian Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Commun. Mag.*, 40(8) :102–114, August 2002.
- [AT07] Robert J. Adler and Jonathan E. Taylor. *Random fields and geometry*. Monographs in Mathematics. Springer, 2007.
- [AUG] AUG. Alert : Automated local evaluation in real time. <http://www.alertsystems.org/>.
- [Bas06] William F. Basener. *Topology and Its Applications*. Wiley-Blackwell, 2006.
- [Bau11] Ulrich Bauer. *Persistence in discrete Morse theory*. PhD thesis, Georg-August-Universität Göttingen, 2011.
- [BDM13] Jean-Daniel Boissonnat, Tamal K. Dey, and Clément Maria. The compressed annotation matrix : an efficient data structure for computing persistent cohomology. Technical Report 8195, INRIA Sophia Antipolis, April 2013, 1304.6813.

- [BFL⁺13] Sivaraman Balakrishnan, Brittany Fasy, Fabrizio Lecci, Alessandro Rinaldo, Aarti Singh, and Larry Wasserman. Statistical inference for persistent homology. *pre-print*, March 2013, 1303.7117.
- [BG09] Yury Baryshnikov and Robert Ghrist. Target enumeration via Euler characteristic integrals. *SIAM Journal on Applied Mathematics*, 70(3) :825–844, 2009.
- [BG10] Yuliy Baryshnikov and Robert Ghrist. Euler integration over definable functions. *Proc. National Acad. Sci.*, 107(21) :9525–9530, May 2010.
- [BGS00] P. Bonnet, J. Gehrke, and P. Seshadri. Querying the physical world. *Personal Communications, IEEE*, 7(5) :10–15, October 2000.
- [BHK⁺06] Tatiana Bokareva, Wen Hu, Salil Kanhere, Branko Ristic, Neil Gordon, Travis Bessell, Mark Rutten, and Sanjay Jha. Wireless sensor networks for battlefield surveillance. *Proc. of the Land Warfare Conference*, 2006.
- [BKR14] Ulrich Bauer, Michael Kerber, and Jan Reininghaus. Clear and compress : Computing persistent homology in chunks. In Peer-Timo Bremer, Ingrid Hotz, Valerio Pascucci, and Ronald Peikert, editors, *Topological Methods in Data Analysis and Visualization III*, Mathematics and Visualization, pages 103–117. Springer International Publishing, 2014.
- [BKX⁺06] Xiaole Bai, Santosh Kumar, Dong Xuan, Ziqiu Yun, and Ten H. Lai. Deploying wireless sensors to achieve both coverage and connectivity. *Mobi-Hoc'06*, 2006.
- [BL14] Bruno Benedetti and Frank H. Lutz. Random discrete morse theory and a new library of triangulations. *Experimental Mathematics*, 23(1) :66–94, 2014.
- [BLL05] S. Brlek, G. Labelle, and A. Lacasse. The discrete green theorem and some applications in discrete geometry. *Theoretical Computer Science*, 346(2–3) :200–225, 2005. In memoriam : Alberto Del Lungo (1965-2003).
- [BLW12] Ulrich Bauer, Carsten Lange, and Max Wardetzky. Optimal topological simplification of discrete functions on surfaces. *Discrete & Computational Geometry*, 47 :347–377, March 2012.
- [BM12] Jean-Daniel Boissonnat and Clément Maria. The simplex tree : An efficient data structure for general simplicial complexes. In Leah Epstein and Paolo Ferragina, editors, *Algorithms – ESA 2012*, volume 7501 of *Lecture Notes in Computer Science*, pages 731–742. Springer Berlin Heidelberg, 2012.
- [BMS14] Ulrich Bauer, Axel Munk, Hannes Sieling, and Max Wardetzky. Persistent homology meets statistical inference - a case study : Detecting modes of one-dimensional signals. *pre-print*, April 2014, 1404.1214.
- [Bot88] Raoul Bott. Morse theory indomitable. *Publications mathématiques de l'I.H.É.S.*, 68 :99–114, 1988.
- [Bre80] Richard P. Brent. An improved Monte-Carlo factorization algorithm. *Nordisk Tidskrift for informationsbehandling (BIT)*, 20 :176–180, 1980.

- [Bri07] Robert Bridson. Fast poisson disk sampling in arbitrary dimensions. In *ACM SIGGRAPH*, volume 2007, page 5, San Diego, CA, USA, 2007.
- [BT82] Raoul Bott and Loring W. Tu. *Differential Forms in Algebraic Topology*. GTM. Springer, 1982.
- [BXY⁺08] Xiaole Bai, Dong Xuan, Ziqiu Yun, Ten H. Lai, and Weijia Jia. Complete optimal deployment patterns for full-coverage and k-connectivity wireless sensor networks. *MobiHoc'08*, 2008.
- [BYX⁺08] Xiaole Bai, Ziqiu Yun, Dong Xuan, Ten H. Lai, and Weijia Jia. Deploying four-connectivity and full-coverage wireless sensor networks. *IEEE INFOCOM*, 2008.
- [Car78] Carnegie-Mellon University. *Proceedings of the Distributed Sensor Nets Workshop*, Pittsburgh, PA, 1978.
- [Car09] Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2) :255–308, April 2009.
- [Car14] Gunnar Carlsson. Topological pattern recognition for point cloud data. *Acta Numerica*, 23 :289–368, May 2014.
- [CdGD13] Keenan Crane, Fernando de Goes, Mathieu Desbrun, and Peter Schröder. Digital geometry processing with discrete exterior calculus. In *ACM SIGGRAPH 2013 courses*, SIGGRAPH '13, New York, NY, USA, 2013. ACM.
- [CF10] Chao Chen and Daniel Freedman. Measuring and computing natural generators for homology groups. *Computational Geometry*, 43(2) :169–181, 2010. Special Issue on the 24th European Workshop on Computational Geometry (EuroCG'08).
- [CGN13] Justin Curry, Robert Ghrist, and Vidit Nanda. Discrete morse theory for computing cellular sheaf cohomology. *pre-print*, December 2013, 1312.6454.
- [CGR12] J. Curry, R. Ghrist, and M. Robinson. Euler calculus and its applications to signals and sensing. In *Proceedings of Symposia in Applied Mathematics*, AMS, 2012.
- [Cha] Anantha Chandrakasan. Mit μ amps project website. <http://www-mtl.mit.edu/researchgroups/icsystems/uamps/>.
- [CIS08] Gunnar Carlsson, Tigran Ishkhanov, Vin Silva, and Afra Zomorodian. On the local behavior of spaces of natural images. *Int. J. Comput. Vision*, 76(1) :1–12, January 2008.
- [CK11] Chao Chen and Michael Kerber. Persistent homology computation with a twist. In *In 27th European Workshop on Computational Geometry (EuroCG)*, pages 197–200, Morschach, Switzerland, March 2011.
- [CK13] Harish Chintakunta and Hamid Krim. Distributed computation of homology using harmonics, June 2013, 1306.1158.
- [CKL07] Ai Chen, Santosh Kumar, and Ten H. Lai. Designing localized algorithms for barrier coverage. *MobiCom'07*, 2007.

- [CLR09] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, 3rd edition edition, 2009.
- [Cro05] Martin D. Crossley. *Essential Topology*. Springer, 2005.
- [CS09] Teddy M. Cheng and Andrey V. Savkin. A distributed self-deployment algorithm for the coverage of mobile wireless sensor networks. *IEEE Commun. Lett.*, 13, 2009.
- [CS10] Gunnar Carlsson and Vin Silva. Zigzag persistence. *Foundations of Computational Mathematics*, 10(4) :367–405, 2010.
- [CSEH07] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37 :103–120, 2007.
- [CSZ⁺09] Yebin Chen, Jian Shu, Sheng Zhang, Linlan Liu, and Limin Sun. Data fusion in wireless sensor networks. In *Electronic Commerce and Security, 2009. ISECS '09. Second International Symposium on*, volume 2, pages 504–509, 2009.
- [CTT10] Venkat Chandar, Aslan Tchamkerten, and David Tse. Asynchronous capacity per unit cost. arxiv, July 2010, 1007.4872.
- [Cur13] Justin Curry. Sheaves, cosheaves and applications. *pre-print*, March 2013, 1303.3255.
- [DGJ12] P. Dłotko, R. Ghrist, M. Juda, and M. Mrozek. Distributed computation of coverage in sensor networks by homological methods. *Applicable Algebra in Engineering, Communication and Computing*, 23 :29–58, 2012.
- [DHK11] Tamal K. Dey, Anil N. Hirani, and Bala Krishnamoorthy. Optimal homologous cycles, total unimodularity, and linear programming. *SIAM J. Comput.*, 40(4) :1026–1044, July 2011.
- [DHL05] Mathieu Desbrun, Anil N. Hirani, Melvin Leok, and Jerrold E. Marsden. Discrete exterior calculus. *pre-print*, 2005, math/0508341.
- [Die00] Reinhard Diestel. *Graph Theory*. Graduate Texts in Mathematics. Springer, second edition edition, 2000.
- [DMA14] Roudy Dagher, Nathalie Mitton, and Ibrahim Amadou. Towards wsn-aided navigation for vehicles in smart cities : An application case study. In *1st International IEEE Percom Workshop on Pervasive Systems for Smart Cities*, Budapest, Hungary, March 2014.
- [DP10] Walteneus Dargie and Christian Poellabauer. *Fundamentals of Wireless Sensor Networks : Theory and Practice*. Wiley, 2010.
- [dSG06] V. de Silva and Robert Ghrist. Coordinate-free coverage in sensor networks with controlled boundaries via homology. *International Journal of Robotics Research*, 25(12) :1205–1222, December 2006.
- [dSG07a] V. de Silva and R. Ghrist. Coverage in sensor networks via persistent homology. *Alg. & Geom. Topology*, 7 :339–358, 2007.

- [dSG07b] V. de Silva and R. Ghrist. Homological sensor networks. *Notices of the American Mathematical Society*, 54(1) :10–17, January 2007.
- [dSGM05] Vin de Silva, Robert Ghrist, and Abubakr Muhammad. Blind swarms for coverage in 2-D. In *Proceedings of Robotics : Science and Systems*, Cambridge, USA, June 2005.
- [dSMVj11a] Vin de Silva, Dmitriy Morozov, and Mikael Vejdemo-johansson. Dualities in persistent (co)homology. *Inverse Problems*, 27(12) :124003, December 2011, 1107.5665.
- [dSMVJ11b] Vin de Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. Persistent cohomology and circular coordinates. *Discrete & Computational Geometry*, pages 737–759, 2011.
- [DSW10] Tamal K. Dey, Jian Sun, and Yusu Wang. Approximating loops in a shortest homology basis from point data. In *Proceedings of the Twenty-sixth Annual Symposium on Computational Geometry*, SoCG '10, pages 166–175, New York, NY, USA, 2010. ACM.
- [DTD12] M.P. Durisic, Z. Tafa, G. Dimic, and V. Milutinovic. A survey of military applications of wireless sensor networks. In *Proceedings of Mediterranean Conference on Embedded Computing (MECO 2012)*, pages 196–199, Bar, Montenegro, June 2012.
- [Edm65] Jack Edmonds. Paths, trees, and flowers. *Canad. J. Math.*, 17 :449–467, February 1965.
- [EH09] Herbert Edelsbrunner and John Harer. The persistent morse complex segmentation of a 3-manifold. In Nadia Magnenat-Thalmann, editor, *3D Physiological Human Workshop*, volume 5903 of *Lecture Notes in Computer Science*, pages 36–50, Zermatt, Switzerland, November 2009. Springer.
- [EH10] Herbert Edelsbrunner and John L. Harer. *Computational topology*. American Mathematical Society, 2010.
- [ELZ02] Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. *Discrete Comput Geom*, 28 :511–533, 2002.
- [Eri11] Jeff Erickson. Combinatorial optimization of cycles and bases. In *in Proceedings of Symposia in Applied Mathematics*, AMS Short Course on Computational Topology, 2011.
- [Flo67] Robert W. Floyd. Nondeterministic algorithms. *Journal of the ACM*, 14(4) :636–644, October 1967.
- [FM07] Massimo Franceschetti and Ronald Meester. *Random networks for communication, from statistical physics to information systems*. Cambridge series in statistical and probabilistic mathematics. Cambridge University Press, 2007.
- [For98a] Robin Forman. Combinatorial vector fields and dynamical systems. *Mathematische Zeitschrift*, 228(4) :629–681, 1998.

- [For98b] Robin Forman. Morse theory for cell complexes. *Advances in Mathematics*, 134 :90–145, 1998.
- [For98c] Robin Forman. Witten–Morse theory for cell complexes. *Topology*, 37(5) :945–979, March 1998.
- [For99] Robin Forman. *Combinatorial Differential Topology and Geometry*, volume 38, pages 177–206. MSRI, 1999.
- [For02a] Robin Forman. Discrete morse theory and the cohomology ring. *Trans. Amer. Math. Soc.*, 354(12) :5063–5085, June 2002.
- [For02b] Robin Forman. A user’s guide to discrete morse theory. In *Séminaire Lotharingien de Combinatoire*, volume 48, 2002.
- [For04] Robin Forman. *Six Themes on Variation*, volume 26, chapter How many equilibria are there? An introduction to Morse theory. AMS, 2004.
- [FSR09] Eric Fleury and David Simplot-Ryl. *Réseaux de capteurs : Théorie et modélisation*. Hermes Science Publications, 2009.
- [Gag92] Douglas W. Gage. Command control for many-robot systems. *Unmanned Systems*, 10 :28–34, 1992.
- [GBH00] Steven D. Gribble, Eric A. Brewer, Joseph M. Hellerstein, and David Culler. Scalable, distributed data structures for internet service construction. In *Proceedings of the 4th Conference on Symposium on Operating System Design & Implementation*, pages 22–22, San Diego, California, 2000.
- [GBH08] Attila Gyulassy, Peer-Timo Bremer, Bernd Hamann, and Valerio Pascucci. A practical approach to morse-smale complex computation : Scalability and generality. *IEEE Trans. Vis. Comput. Graphics*, 14(6) :1619–1626, 2008.
- [GCK14a] Jennifer Gamble, Harish Chintakunta, and Hamid Krim. Adaptive tracking of representative cycles in regular and zigzag persistent homology. *pre-print*, November 2014, 1411.5442.
- [GCK14b] Jennifer Gamble, Harish Chintakunta, and Hamid Krim. Coordinate-free quantification of coverage in dynamic sensor networks. *pre-print*, November 2014, 1411.7337.
- [GD08] Amitabha Ghosh and Sajal K. Das. Coverage and connectivity issues in wireless sensor networks : A survey. *Pervasive and Mobile Computing*, 4(3) :303–334, 2008.
- [GGB⁺09] D. Goldsmith, E. Gaura, J. Brusey, J. Shuttleworth, R. Hazelden, and M. Langley. Wireless sensor networks for aerospace applications—thermal monitoring for a gas turbine engine. *Nanotechnology Conference*, 1 :507–512, 2009.
- [GH11] Robert Ghrist and Yasuaki Hiraoka. Applications of sheaf cohomology and exact sequences on network codings. Pre-print, 2011.
- [GH12] Dongbing Gu and Huosheng Hu. Spatial gaussian process regression with mobile sensor networks. *IEEE Transactions on Neural Networks and Learning Systems*, 23(8) :1279–1290, August 2012.

- [Ghr08a] Robert Ghrist. Barcodes : the persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1) :61–75, January 2008.
- [Ghr08b] Robert Ghrist. Three examples of applied and computational homology. *Nieuw Archief voor Wiskunde*, 5/9 :122–125, June 2008.
- [Ghr14] Robert Ghrist. *Elementary Applied Topology*. CreateSpace Independent Publishing Platform, September 2014.
- [Gib10] Peter Giblin. *Graphs, Surfaces and Homology*. Cambridge University Press, 3rd edition, 2010.
- [GK00] P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Trans. Inf. Theory*, 46(2) :388–404, March 2000.
- [GM05] Robert Ghrist and Abubakr Muhammad. Coverage and hole-detection in sensor networks via homology. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, IPSN '05. IEEE Press, 2005.
- [GNB⁺14] Mayank Goswami, Chien-Chun Ni, Xiaomeng Ban, Jie Gao, Xianfeng David Gu, and Vamsi Pingali. Load balanced short path routing in large-scale wireless networks using area-preserving maps. In *Proceedings of the 15th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pages 63–72, New York, NY, USA, 2014.
- [GR01] Chris Godsil and Gordon Royle. *Algebraic Graph Theory*. Graduate Texts in Mathematics. Springer, 2001.
- [GR11] Robert Ghrist and Michael Robinson. Euler–bessel and euler–fourier transforms. *Inverse Problems*, 27(12) :124006, 2011.
- [GRS08] Annarita Giani, Tanya Roosta, and Shankar Sastry. Integrity checker for wireless sensor networks in health care applications. *Pervasive Computing Technologies for Healthcare*, pages 135–138, 2008.
- [Hat01] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2001.
- [HB06] Mohamed Hefeeda and Majid Bagheri. Efficient k-coverage algorithms for wireless sensor networks, 2006.
- [Hew01] Mark Hewish. Little brother is watching you. *Jane’s international defense review*, 2001.
- [Hir03] Anil N. Hirani. *Discrete Exterior Calculus*. PhD thesis, California Institute of Technology, Pasadena, California, May 2003.
- [HK73] J. Hopcroft and R. Karp. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM J. Comput.*, 2(4) :225–231, 1973.
- [HKS⁺04] Tian He, Sudha Krishnamurthy, John A. Stankovic, Tarek Abdelzaher, Liqian Luo, Radu Stoleru, Ting Yan, and Lin Gu. Energy-efficient surveillance system using wireless sensor networks. *Proceedings of the 2nd International Conference on Mobile Systems*, pages 270–283, 2004.

- [HLH14] Bin He, Yonggang Li, Hongwei Huang, and Haifeng Tang. Spatial-temporal compression and recovery in a wireless sensor network in an underground tunnel environment. *Knowledge and Information Systems*, 41 :449–465, 2014.
- [HMM14] Shaun Harker, Konstantin Mischaikow, Marian Mrozek, and Vidit Nanda. Discrete morse theoretic algorithms for computing homology of complexes and maps. *Foundations of Computational Mathematics*, 14(1) :151–184, February 2014.
- [HT03] Chi-Fu Huang and Yu-Chee Tseng. The coverage problem in a wireless sensor network. In *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications (WSNA '03)*, pages 115–121, San Diego, CA, USA, September 2003.
- [HY61] John G. Hocking and Gail S. Young. *Topology*. Dover, 1961.
- [IBS⁺10] François Ingelrest, Guillermo Barrenetxea, Gunnar Schaefer, Martin Vetterli, Olivier Couach, and Marc Parlange. Sensorscope : Application-specific sensor network for environmental monitoring. *ACM Trans. Sen. Netw.*, 6(2) :17 :1–17 :32, March 2010.
- [ILH13] Md. Motaharul Islam, Jun Hyuk Lee, and Eui-Nam Huh. An efficient model for smart home by the virtualization of wireless sensor network. *International Journal of Distributed Sensor Networks*, 2013.
- [IM05] Mohammad Ilyas and Imad Mahgoub. *Handbook of Sensor Networks : Compact Wireless and Wired Sensing Systems*. CRC Press, 2005.
- [IOT12] K. Ishibashi, H. Ochiai, and V. Tarokh. Energy harvesting cooperative communications. In *IEEE International Symposium on Personal Indoor and Mobile Radio Communications*, pages 1819–1823, Sydney, Australia, September 2012.
- [Jay82] E. T. Jaynes. On the rationale of maximum-entropy methods. *Proceedings of the IEEE*, 70(9) :939–952, 1982.
- [JS08] Jehn-Ruey Jiang and Tzu-Ming Sung. Maintaining connected coverage for wireless sensor networks. In *Distributed Computing Systems Workshops, 2008. ICDCS '08. 28th International Conference on*, pages 297–302, June 2008.
- [Jun05] Dieter Jungnickel. *Graphs, Networks and algorithms*. Algorithms and Computation in Mathematics. Springer, second edition edition, 2005.
- [KIR11] A. Kumar, P. Ishwar, and K. Ramchandran. High-resolution distributed sampling of bandlimited fields with low-precision sensors. *IEEE Trans Inform. Theory*, 57(1) :476–492, January 2011.
- [KKM05] Henry King, Kevin Knudson, and Neža Mramor. Generating discrete morse functions from point data. *Experimental Mathematics*, 14(4) :435–444, 2005.
- [KLA05] Santosh Kumar, Ten H. Lai, and Anish Arora. Barrier coverage with wireless sensors. *MobiCom'05*, 2005.

- [KLC⁺10] Jeonggil Ko, Jong Hyun Lim, Yin Chen, Rvǎzvan Musvaloiu-E, Andreas Terzis, Gerald M. Masson, Tia Gao, Walt Destler, Leo Selavo, and Richard P. Dutton. Medisn : Medical emergency detection in sensor networks. *ACM Trans. Embed. Comput. Syst.*, 10(1) :11 :1–11 :29, August 2010.
- [KMM03] T. Kaczynski, K. Mischaikow, and M. Mrozek. *Computational Homology*. Springer, 2003.
- [Koz08] Dmitry Kozlov. *Combinatorial Algebraic Topology*. Springer, 2008.
- [KPC⁺07] Sukun Kim, Shamim Pakzad, David Culler, James Demmel, Gregory Fenves, Steven Glaser, and Martin Turon. Health monitoring of civil infrastructures using wireless sensor networks. In *International Conference on Information Processing in Sensor Networks*, IPSN '07, pages 254–263, New York, NY, USA, 2007. ACM.
- [KSJPB99] J. M. Kahn K. S. J. Pister and B. E. Boser. Smart dust : Wireless networks of millimeter-scale sensor nodes. *Electronics Research Laboratory Research Summary*, 1999.
- [Kuh55] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2) :83–97, 1955.
- [Kum04] Jean Kumagai. The secret life of birds. *IEEE Spectrum*, 2004.
- [KYX14] Zhiping Kang, Honglin Yu, Qingyu Xiong, and Haibo Hu. Spatial-temporal correlative fault detection in wireless sensor networks. *International Journal of Distributed Sensor Networks*, 2014.
- [LCL⁺11] Mo Li, Weifang Cheng, Kebin Liu, Yuan He, Xiangyang Li, and Xiangke Liao. Sweep coverage with mobile sensors. *IEEE Transactions on Mobile Computing*, 10(11) :1534–1545, November 2011.
- [LDW08] Benyuan Liu, Olivier Dousse, Jie Wang, and Anwar Saipulla. Strong barrier coverage of wireless sensor networks. *MobiHoc'08*, 2008.
- [LLS09] Sang Hyuk Lee, Soobin Lee, Heecheol Song, and Hwang Soo Lee. Wireless sensor network design for tactical military applications : Remote large-scale environments. *Military Communications Conference, 2009. MILCOM 2009. IEEE*, pages 1–7, October 2009.
- [LLT04] Thomas Lewiner, Hélio Lopes, and Geovan Tavares. Applications of for-man s discrete morse theory to topology visualization and mesh compression. *Transactions on Visualization and Computer Graphics*, 10(5) :499–508, September 2004.
- [LNS96] Witold Litwin, Marie-Anna Neimat, and Donovan A. Schneider. Lh* — a scalable, distributed data structure. *ACM Trans. Database Syst.*, 21(4) :480–525, December 1996.
- [LTD11] Louise Lamont, Mylène Toulgoat, Mathieu Déziel, and Glenn Patterson. Tiered wireless sensor network architecture for military surveillance applications. In *The Fifth International Conference on Sensor Technologies and Applications (SensorComm 2011)*, pages 288–294, Nice, France, August 2011.

- [Mas91] William S. Massey. *A basic course in algebraic topology*. Springer, 1991.
- [MBE10] K. Martinez, P. Basford, J. Ellul, and R. Clarke. Field deployment of low power high performance nodes. In *Distributed Computing Systems Workshops (ICDCSW), 2010 IEEE 30th International Conference on*, pages 199–205, June 2010.
- [ME06] Abubakr Muhammad and Magnus Egerstedt. Control using higher order laplacians in network topologies. In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems*, pages 1024–1038, Kyoto, Japan, July 2006.
- [ME01] Aridj Mohamed and Zegour Djamel Eddine. Th* :scalable distributed trie hashing. *International Journal of Computer Sciences*, 7(6) :109–115, November 201, 1205.0439.
- [METS+06] Razvan Musaloiu-E, Andreas Terzis, Katalin Szlavecz, Alex Szalay, Joshua Cogan, and Jim Gray. Life under your feet : A wireless soil ecology sensor network. *IEEE*, 2006.
- [Mil63] John Willard Milnor. *Morse theory*. Princeton University Press, 1963.
- [Mil07] Kevin L. Mills. A brief survey of self-organization in wireless sensor networks : Research articles. *Wirel. Commun. Mob. Comput.*, 7(7) :823–834, September 2007.
- [MJ07a] Abubakr Muhammad and Ali Jadbabaie. Decentralized computation of homology groups in networks by gossip. In *American Control Conference (ACC'07)*, pages 3438–3443, New York City, USA, July 2007.
- [MJ07b] Abubakr Muhammad and Ali Jadbabaie. Dynamic coverage verification in mobile sensor networks via switched higher order laplacians. In *Robotics : Science and Systems III*, 2007.
- [MKT13] Isa Maleki, Reza Seyyed Khaze, Mahmoodi Marjan Tabrizi, and Ali Bagherinia. A new approach for area coverage problem in wireless sensor networks with hybrid particle swarm optimization and differential evolution algorithms. *International Journal of Mobile Network Communications & Telemacs*, 3(6) :61–75, December 2013.
- [ML78] Saunders Mac Lane. *Categories for the Working Mathematician*. Springer, 1978.
- [MN13] Konstantin Mischaikow and Vidit Nanda. Morse theory for filtrations and efficient computation of persistent homology. *Discrete & Computational Geometry*, 50(2) :330–353, 2013.
- [MOJ06] Aleksandar Milenkovic, Chris Otto, and Emil Jovanov. Wireless sensor networks for personal health monitoring : Issues and an implementation. *Computer Communications*, 29 :2521–2533, 2006.
- [MSMSM13] Afsaneh Minaie, Ali Sanati-Mehrziy, Paymon Sanati-Mehrziy, and Reza Sanati-Mehrziy. Application of wireless sensor networks in health care system. In *ASEE Annual Conference & Exposition*, Atlanta, GE, USA, June 2013.

- [MT10] Geoff V Merret and Yen Kheng Tan, editors. *Wireless Sensor Networks : Application-Centric Design*. InTech, 2010.
- [Mun57] James Munkres. Algorithms for assignment and transportation problems. *SIAM Journal*, 5(1), March 1957.
- [MW10] Marian Mrozek and Thomas Wanner. Coreduction homology algorithm for inclusions and persistent homology. *Computers & Mathematics with Applications*, 60(10) :2812–2833, 2010.
- [MYM11] G.R. Mendez, M.A.M. Yunus, and S.C. Mukhopadhyay. A wifi based smart wireless sensor network for an agricultural environment. In *Sensing Technology (ICST), 2011 Fifth International Conference on*, pages 405–410, Palmerston North, November 2011.
- [NCV08a] A. Nordio, C.-F. Chiasserini, and E. Viterbo. Performance of linear field reconstruction techniques with noise and uncertain sensor locations. *IEEE Trans. Signal Process.*, 56(8) :3535–3547, August 2008.
- [NCV08b] A. Nordio, C.-F. Chiasserini, and E. Viterbo. Reconstruction of multidimensional signals from irregular noisy samples. *IEEE Trans. Signal Process.*, 56(9) :4274–4285, September 2008.
- [NR11] J. Nachtigall and J.-P. Redlich. Wireless alarming and routing protocol for earthquake early warning systems. In *New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on*, pages 1–6, February 2011.
- [PAK⁺05] N. Patwari, J. N. Ash, S. Kyperountas, III Hero, A. O., R. L. Moses, and N. S. Correal. Locating the nodes : cooperative localization in wireless sensor networks. *IEEE Signal Processing Magazine*, 22(4) :54–69, July 2005.
- [Pen93] Roger Penrose. On the cohomology of impossible figures. In Michele Emmer, editor, *The visual mind*, pages 27–29. MIT Press, Cambridge, MA, USA, 1993.
- [Pie91] Benjamin C. Pierce. *Basic Category Theory for Computer Scientists*. MIT, 1991.
- [PK00] G. J. Pottie and W. J. Kaiser. Wireless integrated network sensors. *Communications of the ACM*, 43(5), May 2000.
- [Poi95] Henri Poincaré. Analysis situs. *Journal de l'École polytechnique*, 2(1) :1–121, 1895.
- [PSM⁺04] Joseph Polastre, Robert Szewczyk, Alan Mainwaring, David Culler, and John Anderson. *Wireless Sensor Networks*, chapter Analysis of Wireless Sensor Networks for Habitat Monitoring, pages 399–423. Kluwer Academic Publishers, Norwell, MA, USA, 2004.
- [QXF⁺14] Lin Qi, Mark Xu, Zetian Fu, Trebar Mira, and Xiaoshuan Zhang. C2slds : A wsn-based perishable food shelf-life prediction and {LSFO} strategy decision support system in cold chain logistics. *Food Control*, 38(0) :19–29, 2014.

- [RAdS⁺00] J.M. Rabaey, M.J. Ammer, Jr. da Silva, J.L., D. Patel, and S. Roundy. Picoradio supports ad hoc ultra-low power wireless networking. *Computer*, 33(7) :42–48, Jul 2000.
- [RF10] P. Rost and G. Fettweis. On the transmission-computation-energy tradeoff in wireless and fixed networks. arxiv, August 2010, 1008.4565.
- [RLS04] Zhi Wang Ruizhong Lin and Youxian Sun. Wireless sensor networks solutions for real time monitoring of nuclear power plant. *IEEE*, 2004.
- [RM04] Kay Romer and Friedemann Mattern. The design space of wireless sensor networks. *Wireless Communications, IEEE*, 11(6) :54–61, December 2004.
- [RM09] G. Reise and G. Matz. Distributed sampling and reconstruction of non-bandlimited fields in sensor networks based on shift-invariant spaces. In *EEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2061–2064, April 2009.
- [Rob13a] Michael Robinson. Analyzing wireless communication network vulnerability with homological invariants. *pre-print*, November 2013, 1311.1532.
- [Rob13b] Michael Robinson. The nyquist theorem for cellular sheaves. In *International Conference on Sampling Theory and Applications*, pages 293–296, Bremen, Germany, July 2013. 1307.7212.
- [Rob14a] Michael Robinson. A sheaf-theoretic perspective on sampling. *pre-print*, May 2014, 1405.0324.
- [Rob14b] Michael Robinson. *Topological signal processing*. Springer, 2014.
- [RW05] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005.
- [RWS11] V. Robins, P. J. Wood, and A. P. Sheppard. Theory and algorithms for constructing discrete morse complexes from grayscale digital images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 33(8) :1646–1658, 2011.
- [SAA04] Jaspal S. Sandhu, Alice M. Agogino, and Adrian K. Agogino. Wireless sensor networks for commercial lighting control : Decision making with multi-agent systems. *AAAI Workshop on Sensor Networks*, 10 :131–140, 2004.
- [Sat99] Hajime Sato. *Algebraic Topology : An Intuitive Approach*, volume 183 of *Translations of Mathematical Monographs*. American Mathematical Society, 1999.
- [SCD⁺05] J. A. Stankovic, Q. Cao, T. Doan, L. Fang, Z. He, R. Kiran, S. Lin, S. Son, R. Stoleru, and A. Wood. Wireless sensor networks for in-home healthcare : Potential and challenges. *HCMDSS*, 2005.
- [SCI⁺01] Eugene Shih, Seong-Hwan Cho, Nathan Ickes, Rex Min, Amit Sinha, Alice Wang, and Anantha Chandrakasan. Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, MobiCom '01, pages 272–287, New York, NY, USA, 2001. ACM.

- [SCL⁺05] Victor Shnayder, Borrong Chen, Konrad Lorincz, Thaddeus R. F. Fulford-Jones, and Matt Welsh. Sensor networks for medical care. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems*, 2005.
- [SG13] R. Sarkar and Jie Gao. Differential forms for target tracking and aggregate queries in distributed networks. *IEEE/ACM Transactions on Networking*, 21(4) :1159–1172, 2013.
- [SNM07] Ivan Stoianov, Lama Nachman, Sam Madden, and Timur Tokmouline. Pipenet : a wireless sensor network for pipeline monitoring. In *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, IPSN '07, pages 264–273, New York, NY, USA, 2007. ACM.
- [SP04] E. Shi and A. Perrig. Designing secure sensor networks. *Wireless Communications, IEEE*, 11(6) :38–43, December 2004.
- [SRL07] Isabela G. Siqueira, Linnyer Beatrys Ruiz, Antonio A. F. Loureiro, and José Marcos Nogueira. Coverage area management for wireless sensor networks. *Int. J. Netw. Manag.*, 17(1) :17–31, January 2007.
- [Sta09] Stanislava Stankovic. Medical applications based on wireless sensor networks. *Transactions on Internet Research*, 5(2), July 2009.
- [SZG⁺08] R. Sarkar, Xianjin Zhu, Jie Gao, L. J. Guibas, and J. S. B. Mitchell. Isocontour queries and gradient descent with guaranteed delivery in sensor networks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, 2008.
- [TCC13] Aslan Tchamkerten, Venkat Chandar, and Giuseppe Caire. Energy and sampling constrained asynchronous communication. arxiv, February 2013, 1302.6574.
- [TSJ10] Alireza Tahbaz-Salehi and Ali Jadbabaie. Distributed coverage verification in sensor networks without location information. *IEEE Transactions on Automatic Control*, 55(8) :1837–1849, August 2010.
- [TVP⁺11] D.S. Tudose, A. Voinescu, M. Petrareanu, A. Bucur, D. Loghin, A. Bostan, and N. Tapus. Home automation design using 6lowpan wireless sensor networks. In *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*, pages 1–6, June 2011.
- [UV13] Jayakrishnan Unnikrishnan and Martin Vetterli. Sampling and reconstruction of spatial fields using mobile sensors. *IEEE Transactions on Signal Processing*, 61(9) :2328–2340, May 2013.
- [VDM13a] A. Vergne, L. Decreasefond, and P. Martins. Reduction algorithm for simplicial complexes. In *INFOCOM, 2013 Proceedings IEEE*, pages 475–479, 2013.
- [VDM13b] Anaïs Vergne, Laurent Decreasefond, and Philippe Martins. Simplicial homology for future cellular networks. *pre-print*, December 2013, 1312.1664.

- [VRM⁺12] J. Valverde, V. Rosello, G. Mujica, J. Portilla, A. Uriarte, and T. Riesgo. Wireless sensor network for environmental monitoring : Application in a coffee factory. *International Journal of Distributed Sensor Networks*, 2012.
- [VWS⁺06] G. Virone, A. Wood, L. Selavo, Q. Cao, L. Fang, T. Doan, Z. He, R. Stoleru, S. Lin, and J.A. Stankovic. An advanced wireless sensor network for health monitoring. In *Transdisciplinary Conference on Distributed Diagnosis and Home Healthcare (D2H2)*, Arlington, VA, April 2006.
- [WALW⁺06] Geoffrey Werner-Allen, Konrad Lorincz, Matt Welsh, Omar Marcillo, Jeff Johnson, Mario Ruiz, and Jonathan Lees. Deploying a wireless sensor network on an active volcano. *IEEE Computer Society*, 2006.
- [WC07] Andrew Wheeler and Ember Corporation. Commercial applications of wireless sensor networks using zigbee. *Communications Magazine, IEEE*, 45(4) :70–77, April 2007.
- [Wol59] Philip Wolfe. The simplex method for quadratic programming. *Econometrica*, 27(3) :382–398, July 1959.
- [WTH08] Michael Winkler, Klaus-Dieter Tuchs, Kester Hughes, and Graeme Barclay. Theoretical and practical aspects of military wireless sensor networks. *Journal of Telecommunications & Information Technology*, 2, 2008.
- [WXX12] Yang Weng, Lihua Xie, and Wendong Xiao. Sensor selection for random field estimation in wireless sensor networks. *Journal of Systems Science and Complexity*, 25 :46–59, 2012.
- [XRC⁺04] Ning Xu, Sumit Rangwala, Krishna Kant Chintalapudi, Deepak Ganesan, Alan Broad, Ramesh Govindan, and Deborah Estrin. A wireless sensor network for structural monitoring. In *Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems*, SenSys '04, pages 13–24, New York, NY, USA, 2004. ACM.
- [YDC05] Shuhui Yang, Fei Dai, M. Cardei, and Jie Wu. On multiple point coverage in wireless sensor networks. In *Mobile Adhoc and Sensor Systems Conference, 2005. IEEE International Conference on*, November 2005.
- [YMD12] Feng Yan, Philippe Martins, and Laurent Decreusefond. Accuracy of homology based approaches for coverage hole detection in wireless sensor networks. In *International Conference on Communications*, Ottawa, Canada, 2012.
- [YMG08] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Comput. Netw.*, 52(12) :2292–2330, August 2008.
- [YWM05] Liyang Yu, Neng Wang, and Xiaoqiao Meng. Real-time forest fire detection with wireless sensor networks. *IEEE*, 2005.
- [YXCO11] MI Yunfei Xu, Jongeun Choi, and Songhwai Oh. Mobile sensor network navigation using gaussian processes with truncated observations. *IEEE Trans. Robot.*, 27(6) :1118–1131, 2011.

- [ZC05] Afra Zomorodian and Gunnar Carlsson. Computing persistent homology. *Discrete Comput. Geom.*, 33(2) :249–274, February 2005.
- [ZG04] Feng Zhao and Leonidas Guibas. *Wireless Sensor Networks : An Information Processing Approach*. Morgan Kaufmann, 2004.
- [ZGC13] Mengyi Zhang, Alban Goupil, Maxime Colas, and Guillaume Gellé. Wsn sensing coverage based on correlation. In *Proceedings of International Conference on Wireless Communications and Signal Processing (WCSP)*, Hangzhou, China, October 2013.
- [ZGC14] Mengyi Zhang, Alban Goupil, Maxime Colas, and Guillaume Gellé. Adaptive and distributed homology computation with application to large scale sensor networks. In *Proceedings of International Conference on Wireless Communications and Signal Processing (WCSP)*, Hefei, China, October 2014.
- [ZGC15] Mengyi Zhang, Alban Goupil, Maxime Colas, and Guillaume Gellé. Activation adaptative et distribuée des unités d’un réseau de capteurs. In *Soumission au colloque Grets, 2015*.
- [ZH05] Honghai Zhang and Jennifer C. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc & Sensor Wireless Networks*, 2005.
- [ZLZ12] Yiyang Zhang, Xiangzhen Li, Suxiang Zhang, and Yan Zhen. Wireless sensor network in smart grid : Applications and issues. *Information and Communication Technologies (WICT), 2012 World Congress*, pages 1204–1208, October 2012.
- [Zom05] Afra J. Zomorodian. *Topology for Computing*. Cambridge University Press, 2005.
- [Zom09] Afra Zomorodian. *Algorithms and theory of computation handbook*, chapter Computational Topology, pages 1–31. Chapman and Hall, second edition, 2009.
- [ZSL⁺10] Wei Zeng, Rik Sarkar, Feng Luo, Xianfeng David Gu, and Jie Gao. Resilient routing for sensor networks using hyperbolic embedding of universal covering space. In *Proc. of the 29th Annual IEEE Conference on Computer Communications (INFOCOM’10)*, April 2010.
- [ZST06] Qing Zhao, Ananthram Swami, and Lang Tong. The interplay between signal processing and networking in sensor networks. *IEEE Signal Processing Magazine*, 23(4) :84–93, July 2006.
- [ZZS12] Chuan Zhu, Chunlin Zheng, Lei Shu, and Guangjie Han. A survey on coverage and connectivity issues in wireless sensor networks. *Journal of Network and Computer Applications*, 35(2) :619–632, 2012. Simulation and Testbeds.

Optimisation de la couverture de communications et de mesure dans les réseaux de capteurs

Un réseau de capteurs sans fil résulte du déploiement d'un ensemble de petites unités autonomes interagissant via un réseau construit grâce à leur module de communication qui observent leur environnement par des capteurs pour ensuite traiter et/ou sauvegarder cette information via leur capacité calculatoire et de stockage.

La couverture est la seule représentation disponible aux réseaux de capteurs de l'espace physique environnant. Par conséquent, il est essentiel de pouvoir qualifier et quantifier sa qualité notamment concernant la présence de trous.

Nos travaux utilisent la topologie algébrique pour répondre à ces problèmes. Plus précisément, nous définissons dans un premier temps une notion de trou de couverture d'un champ scalaire qui mesure la qualité de l'estimation par le réseau de capteurs sans pour autant connaître la position des capteurs. Cela permet d'utiliser l'homologie simpliciale pour déterminer la qualité de la couverture globale et accessoirement de mettre en veille certains capteurs surnuméraires tout en garantissant la couverture.

Puis, afin de rendre le résultat précédent facilement calculable par un réseau de capteurs grâce à une distribution du calcul qui supporte en plus le passage à l'échelle, nous utilisons la théorie de Morse discrète pour faire le calcul des groupes d'homologie nécessaires à notre application précédente.

Enfin, cette dernière approche est rendue suffisamment souple pour permettre le suivi temporel des modifications de la couverture de manière délocalisée. Cela permet non seulement de suivre la qualité de la couverture lorsque l'environnement se modifie mais aussi de proposer un schéma distribué de mise en veille des capteurs afin d'augmenter la durée de vie du réseau de capteurs tout en garantissant une couverture suffisante.

Mots clefs réseaux de capteurs, couverture, topologie algébrique, homologie

Communication and measurement coverage optimization in Wireless Sensor Networks

A wireless sensor network consists of a set of small autonomous units that interact via a network built by their communication modules. They observe their environment by their sensors and then they manage this information according to their computational capacity and storage.

The coverage is the only representation available to the sensor network of its environment. Therefore, it is essential to quantify the quality of coverage especially related to the presence of holes.

Our work uses algebraic topology to solve these problems. We first define a notion of the coverage hole in a scalar field, which measures the quality of the estimation by the sensor network without knowing the positions of the sensors. It allows the simplicial homology tool to determine the quality of the overall coverage and put certain redundant sensors into sleeping mode with the guarantee of the coverage.

Then, to make the previous result easier to compute by a sensor network, the discrete Morse theory is used. It allows a distributed computation of the previous homology groups while supporting scalability necessary in sensor networks domain.

Finally, one flexible approach that allows time varying tracking which allows a coverage is proposed in a distributed way. When the environment changes, this approach can not only guarantee the capability of monitoring of coverage quality, but also proposes a scheme to send to sleep the redundant sensors in order to increase the lifetime of the sensor network with adequate coverage.

Keywords sensor networks, coverage, algebraic topology, homology

Discipline Génie informatique, automatique et traitement du signal

Unité de recherche Centre de Recherche en STIC, Université de Reims Champagne-Ardenne