Algebraic topology for wireless sensor networks

Anaïs Vergne

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2013-ENST-0070

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité “Informatique et Réseaux”

présentée et soutenue publiquement par

Anaïs VERGNE

le 28 novembre 2013

Topologie algébrique

appliquée aux réseaux de capteurs

Directeurs de thèse : **Laurent DECREUSEFOND**
Philippe MARTINS

Jury

M. Samir TOHMÉ, Professeur, Université de Versailles St-Quentin-en-Yvelines, France	Président
M. Frédéric CHAZAL, Directeur de Recherche, Inria Saclay, France	Rapporteur
M. Brian MARK, Professeur, George Mason University, Virginia, USA	Rapporteur
M. Bartłomiej BŁASZCZYSZYN, Directeur de Recherche, Inria Paris, France	Examinateur
M. Jérôme BROUET, Ingénieur, Alcatel-Lucent, France	Examinateur
M. Xavier LAGRANGE, Professeur, Télécom Bretagne, France	Examinateur
M. Laurent DECREUSEFOND, Professeur, Télécom ParisTech, France	Directeur de thèse
M. Philippe MARTINS, Professeur, Télécom ParisTech, France	Directeur de thèse

TELECOM ParisTech
école de l’Institut Mines-Télécom - membre de ParisTech
46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - www.telecom-paristech.fr

T H È S E

Abstract

Simplicial complex representation gives a mathematical description of the topology of a wireless sensor network, i.e., its connectivity and coverage. In these networks, sensors are randomly deployed in bulk in order to ensure perfect connectivity and coverage. We propose an algorithm to discover which sensors are to be switched off, without modification of the topology, in order to reduce energy consumption. Our reduction algorithm can be applied to any type of simplicial complex and reaches an optimum solution. For random geometric simplicial complexes, we find boundaries for the number of removed vertices, as well as mathematical properties for the resulting simplicial complex. The complexity of our reduction algorithm boils down to the computation of the asymptotical behavior of the clique number of a random geometric graph. We provide almost sure asymptotical behavior for the clique number in all three percolation regimes of the geometric graph.

In the second part, we apply the simplicial complex representation to cellular networks and improve our reduction algorithm to fit new purposes. First, we provide a frequency auto-planning algorithm for self-configuration of SON in future cellular networks. Then, we propose an energy conservation fot the self-optimization of wireless networks. Finally, we present a disaster recovery algorithm for any type of damaged wireless network. In this last chapter, we also introduce the simulation of determinantal point processes in wireless networks.

Résumé

La représentation par complexes simpliciaux fournit une description mathématique de la topologie d'un réseau de capteurs, c'est-à-dire sa connectivité et sa couverture. Dans ces réseaux, les capteurs sont déployés aléatoirement en grand nombre afin d'assurer une connectivité et une couverture parfaite. Nous proposons un algorithme qui permet de déterminer quels capteurs mettre en veille, sans modification de topologie, afin de réduire la consommation d'énergie. Notre algorithme de réduction peut être appliqué à tous les types de complexes simpliciaux, et atteint un résultat optimal. Pour les complexes simpliciaux aléatoires géométriques, nous obtenons des bornes pour le nombre de sommets retirés, et trouvons des propriétés mathématiques pour le complexe simplicial obtenu. En cherchant la complexité de notre algorithme, nous sommes réduits à calculer le comportement asymptotique de la taille de la plus grande clique dans un graphe géométrique aléatoire. Nous donnons le comportement presque sûr de la taille de la plus grande clique pour les trois régimes de percolation du graphe géométrique.

Dans la deuxième partie, nous appliquons la représentation par complexes simpliciaux aux réseaux cellulaires, et améliorons notre algorithme de réduction pour répondre à de nouvelles demandes. Tout d'abord, nous donnons un algorithme pour la planification automatique de fréquences, pour la configuration automatique des réseaux cellulaires de la nouvelle génération bénéficiant de la technologie SON. Puis, nous proposons un algorithme d'économie d'énergie pour l'optimisation des réseaux sans fil. Enfin, nous présentons un algorithme pour le rétablissement des réseaux sans fil endommagés après une catastrophe. Dans ce dernier chapitre, nous introduisons la simulation des processus ponctuels déterminantaux dans les réseaux sans fil.

List of publications

Main author

- Anaïs Vergne, Laurent Decreusefond and Philippe Martins. *Reduction algorithm for simplicial complexes.* In IEEE INFOCOM, Turin, Italy, 14-19 April 2013, pp. 475-479.
http://hal.archives-ouvertes.fr/hal-00688919
Chapter 3

- Anaïs Vergne, Laurent Decreusefond and Philippe Martins. *Clique number of random geometric graphs.* Submitted, 2013.
http://hal.archives-ouvertes.fr/hal-00864303
Chapter 4

- Anaïs Vergne, Ian Flint, Laurent Decreusefond and Philippe Martins. *Homology based algorithm for disaster recovery in wireless networks* Submitted, 2013.
http://hal.archives-ouvertes.fr/hal-00800520
Chapter 7

- Chapters 5 and 6 will be subjects to future publications.

Collaborations

- Laurent Decreusefond, Eduardo Ferraz, Hugues Randriambololona and Anaïs Vergne. *Simplicial Homology of Random Configurations.* In Advances in Applied Probability 46, 2 (2014) 1-20.
http://hal.archives-ouvertes.fr/hal-00578955
Appendix A

- Laurent Decreusefond, Ian Flint and Anaïs Vergne. *Efficient simulation of the Ginibre process.* Submitted, 2013.
http://hal.archives-ouvertes.fr/hal-00869259
Appendix B

- Feng Yan, Anaïs Vergne, Laurent Decreusefond and Philippe Martins. *Homology-based Distributed Coverage Hole Detection in Wireless Sensor Networks.* Submitted, 2013.
http://hal.archives-ouvertes.fr/hal-00783403

Contents

French detailed summary

L'objectif de ce chapitre est de résumer de manière détaillée en français le travail présenté en anglais dans ce manuscrit.

0.1 Introduction

0.1.1 Motivation

L'utilisation des réseaux de capteurs sans fil a considérablement augmenté pendant ces dernières années. En effet, ils sont utiles dans tous les domaines où la surveillance et l'observation jouent un rôle. Cela va de la surveillance de zone de combat au dénombrement ciblé en agriculture, en passant par le contrôle de l'environnement. Qui plus est, la miniaturisation et le faible coût des circuits électroniques ont permis la naissance de capteurs multifonctions à bas coût. Les capteurs sont ainsi déployés afin de superviser ou récupérer des informations sur un domaine donné. Le facteur clef pour la qualité de service d'un réseau de capteurs sans fils est donc sa topologie. En deux dimensions, la topologie d'un réseau comprend sa connectivité et sa couverture. Par exemple, la connectivité d'un réseau de capteurs est nécessaire pour compter les traversées sur une ligne de capteurs, ou les entrées dans une domaine borné par des capteurs. Dans un cas plus général, les capteurs ont besoin d' être connectés pour transférer les données collectées à un serveur central, étant donné que ces derniers n'ont pas de grandes capacités de mémoire. Ensuite, la couverture d'un réseau de capteurs défini le domaine surveillé, la plupart du temps, une couverture totale est exigée.

La première approche à la gestion des réseaux de capteurs sans fil est de découvrir sa qualité de service, c'est-à-dire sa topologie. Pour connaître la topologie d'un réseau, une solution est de déployer les capteurs selon un schéma régulier (hexagonal, grille, losanges, triangulaire...) comme dans [10]. Cependant, le domaine ciblé ne permet pas toujours un déploiement si précis. De plus, il est possible que la topologie d'un réseau de capteurs soit modifiée avec le temps : des capteurs peuvent être détruits, ne plus avoir de batterie, ou encore les communications peuvent être perturbées par le climat. Une seconde approche est ensuite de considérer un déploiement aléatoire, qui peut ainsi générer des amas de capteurs, aussi bien que des trous de couvertures. Il y a donc beaucoup de littérature sur le problème de couverture dans les réseaux de capteurs sans fil déployés aléatoirement. Parmi les méthodes les plus populaires, on peut citer les méthodes basées sur la localisation : par exemple dans [33] où la localisation exacte de chacun des capteurs est nécessaire. On peut aussi citer les méthodes basées sur les distances entre capteurs : comme dans [92] où la couverture est calculée à partir de celles-ci. Cependant, ces deux méthodes nécessitent des informations géométriques qui ne sont pas toujours disponibles. En fait, elles rencontrent les mêmes problèmes que le déploiement selon un schéma régulier : le

domaine ciblé ne permet pas forcément une localisation précise ou des mesures de distances, et ces paramètres peuvent être modifiés par le temps.

C'est pourquoi les méthodes basées sur la connectivité entre capteurs paraissent plus intéressantes vu qu'elles ne nécessitent pas d'informations géométriques. Dans [37], Ghrist et al. ont présenté le complexe de Rips-Vietoris, défini comme le complexe des cliques du graphe de voisinage des capteurs, qui détermine la couverture via l'homologie du complexe. Le calcul de la couverture par l'homologie simpliciale est réduit à de simples manipulations d'algèbre linéaire. [24], [67] ou [94] utilisent l'homologie simpliciale comme outil pour un opérateur afin d' évaluer la qualité d'un réseau. Une version distribuée de ces algorithmes est présentée dans [89] afin de détecter les trous de couverture. D'un autre côté, l'homologie simpliciale sur les configurations aléatoires a aussi été abordée mathématiquement en recherche. Les moments de plusieurs caractéristiques d'homologie simpliciale peuvent être obtenus pour un processus ponctuel binomial, cf [51], ou pour un processus de Poisson, cf [26].

0.1.2 Contributions et plan

Le manuscrit en anglais est organisé comme suit. Premièrement, le sujet de thèse est introduit dans le Chapitre 1. Puis, nous rappelons dans le Chapitre 2 des notions nécessaires en homologie simpliciale pour la compréhension du manuscrit. Ce chapitre est donné en français dans la Section 0.2. Une section sur les travaux connexes est incluse dans la rédaction en anglais.

Dans le Chapitre 3, traduit en français dans la Section 0.3, nous avons pour but de réduire la consommation d'énergie dans les réseaux de capteurs sans fil en mettant en veille les capteurs en surnombre. Nous utilisons les complexes simpliciaux aléatoires pour fournir une représentation précise et transposable de la topologie d'un réseau de capteurs sans fil. Etant donné un complexe simplicial, nous proposons un algorithme qui réduit le nombre de ses sommets, sans modifier sa topologie (i.e. connectivité et couverture). Nous donnons aussi des résultats de simulations pour les cas usuels, principalement les complexes de couverture permettant de représenter les réseaux de capteurs sans fil. Nous montrons que l'algorithme atteint un équilibre de Nash. De plus, nous trouvons une borne supérieure et une borne inférieure pour le nombre de sommets retirés, la complexité de l'algorithme, et l'ordre maximal du complexe final dans le cas du problème de couverture.

Les autres résultats de cette thèse sont résumés en français dans la Section 0.4.

En calculant la complexité de l'algorithme de réduction, on est ramené à la recherche du comportement de la taille du plus grand simplexe du complexe géométrique aléatoire. En vocabulaire de théorie des graphes, cela se traduit par la recherche du comportement du cardinal de la clique maximum du graphe géométrique aléatoire. Dans le Chapitre 4, nous décrivons son comportement asymptotique lorsque le nombre de sommets tend vers l'infini. Ce comportement dépend du régime de percolation dans lequel le graphe se trouve. Les comportements asymptotiques presque sûrs sont explicités dans chacun des trois régimes. Nous donnons aussi les comportements asymptotiques de caractéristiques du graphe en lien avec le cardinal de la clique maximum : le nombre chromatique, le degré maximum, et la taille du stable maximum.

La représentation par complexes simpliciaux n'est pas utiles que pour les réseaux de capteurs sans fil, mais aussi pour tous les types de réseaux sans fil ou la connectivité et la couverture sont des facteurs clef. En particulier, nous avons choisi de considérer les réseaux cellulaires, et dans les Chapitres 5, 6, et 7 nous appliquons la représentation par

complexes simpliciaux aux réseaux cellulaires, et améliorons notre algorithme de réduction pour complexes simpliciaux pour atteindre de nouveaux objectifs.

Les réseaux cellulaires LTE supportent la technologie SON (Self-Organizing Network). En particulier la Release 8 propose la détection automatique des voisins : chaque eNode-B maintient une table de voisinage dynamique. Cette nouveauté renforce l'usage de la représentation par complexes simpliciaux. Dans le Chapitre 5, nous sommes intéressés par l'implémentation de la fonction de configuration automatique pour la planification des fréquences des réseaux cellulaires du futur. Nous donnons un algorithme pour la planification automatique des fréquences qui appelle plusieurs instances de notre algorithme de réduction pour complexes simpliciaux afin de minimiser le nombre de fréquences utilisées tout en maximisant la couverture de chaque fréquence.

Dans le chapitre 6, notre algorithme de réduction pour complexes simpliciaux est modifié significativement pour l'optimisation automatique des réseaux cellulaires en heures creuses. En effet, une des fonctions d'optimisation automatique de la technologie SON est la possibilité de mettre en veille certaines stations de base d'un réseau pendant les heures creuses. Cependant, notre algorithme ne peut pas directement être appliqué étant donné que la qualité de service requise n'est plus la simple couverture, comme pour un réseau de capteurs, mais dépend du trafic utilisateur. L'algorithme de réduction est donc amélioré pour qu'un réseau puisse satisfaire n'importe quelle qualité de service tout en consommant un minimum d' énergie. Nous présentons notre algorithme d' économie d' énergie et discutons ses performances.

Enfin, dans le Chapitre 7, nous proposons un algorithme pour la réparation de réseaux sans fil après un désastre. Nous considérons un réseau endommagé avec des trous de couverture qui doivent être restaurés. Nous proposons un algorithme de recouvrement après un désastre qui ajoute des sommets en surnombre pour couvrir la totalité du domaine, puis utilise notre algorithme de réduction pour atteindre un résultat optimal avec un nombre minimal de sommets ajoutés. Pour l'ajout des nouveaux sommets, nous proposons l'utilisation de processus ponctuels déterminantaux qui créent de la répulsion entre les sommets, et facilite ainsi intrinsèquement l'identification des trous de couverture. Nous comparons dans un premier temps différentes méthodes d'ajout de sommets : déterminantal et classiques. Puis dans un second temps, nous comparons l'algorithme en entier avec l'algorihme glouton pour le problème de couverture d'ensembles.

La manuscrit est conlut par le Chapitre 8, dans lequel les contributions majeures sont rappelées, et les ouvertures possibles discutées.

Finalement, deux annexes sont incluses sur des collaborations qui ne rentrent pas directement dans le sujet principal du manuscrit. Dans l'Annexe A, le calcul de Malliavin est appliqué pour calculer les moments de caractéristiques du complexe simplicial géométrique basé sur un processus de Poisson. Nous proposons dans l'Annexe B une méthode novatrice pour la simulation du processus déterminantal de Ginibre avec un nombre donné de sommets sur un compact.

0.2 Homologie simpliciale

0.2.1 Définitions

Pour représenter un réseau de capteurs, la première idée est un graphe géométrique. Les capteurs sont représentés par des sommets, et une arête est tracée dès que deux capteurs peuvent communiquer entre eux. Cependant, la représentation par graphe a quelques

limitations. Premièrement, il n'y a aucune notion de couverture. Les graphes peuvent être généralisés à des objets combinatoires génériques appelés complexes simpliciaux. Les graphes permettent de modéliser les relations binaires, alors que les complexes simpliciaux peuvent modéliser des relations d'ordre supérieur. Un complexe simplicial est un objet combinatoire constitué de sommets, arêtes, triangles, tétrahèdres, et leurs équivalents n-dimensionnels. Etant donné un ensemble de sommets V et un entier k, un k-simplexe est un sous-ensemble non-ordonné de $k+1$ sommets $[v_0, v_1 \ldots, v_k]$ où $v_i \in V$ et $v_i \neq v_j$ pour tout $i \neq j$. Ainsi, un 0-simplexe est un sommet, un 1-simplexe est une arête, un 2-simplexe un triangle, un 3-simplexe un tétrahèdre, etc, comme on peut voir sur la Figure 1 par exemple.

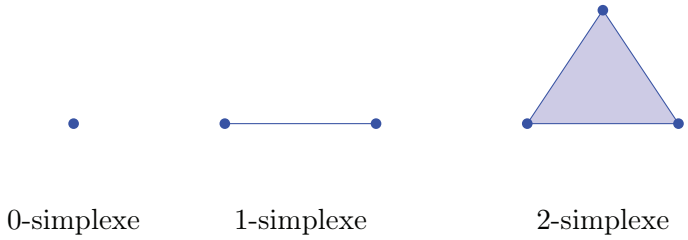


Figure 1: Exemple de k-simplexes.

Tout sous-ensemble de sommets inclus dans l'ensemble des $k+1$ sommets d'un k-simplexe constitue une face de ce k-simplexe. Ainsi, un k-simplexe a exactement $k+1$ $(k-1)$-faces, qui sont des $(k-1)$-simplexes. Par exemple, un tétrahèdre a quatre 3-faces qui sont des triangles. La notion inverse de face est coface : si un simplexe S_1 est une face d'un simplexe plus grand S_2, alors S_2 est une coface de S_1. Un complexe simplicial est un ensemble de simplexes fermé pour l'inclusion des faces, i.e. toutes les faces d'un simplexe sont dans l'ensemble des simplexes, et quand deux faces s'intersectent, leur intersection est un simplexe commun. Un complexe simplicial abstrait est la description purement combinatoire d'un complexe simplicial géométrique, et ainsi n'a pas la propriétd'intersection des simplexes. Pour plus de détails sur la topologie algébrique, nous nous reportons à [40].

Pour le reste de cette dissertation, l'adjectif "abstrait" de complexe simplicial abstrait pourra être omis pour une lecture plus fluide. Cependant, tous les complexes simpliciaux de ce travail sont des complexes simpliciaus abstraits.

On peut définir une orientation pour un simplexe. Un changement d'orientation correspondrait à un changement de signe sur le coefficient. Par exemple, si on échange deux sommets v_i et v_j :

$$[v_0, \ldots, v_i, \ldots, v_j, \ldots, v_k] = -[v_0, \ldots, v_j, \ldots, v_i, \ldots, v_k].$$

Ensuite, on définit l'espece vectoriel des k-simplexes muni d'un endomorphisme, appelé différentielle de carré nul:

Definition 1. *Pour un complexe simplicial abstrait X, pour chaque entier k, $C_k(X)$ est l'espace vectoriel formé par l'ensemble des k-simplexes orientés de X.*

Definition 2. *La différentielle de carré nul ∂_k est la transformation linéaire $\partial_k : C_k \to C_{k-1}$ qui agit sur les éléments de base $[v_0, \ldots, v_k]$ de C_k via*

$$\partial_k[v_0, \ldots, v_k] = \sum_{i=0}^{k} (-1)^i [v_0, \ldots, v_{i-1}, v_{i+1}, \ldots, v_k].$$

La différentielle sur tout k-simplexe est les cycle de ses $k-1$-faces. Cette différentielle donne naissance à un complexe de chaines : une suite d'espaces vectoriels et de transformations linéaires.

$$\dots \xrightarrow{\partial_{k+2}} C_{k+1} \xrightarrow{\partial_{k+1}} C_k \xrightarrow{\partial_k} C_{k-1} \xrightarrow{\partial_{k-1}} \dots \xrightarrow{\partial_1} C_0 \xrightarrow{\partial_0} 0.$$

Finalement, on définit :

Definition 3. *Le k-ième groupe des bords de X est $B_k(X) = \operatorname{im} \partial_{k+1}$.*

Definition 4. *Le k-ième groupe des cycles de X est $Z_k(X) = \ker \partial_k$.*

Si on applique la différentielle à un cycle, elle donne le cycle de ce cycle, ce qui est l'élément nul comme on peut le voir sur la Figure 2. Ainsi un résultat classique dit que pour tout entier k,

$$\partial_k \circ \partial_{k+1} = 0.$$

Il s'ensuit que $B_k \subset Z_k$.

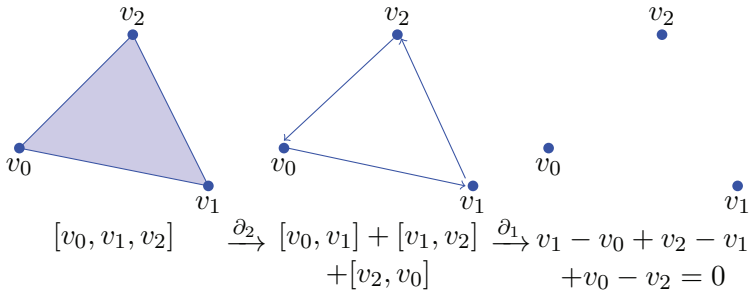


Figure 2: La différentielle appliquée à un 2-simplexe.

On peut maintenant définir le k-ième groupe d'homologie et sa dimension :

Definition 5. *La k-ième homologie de X est défini comme étant le quotient du noyau de la différentielle par son image :*

$$H_k(X) = \frac{Z_k(X)}{B_k(X)}.$$

Definition 6. *Le k-ième nombre de Betti de X est la dimension de sa k-ième homologie :*

$$\beta_k = \dim H_k = \dim Z_k - \dim B_k.$$

On peut calculer les nombres de Betti dans un cas simple par exemple. Soit X un complexe simpliciale formé de 5 sommets $[v_0], \dots, [v_4]$, 6 arêtes $[v_0, v_1]$, $[v_0, v_2]$, $[v_1, v_2]$, $[v_1, v_4]$, $[v_2, v_3]$ et $[v_3, v_4]$, et un triangle $[v_0, v_1, v_2]$. X est représenté dans la Figure 3.

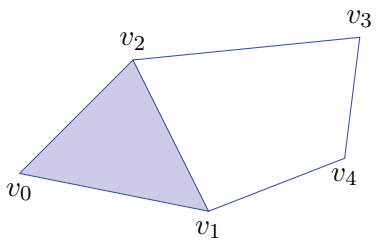


Figure 3: Représentation géométrique de X.

Les différentielles associées à X sont facile à obtenir sous forme matricielle :

$$\partial_1 = \begin{array}{c} \\ [v_0] \\ [v_1] \\ [v_2] \\ [v_3] \\ [v_4] \end{array} \begin{array}{c} \begin{array}{cccccc} [v_0v_1] & [v_0v_2] & [v_1v_2] & [v_1v_4] & [v_2v_3] & [v_3v_4] \end{array} \\ \begin{pmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & -1 & 0 & 0 \\ 0 & 1 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array},$$

$$\partial_2 = \begin{array}{c} \\ [v_0, v_1] \\ [v_0, v_2] \\ [v_1, v_2] \\ [v_1, v_4] \\ [v_2, v_3] \\ [v_3, v_4] \end{array} \begin{array}{c} \begin{array}{c} [v_0, v_1, v_2] \end{array} \\ \begin{pmatrix} 1 \\ -1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \end{array}.$$

La différentielle ∂_0 est l'endomorphisme nul sur l'espace des sommets. On peut donc en déduire les nombres de Betti :

$$\begin{aligned} \beta_0(X) &= \dim \ker \partial_0 - \dim \operatorname{im} \partial_1 \\ &= 5 - 4 \\ &= 1 \\ \beta_1(X) &= \dim \ker \partial_1 - \dim \operatorname{im} \partial_2 \\ &= 2 - 1 \\ &= 1 \end{aligned}$$

0.2.2 Complexes simpliciaux abstraits

Il existe plusieurs types de complexes simpliciaux abstraits célèbres. Nous nous concentrons sur deux complexes particuliers.

Definition 7 (Complexe de Čech)**.** *Soient (X, d) un espace métrique, ω un ensemble fini de points dans X, et ϵ un réel positif. Le complexe de Čech de paramètre ϵ de ω, noté $\mathcal{C}_\epsilon(\omega)$, est le complexe simplicial abstrait dont les k-simplexes sont les $(k+1)$-tuples de points de ω pour lesquels l'intersection des $k+1$ boules de rayon ϵ centrées sur les sommets est non vide.*

Ainsi le complexe de Čech caractérise la couverture d'un domaine, c'ets la représentation que l'on va utiliser pour représenter un réseau de capteurs sans fil.

Cependant, le complexe de Čech est complexe à simuler, il existe une approximation :

Definition 8 (Complexe de Rips-Vietoris)**.** *Soient (X, d) un espace métrique, ω un ensemble fini de points dans X et ϵ un réel positif. Le complexe de Rips-Vietoris de paramètre ϵ de ω, noté $\mathcal{R}_\epsilon(\omega)$, est le complexe simplicial abstrait dont les k-simplexes sont les $(k+1)$-tuples de points de ω qui sont de distance inférieure à ϵ deux à deux.*

On peut voir un exemple de représentation d'un réseau de capteurs par un complexe de Rips-Vietoris dans la Figure 4.

Seule l'information de graphe est nécessaire pour construire le complexe de Rips-Vietoris. De la même manière il ets possible de construire un complexe simplicial à partir de n'importe quel graphe. Chaque k-simplexe est alors inclus dans le complexe si toutes ses $(k-1)$-faces le sont déjà. Le complexe ainsi défini est appelé le complexe de cliques d'un graphe donné.

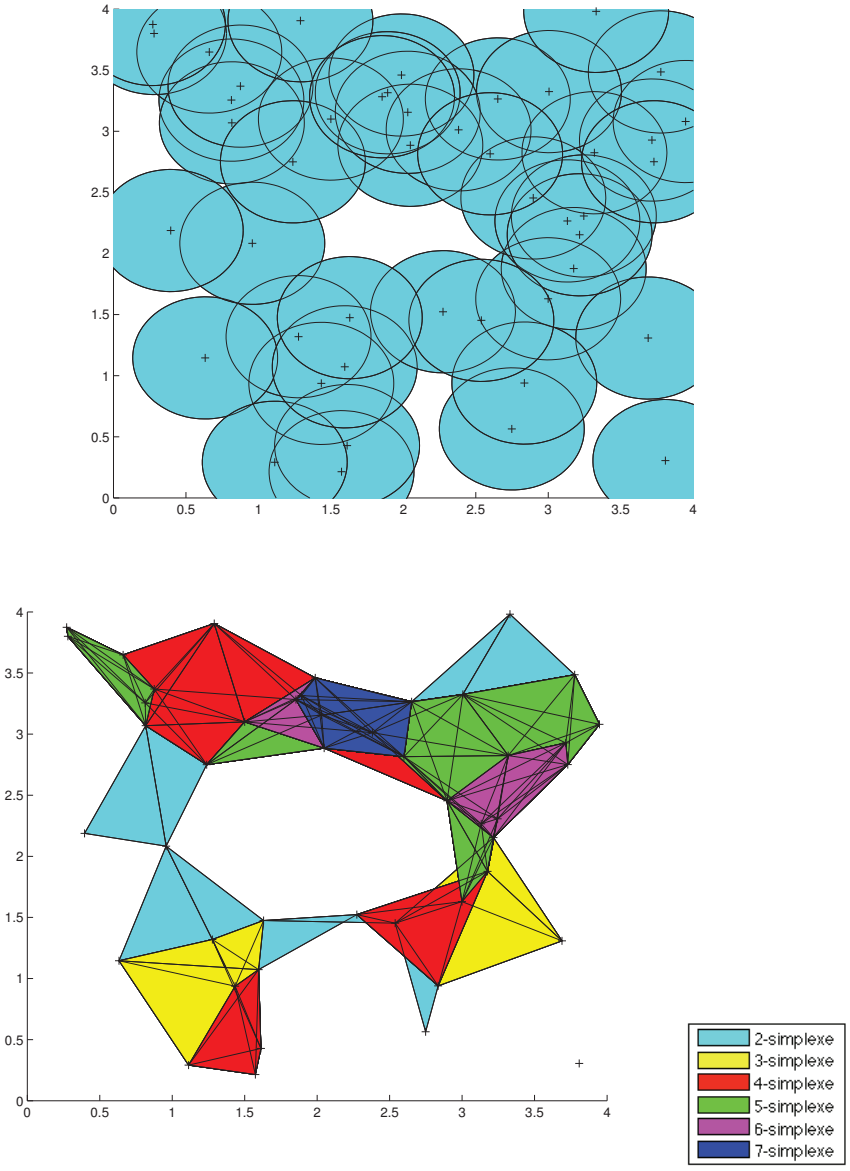


Figure 4: Un réseau de capteurs sans fil et son complexe de Rips-Vietoris associé.

En général, contrairement au complexe de Čech, le complexe de Rips-Vietoris n'est

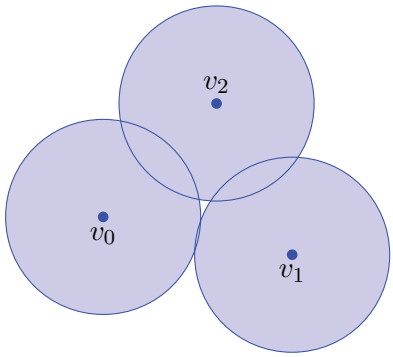
pas topologiquement équivalent à la couverture d'un domaine. Cependant, il existe des relations entre ces deux complexes :

Lemma 1. *Soit ω un ensemble fini de points sur $\mathbb{R}^2$, et ϵ un réel positif. On a*

$$\mathcal{R}_{\sqrt{3}\epsilon}(\omega) \subset \mathcal{C}_\epsilon(\omega) \subset \mathcal{R}_{2\epsilon}(\omega).$$

Une démonstration de ce lemme peut être lue dans [25].

On peut aisément vérifier que le complexe de Čech $\mathcal{C}_\epsilon(\omega)$ et le complexe de Rips-Vietoris $\mathcal{R}_{2\epsilon}(\omega)$ diffèrent seulement sur des triangles spécifiques. Par exemple, si on considère l'ensemble de trois sommets avec leurs disques de communication de rayon ϵ :



Alors leur représentation par le complexe de Čech sera trois 1-simplexes et aucun 2-simplexes :

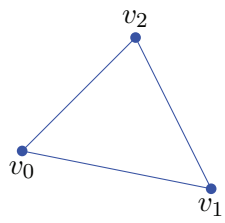


Figure 5: Complexe de Čec $\mathcal{C}_\epsilon(\omega)$

Cependant, comme le complexe de Rips-Vietoris est entièremnt construit à partir de la description du graphe, il y a un 2-simplexe dès qu'il y a trois 1-simplexes reliant trois 0-simplexes :

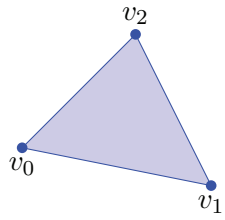


Figure 6: Complexe de Rips-Vietoris $\mathcal{R}_{2\epsilon}(\omega)$

L'absence de 2-simplexe du complexe de Čech peut être observée dans le complexe de Rips-Vietoris $\mathcal{R}_{\sqrt{3}\epsilon}(\omega)$:

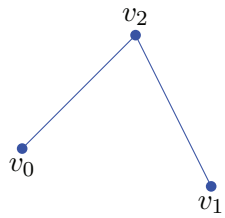


Figure 7: Complexe de Rips-Vietoris $\mathcal{R}_{\sqrt{3}\epsilon}(\omega)$

Pour les complexes simpliciaux de couverture, que sont le complexe de Čech et le complexe de Rips-Vietoris, les nombres de Betti représentent le nombre de trous de k dimensions. En effet, le k-ième nombre de Betti β_k compte le nombre de cycles de k-simplexes qui ne sont pas remplis par des $(k+1)$-simplexes. Par exemple, β_0 est le nombre de trous 0-dimensionels, c'est-à-dire le nombre de composantes connexes. Et β_1 est le nombre de trous dans le plan, puis β_2 est le nombre de vides à l'intérieur d'une surface 3-D donnée. En dimension d, le k-ième nombre de Betti pour $k \geq d$ n'a aucun sens géométrique.

0.3 Algorithme de réduction pour complexes simpliciaux

0.3.1 Introduction

Les capteurs sont des systèmes autonomes : ils ne sont pas branchés électriquement ni reliés entre eux. Leur autonomie est donc un problème majeur, et l'économie d'énergie un point crucial dans la gestion des réseaux de capteurs sans fil. Il existe même plusieurs définitions pour la durée de vie d'un réseau de capteurs sans fil, comme expliqué dans [29]. Notre approche de la durée de vie du réseaux est plutôt naïve : nous considèrons une image statique du réseau. Pour contrebalancer la sensibilité d'un réseau de capteurs aux trous de couverture ou aux composantes déconnectées, une solution bien connue est de déployer un nombre excessif de capteurs. En utilisant plus de capteurs que nécessaire pour couvrir un domaine ou connecter un réseau, on assure une couverture redondante et l'entière connectivité. Cependant, cette solution a un coût en matériel, en maintenance, aussi bien qu'en autonomie. Ainsi, une approche naïve pour améliorer la durée de vie d'un réseau de capteurs sans fil et réduire la consommation d'énergie serait donc logiquement de mettre certains capteurs en veille, comme ils sont en surnombre. Cependant, si cela est fait au hasard, cela peut modifier la topologie du réseau en créant un trou de couverture, ou en cassant la connectivité.

C'est pourquoi nous proposons ici un algorithme qui retourne l'ensemble des capteurs qui peuvent être mis en veille sans modification de la topologie du réseau. Etant donné un complexe simplicial, notre algorithme enlève les sommets selon un ordre optimisé, tout en gardant la topologie du complexe intacte. Un exemple d'une exécution de l'algorithme est donnée en Figure 8.

Nous montrons que l'algorithme atteint un équilibre de Nash : chaque sommet du complexe simplicial final est nécessaire au maintien de l'homologie. Cela signifie que l'agorithme atteint un optimum local. Nous évaluons une borne inférieur et une borne supérieure pour

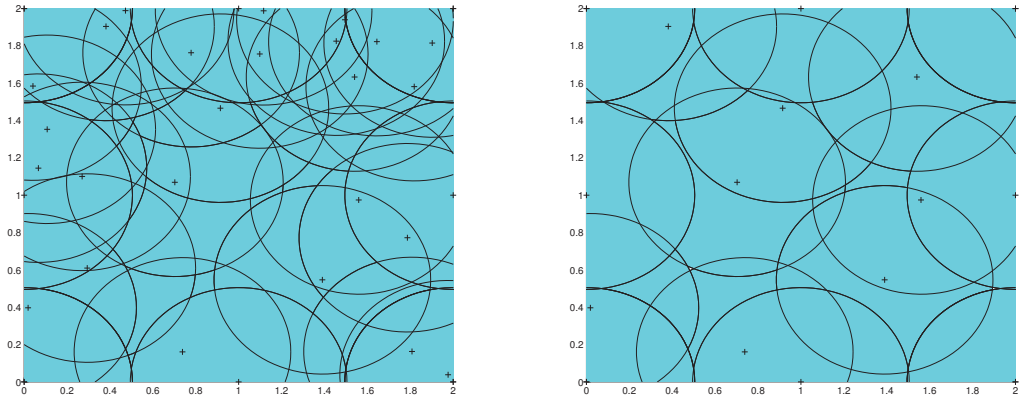


Figure 8: Un réseau de capteurs avant et après l'exécution de l'algorithme de couverture.

le nombre de sommets retirés. La complexité en moyenne de l'algorithme est analysée pour deux types de complexes simpliciaux aléatoires : le complexe de cliques d' Erdös-Rényi, et les complexes géométriques basés sur un processus de Poisson. Nous montrons que cette complexité est polynomiale dans le premier cas, et exponentielle dans le deuxième. Nous donnons aussi des caractéristiques du complexe simplicial final pour le cas de l'application à la couverture des réseaux de capteurs sans fil.

C'est le premier algorithme de réduction basé sur la représentation par complexes simlpiciaux, utilisant l'homologie, qui vise à économiser de l'énergie des les réseaux de capteurs sans fil. Une approche usuelle à la gestion de l'énergie dans les réseaux est l'utilisation du graphe de connectivité, comme dans le problème de l'ensemble dominant [43]. Cependant, les graphes sont des objets à deux dimensions. Un sommet connait ses voisins, mais il n'y a pas de repr/'esentation des intéractions entre ceux-ci. Ainsi, il n'y a pas de notion de couverture dans les graphes. Les complexes simpliciaux permettent de représenter ces relations d'ordre supérieur, et sont donc plus adaptés à la représentation des réseaux de capteurs. Plusieurs travaux peuvent paraitre relier d'un premier abord à notre travail, mais ils ne font pas exactement la même chose. Dans [30, 50], les auteurs utilisent la réduction des complexes de chaine pour calculer l'homologie, réduisant ainsi le domaine couvert, ce qui le rend inapplicable au problème de couverture. La réduction de complexes témoins, qui est la réduction à un nombre donné de sommets, est utilisée dans [23] pour calculer des invariants topologiques. Dans ce dernier papier, comme dans les articles de réduction des complexes de chaine, les auteurs utilisent la réduction pour calculer l'homologie, alors que nous utilisons l'homologie pour réduire de manière optimale un complexe simplicial. Finalement, les auteurs de [17] propose une approche basée sur la théorie des jeux pour la gestion de l'énergie, où ils définissent une fonction de couverture. Cependant, cette méthode nécessite des informations de localisation précises, ainsi que la connaissance de la couverture. De plus, les auteurs identifient des solutions sous-optimales, qui ne garantissent pas une couverture intacte.

Le reste de cette section est organisée comme suit. La Section 0.3.2 est dévouée à la description de notre algorithme de réduction. Des résultats de simulation sont donnés dans la Section 0.3.3. Finalement, dans la Section 0.3.4, nous discutons des propriétés mathématiques de l'algorithme.

0.3.2 Algorithme

Dans cette section, nous présentons l'algorithme de réduction qui donne quels capteurs peuvent être mis en veille dans un réseau de capteurs sans modification de sa topologie. Dans cet algorithme on utilise l'homologie simpliciale pour représenter le réseau de capteurs sans fil sans information de localisation, et pour caractériser sa topologie. Mais on utilise aussi l'information venant de la représentation par complexes simpliciaux pour identifier les capteurs redondants, l'idée générale étant de retirer les capteurs appartenant aux simplexes les plus grands.

L'algorithme a besoin de deux entrées. Tout d'abord on a besoin du complexe simplicial abstrait entièrement décrit, c'est-à-dire avec tous les simplexes explicités. Avec seulement un complexe simplicial, la réduction optimale sans modification de l'homologie sera toujours de réduire le complexe à un unique sommet. C'est pourquoi on a aussi besoin en deuxième entrée d'une liste de sommets qui doivent être conservés par l'algorithme de réduction. On appelle ces sommets les sommets *critiques*. On enlève ensuite des sommets non critiques et leurs faces un par un du complexe simplicial sans modification de la topologie, i.e. sans modification des nombres de Betti. A la sortie, on obtient le complexe simplicial *final* et la liste des sommets retirés.

Il y a autant de groupes d'homologie non nuls, soit de nombres de Betti non nuls, que de tailles de simplexes dans le complexe simplicial abstrait. Ainsi on peut définir différents algorithms suivant le nombre de nombres de Betti qui doivent être inchangés. On note k_0 le nombre de nombres de Betti que l'algorithme prend en compte.

Dans le cas de l'application aux réseaux de capteurs sans fil, le complexe simplicial abstrait sera typiquement un complexe de Čech ou de Rips-Vietoris en deux dimensions. Les seuls nombres de Betti d'un complexe de Čech ou de Rips-Vietoris qui ont une interprétation géométrique sont β_0 et β_1 en deux dimensions. On considère donc deux algorithmes :

- Le premier algorithme, appelé l'algorithme de connectivité, maintient seulement la connectivité du complexe simplicial, et ne prend pas en compte la couverture. La topologie du complexe est alors spécifiée par le nombre de composantes connectées β_0 et $k_0 = 1$.

- Le deuxième algorithme, algorithme de couverture, prend en compte à la fois la connectivité et la couverture, i.e. il conserve le nombre de composantes connexes β_0 et le nombre de trous de couverture β_1, et $k_0 = 2$. Cet algorithme est le cas général en deux dimensions.

The list of critical vertices can be viewed as a list of active sensors that have to stay connected as they are, or extremity sensors of a line-shaped network for the connectivity algorithm. In the coverage algorithm, the critical vertices will be the vertices lying on the boundary of the area that is to stay covered, that includes both the external boundary and the holes boundary. We need all the *boundary* vertices in order to not shrink the area, nor enlarge coverage holes. While the external boundary vertices are quite easy to discover: using the convex hull, or directly defined by the network manager; the hole boundary vertices are more tricky to obtain. the authors of [89] propose an algorithm in order to find them. But in the main application of our algorithm: power consumption reduction in wireless sensor networks, we consider that there are too many sensors to cover an area that we want to reduce the number: therefore we consider that there is no coverage hole. So the discovery of the hole boundary vertices is not a problem.

La liste des sommets critiques peut être vue comme une liste de capteurs actifs qui doivent rester connectés comme ils sont, ou des capteurs aux extrémités d'un réseau en ligne pour l'algorithme de connectivité. En ce qui concerne l'algorithme de couverture, les sommets critiques seront les capteurs déployés sur la frontière de la zone qui doit rester couverte, cela inclut à la fois la frontière externe et le contour des trous de couverture. On a besoin de tous les sommets de *bordure* afin de ne pas réduire la zone, ou augmenter les trous de couverture. Les sommets de la bordure externe sont assez facile à obtenir : en utilisant l'enveloppe convexe, ou même par définition de l'opérateur de réseau dnas le cas d'un réseau de capteurs. Les sommets en bordure de trous de couverture peuvent être assez complexes à localiser. Les auteurs de [89] propose un algorithme afin de les trouver. Dans l'application principale de notre algorithme : l'économie d'énergie dans les réseaux de capteurs sans fil, on considère qu'il y a des capteurs en surnombre dont on veut réduire le nombre : donc on considère qu'il n'y a pas de trou de couverture. Ainsi la recherche des sommets en bordure de trous n'est pas un problème.

On définit maintenant l'hypothèse de domaine entier pour l'application aux réseaux de capteurs, i.e. quand l'algorithme de réduction est appliqué à un complexe de Čech ou de Rips-Vietoris en moins de deux dimensions :

Definition 9 (Hypothèse de domaine entier)**.** *En dimension $d \leq 2$, on définit si $k_0 = d$, pour un complexe de Čech ou de Rips-Vietoris avec $\beta_0 = 1$ et si $k_0 = 2$, $\beta_1 = 0$, l'hypothèse de domaine entier qui est satisfaite lorsque tous les sommets du comlexe simplicial abstrait sont dans le même domaine géométrique définie par les sommets critiques.*

Pour l'algorithme de connectivité en une dimension, cela signifie que les sommets critiques sont deux sommets extrêmes et les autres sommets doivent être sur le même chemin reliant les deux sommets critiques.

Pour l'algorithme de couverture, cela signifie que les sommets critiques sont les sommets de bordure et tous les autres sommets sont à l'intérieur de l'aire définie par les sommets critiques.

On peut remarquer qu'il est toujours possible de satisfaire l'hypothèse de domaine entier en enlevant avant l'algorithme de réduction, tous les sommets qui ne sont pas dans le chemin défini par les sommets critiques pour l'algorithme de connectivité en une dimension, ou les sommets qui ne sont pas à l'intérieur de l'enveloppe convexe des sommets critiques en deux dimensions.

0.3.2.1 Degrés

La première étape de l'algorithme est le calcul d'un nombre, qu'on appelle *degré*, défini pout tout k_0-simplexe, où k_0 est le nombre de nombre de Betti à conserver. Afin de connecter des sommets, on a seulement besoin de 1-simplexes, pour couvrir un domaine, on a de la même manière seulement besoin de 2-simplexes. Ainsi les simplexes plus grands, i.e. les simplexes avec plus que $k_0 + 1$ sommets sont superflus pour le problème de connectivité ou de couverture pour les k_0 premiers nombres de Betti. On essaye de caractériser la superficialité des k_0-simplexes avec la définition suivante :

Definition 10. *Pour k_0 entier, le degré d'un k_0-simplexe $[v_0, v_1, \ldots, v_{k_0}]$ est la taille de sa plus grande coface :*

$$D[v_0, v_1, \ldots, v_{k_0}] = \max\{d \mid [v_0, v_1, \ldots, v_{k_0}] \subset d\text{-}simplexe\}.$$

Par définition on a $D[v_0, v_1, \ldots, v_{k_0}] \geq k_0$.

Pour la suite, on note $s_k(X)$ ou simplement s_k le nombre de k-simplexes du complexe simplicial X. On note aussi $D_1, \ldots, D_{s_{k_0}}$ les s_{k_0} degrés d'un complexe simplicial. Ils sont calculés selon l'Algorithme 1.

Algorithm 1 Calcul des degrés

for $i = 1 \to s_{k_0}$ **do**
  Soient $(v_0, \ldots, v_{k_0})$ les sommets du i-ème k_0-simplexe
  $k = k_0$
  while $(v_0, \ldots, v_{k_0})$ sont sommets d'un $(k+1)$-simplexe **do**
    $k = k + 1$
  end while
  $D_i = k$
end for
return $D_1, \ldots, D_{s_{k_0}}$

On peut voir un exemple de valeurs pour le degré de 2-simplexes dans la Figure 9 : quand un 2-simplexe est isolé son degré est 2, quand c'est la face d'un tétrahèdre il devient 3.

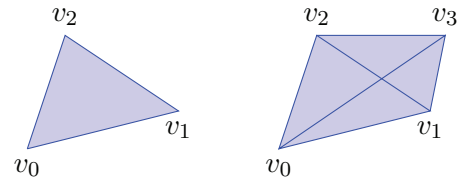


Figure 9: Exemple de valeurs de degrés de 2-simplexes.

0.3.2.2 Indices

Le but de l'algorithme est de retirer des sommets, et non pas des k_0-simplexes, donc on doit faire descendre l'information de superficialité des k_0-simplexes au niveau des sommets, à l'aide d'un *indice*. On considère un sommet sensible si son retrait entraine un changement dans les nombres de Betti du complexe. Un sommet est aussi sensible que son k_0-simplexe le plus sensible. Donc l'indice d'un sommet est le minimum des degrés des k_0-simplexes dont il est sommet :

Definition 11. *L'indice d'un sommet v est le minimum des degrés des k_0-simplexes dont il ets sommet :*

$$I[v] = \min\{D[v_0, v_1, \ldots, v_{k_0}] \mid v \in [v_0, v_1, \ldots, v_{k_0}]\},$$

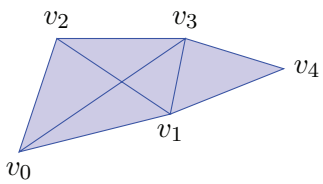
Si un sommet v n'est le sommet d'aucun k_0-simplexe alors $I[v] = 0$.

Soient $v_1, v_2, \ldots, v_{s_0}$ les sommets du complexe simplicial, le calcul des s_0 indices est fait de la manière décrite dans l'Algorithme 2.

Algorithm 2 Calcul des indices

```
for i = 1 → s_0 do
  I[v_i] = 0
  for j = 1 → s_{k_0} do
    if v_i est sommet du j-ième k_0-simplexe then
      if I[v_i] == 0 then
        I[v_i] = D_j
      else
        I[v_i] = min{I[v_i], D_j}
      end if
    end if
  end for
end for
return I[v_1], ..., I[v_{s_0}]
```

On peut voir dans la Figure 10 un exemple de valeur pour les indices des sommets d'un complexe simplicial. Les sommets d'un k_0-simplexe sont plus sensible que les sommets n'appartenant qu'à des simplexes plus grands.



$$D[v_0, v_1, v_2] = D[v_0, v_1, v_3] = D[v_0, v_2, v_3] = D[v_1, v_2, v_3] = 3$$
$$D[v_1, v_3, v_4] = 2$$
$$I[v_0] = I[v_2] = 3 \text{ and } I[v_1] = I[v_3] = I[v_4] = 2$$

Figure 10: Exemple de valeurs d'indices de sommets.

L'indice d'un sommet est ainsi un indicateur de la densité de sommets autour de lui : un indice de k_0 indique qu'au moins une de ses k_0-cofaces n'est pas la face d'un simplexe plus grand. Tandis qu'un indice plus élevé indique que toutes ses k_0-cofaces sont les faces de simplexes plus grands. L'idée dénérale de l'algorithme est donc de retirer les sommets de plus grand indice.

Remark 1. *Un indice de* 0 *indique que le sommet n'appartient à aucun k_0-simplexe : c'est-à-dire que le sommet est isolé au k_0-ième degré. Pour $k_0 = 1$, cela signifie que le sommet est déconnecté de tous les autres. Pour $k_0 = 2$, le sommet est seulement relié aux autres sommets par des arêtes au plus, il ets donc dans un trou de couverture. Lorsque l'hypothèse de domaine entier est vérifiée, ces cas-là n'existent pas.*

0.3.2.3 Ordre optimisé

L'algorithme enlève maintenant les sommets du complexe simplicial initial suivant un ordre optimisé. On commence par calculer les k_0 premiers nombres de Betti en utilisant l'algèbre linéaire. Puis, les degrés de tous les k_0-simplexes et les indices de tous les sommets sont calculés comme expliqués dans la section précédente. Les sommets critiques de la liste donnée en entrée sont affectés d'un indice négatif afin de les identifier comme inenlevables. Les indices nous donnent ensuite un ordre pour le retrait successif des sommets : plus l'indice d'un sommet est grand, plus le sommet est spuperficiel pour l'homologie k_0-ième du complexe. Ainsi, les sommets de plus grand indice sont candidats au retrait : un est choisi aléatoirement. Le retrait d'un sommet entraîne le retrait de toutes ses faces.

A chaque retrait de sommet, on doit vérifier que l'homologie n'a pas été modifiée. On calcule les k_0 premiers nombres de Betti à l'aide des différentielles de carré nul à chaque retrait de sommet. Ce calcul est instantané vu que le complexe est déjà construit, et seules les matrices d'adjacence sont nécessaires. Si le retrait du sommet modifie l'homologie, le sommet est remis dans le complexe. De plus, son indice devient temporairement négatif pour ne pas que le sommet soit choisi au prochain tirage pour le retrait suivant. Les indices temporaires sont recalculées au prochain retrait effectif de sommet.

Sinon, si le retrait ne modifie pas l'homologie, celui-ci est confirmé. Les degrés modifiés des k0-simplexes et les indices des sommets sont recalculés. On peut remarquer que seulement les sommets d'indice maximum peuvent avoir leur indice modifié, comme expliqué dans le Lemme 2. De plus, afin d'améliorer la performance de l'algorithme il ets possible de seulement calculer les degrés impactés par le retrait. Il suffit de marquer les k-simplexes qui sont les plus grandes cofaces de k_0-simplexes. Et quand l'une d'elles disparait, le degré de ses k_0-faces peuvent être modifiés.

Lemma 2. *Quand un sommet d'indice $I_{\max}$ est retiré du complexe, seulement les sommets partageant un $I_{\max}$-simplexe avec celui-ci, et d'indice $I_{\max}$ peuvent avoir leur indice modifié.*

Proof Soit w le sommet retiré d'indice $I_{\max}$, et soit v un sommet quelconque du complexe simplicial.

Si v ne partage pas de simplexe avec w, aucun des degrés de ses k_0-simplexes ne sera modifié, par conséquent son indice ne sera pas modifié non plus.

Ainsi, on peut considérer que le plus grand simplexe commun de v et w est un k-simplexe, $k > 0$. Si $k < k_0$, alors le retrait de w et de ce k-simplexe n'a aucune conséquence sur l'indice de v par définition. Puis si $k_0 \leq k < I_{\max}$ alors w est d'indice $k < I_{\max}$, ce qui est absurde. On peut donc supposer que $k \geq I_{\max}$. Soit l'indice de v est strictement inférieur à $I_{\max}$ est vient d'un simplexe non partagé avec w, et n'est donc pas affecté par le retrait de w. Ou alors, si l'indice de v est $I_{\max}$, il peut toujours venir d'un simplexe non partagé avec w, auquel cas il ne change pas. Ou si l'indice de v vient d'un $I_{\max}$-simplexe commun avec w, alors l'indice de v est modifié. C'est le seul cas où il l'est.

L'algorithme continue de retirer des sommets jusqu'à ce que tous les sommets restant soient inenlevables, atteignant ainsi un résultat optimal. Tous les sommets sont inenlevables quand tous les indices sont strictements inférieurs à k_0. Par définition d'un indice, cela veut dire que tous les indices sont soit nuls, soit négatifs (temporairement ou non).

Certaines choses peuvent être améliorées avec l'hypothèse de domaine entier dans le cas de l'application aux réseaux de capteurs sans fil. Tout d'abord, la condition d'arrêt peut être améliorée à ce que $I_{\max}$ soit inférieur ou égal à k_0 :

Lemma 3. *Sous l'hypothèse de domaine entier, l'algorithme peut s'arrêter lorsque tous les indices sont inférieurs ou égaux à k_0.*

Proof On suppose que les données satisfont l'hypothèse de domaine entier. Soit v un sommet d'indice $I(v) = k_0$, cela signifie qu'au moins une de ses k_0-cofaces n'a pas de $(k_0 + 1)$-coface. Le retrait de ce sommet entrainerait en particulier le retrait de ce k_0-simplexe. Comme l'algorithme doit conserver l'homologie que tout le domaine géométrique défini par les sommets critiques sans le réduire, ce retrait entrainerait la création d'un trou k_0-dimensionnel, et donc une incrémentation de β_{k_0-1}.

Pour l'algorithme de connectivité, le retrait d'une arête qui n'est pas le côté d'un triangle entraine une déconnexion dans le chemin reliant les deux sommets critiques. Pour l'algorithme de couverture, le retrait d'un triangle qui n'est pas une face de tétrahèdre entraine la création d'un trou de couverture dans la zone entourée par les sommets critiques.

Puis sous l'hypothèse de domaine entier dans le cas de l'application aux réseaux de capteurs sans fil, tous les sommets inenlevables temporairement (d'indice négatif) le sont définitivement :

Lemma 4. *Sous l'hypothèse de domaine entier, quand le retrait d'un sommet modifie l'homologie du complexe, elle la modifiera toujours.*

Proof Sous l'hypothèse de domaine entier, la distantce entre les sommets critiques ne peut pas diminuer, ni l'aire de la zone entre eux réduire. Comme la taille du domaine n'est pas modifiée, comme dans le peuve du Lemme 3, le retrait d'un sommer qui a entrainé une modification d'un nombre de Betti entrainera toujours le même changement.

Remark 2. *On peut omettre l'ordre optimisé des sommets, et seulement retirer tous les sommets d'indice strictement supérieur à k_0 quand l'homologie n'est pas modifiée. Le calcul des degrés est alors limité au choix plus grand que k_0 ou non. En faisant cela, on perd l'ordre optimisé pour le retrait des sommets. Dans ce cas, l'algorithme peut alors être distribué, chaque noeud peut faire tourner un algorithme décentralisé avec pour seule information : ses voisins et leurs relations entre eux. Cela a été fait dans [91].*

On donne dans l'Algorithme 3 l'algorithme entier pour la conservation des k_0 premiers nombres de Betti.

Algorithm 3 Algorithme de réduction

Require: Complexe simplicial X, liste L_C des sommets critiques.

```
Calcul de β_0(X), ..., β_{k_0-1}(X)
Calcul de D_1(X), ..., D_{s_{k_0}}(X)
Calcul de I[v_1(X)], ..., I[v_{s_0}(X)]
for all v ∈ L_C do
  I[v] = −1
end for
I_max = max{I[v_1(X)], ..., I[v_{s_0}(X)]}
while I_max ≥ k_0 do
  Tirage de w sommet d'indice I_max
  X' = X\{w}
  Calcul de β_0(X'), ..., β_{k_0-1}(X')
  if β_i(X') ≠ β_i(X) pour un i = 0, ..., k_0 − 1 then
    I[w] = −1
  else
    Calcul de D_1(X'), ..., D_{s'_{k_0}}(X')
    for i = 1 → s'_0 do
      if I[v_i(X')] == I_max then
        Calcul de I[v_i(X')]
      end if
      if I[v_i(X')] == −1 && v_i ∉ L_C then
        Calcul de I[v_i(X')]
      end if
    end for
    X = X'
  end if
  I_max = max{I[v_1(X)], ..., I[v_{s_0}(X)]}
end while
return X
```

0.3.3 Simulations

Les simulations présentées ici ont pour but d'illustrer l'algorithme. Les résultats de l'algorithme sont hautement dépendants des choix de paramètres. Pour l'algorithme de connectivité le pourcentage de sommets retirés est lié au fait les sommets critiques soient reliés entre eux sans intermédiaire ou non, et comment. Pour l'algorithme de couverture, le pourcentage de sommets retirés est lié au rapport entre le nombre initial de sommets dans le complexe et le nombre de sommets nécessaires pour couvrir la zone définie par les sommets critiques. On a simulé l'algorithme de réduction sur deux complexes différents.

Tout d'abord on considère le complexe d' Erdös-Rényi, complexe de cliques du graphe éponyme :

Definition 12 (Complexe d'Erdös-Rényi)**.** *Soit n un entier et p un réel dans $[0, 1]$, le complexe d'Erdös-Rényi de paramètres n et p, appelé $G(n, p)$, est un complexe simplicial abstrait de n sommets. Puis, chaque arête est incluse avec la probabilité p indépendamment des autres arêtes. Enfin un k-simplexe, avec $k \geq 2$, est inclus si et seulement si toutes ses faces le sont déjà.*

On peut voir dans la Figure 11 une réalisation de l'algorithme de réduction conservant le nombre de composantes connexes β_0 sur un complexe d' Erdös-Rényi de paramètre $n = 15$ et $p = 0.3$. Les sommets critiques sont choisis aléatoirement : un sommet est critique avec probabilité $p_c = 0.5$ indépendamment des autres sommets. Les sommets critiques sont entourés par des cercles, et les sommets non critiques gardés pour maintenir la connectivité entre les sommets critiques sont étoilés.

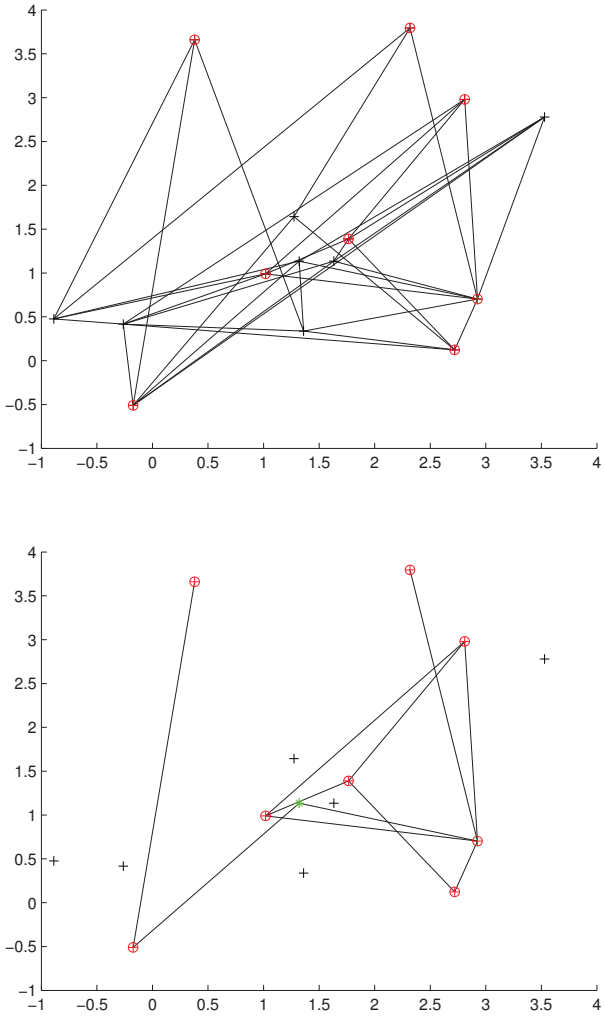


Figure 11: Complexe d'Erdös-Rényi avant et après l'algorithm de réduciton pour $k_0 = 1$.

Avec les paramètres $n = 60$ sommets, et $p = 0.2$, en moyenne sur 1000 configurations, avec une unique composante connexe, et p_c prenant des valeurs entre 0.1 et 0.5 (200 configurations par valeur), l'algorithme a retiré 98% des sommets non critiques :

p_c	Pourcentage de sommets retirés
0.1	94.96%
0.2	97.14%
0.3	98.59%
0.4	99.43%
0.5	99.87%

Puis on choisit d'illustrer le cas de l'application aux réseaux de capteurs sans fil avec un complexe de Rips-Vietoris en deux dimensions. On simule l'ensemble des sommets avec un processus de Poisson :

Definition 13. *Un processus de Poisson ω d'intensité λ sur un Borelien X est défini par :*

i) Pour tout $A \in \mathcal{B}(X)$, le nombre de points dans A, $\omega(A)$, est une variable aléatoire suivant une loi de Poisson de paramètre $\lambda S(A)$, $\Pr(\omega(A) = k) = e^{\lambda S(A)} \frac{(\lambda S(A))^k}{k!}$.

ii) Pour $A, A' \in \mathcal{B}(X)$ disjoints, les variables aléatoires $\omega(A)$ et $\omega(A')$ sont indépendantes.

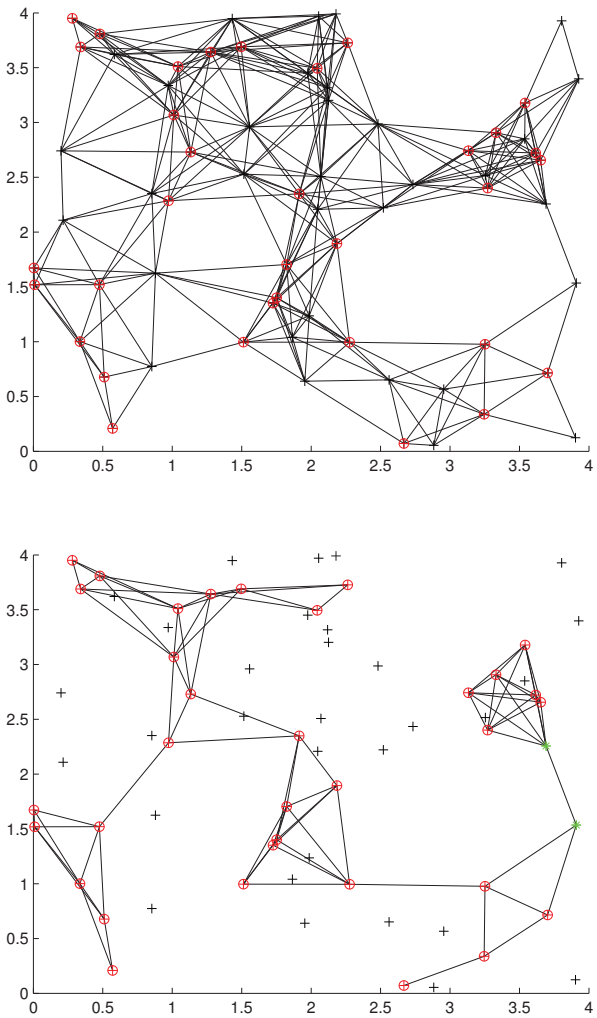


Figure 12: Un complexe de Rips-Vietoris avant et après l'algorithme de connectivité.

On peut voir dans la Figure 12 une réalisation de l'algorithme de connectivité sur un complexe de Rips-Vietoris de paramètre $\epsilon = 1$ basé sur un processus de Poisson d'intensité $\lambda = 4$ sur un carré de côté 4, avec des sommets critiques aléatoires. Un sommet est choisi critique avec probabilité $p_c = 0.5$ indépendamment des autres sommets.

Pour cette configuration de paramètres avec une composante connexe, en moyenne sur 1000 exécutions, l'algorithme de connectivité a retiré 96.01% des sommets non critiques.

On peut voir dans la Figure 13 une réalisation de l'algorithme de couverture sur un complexe de Rips-Vietoris de paramètre $\epsilon = 1$ basé sur un processus de Poisson d'intensité $\lambda = 4.2$ sur un carré de côté $a = 2$, avec une frontière fixe de sommets sur le périmètre du carré. Les sommets critiques sont ceux de la frontière, satisfaisant ainsi l'hypothèse de domaine entier pour l'algorithme de couverture. Ils sont encerclés sur la figure.

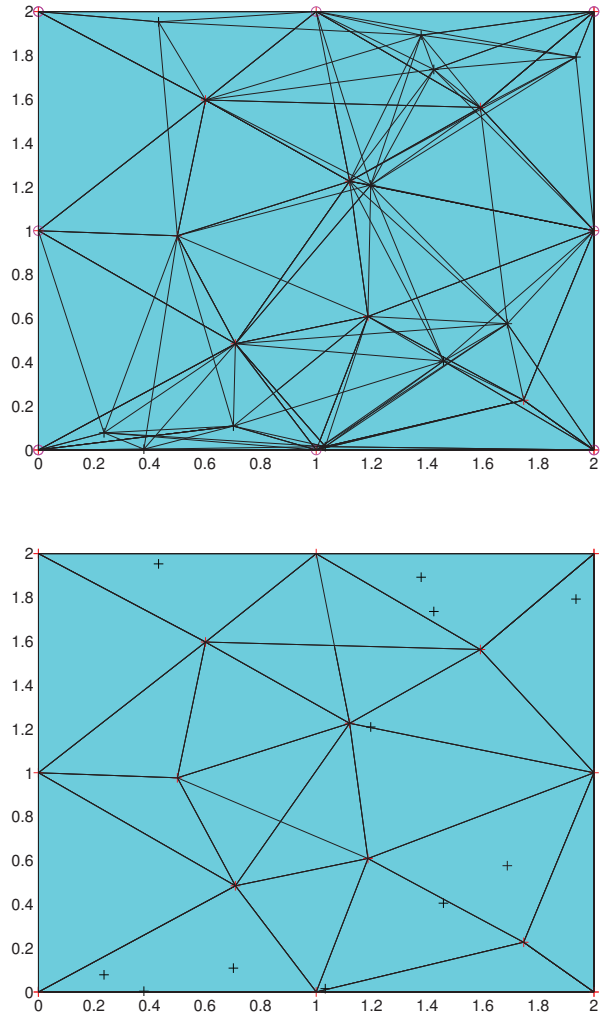


Figure 13: Un complexe de Rips-Vietoris avant et après l'algorithme de couverture.

Pour assurer la couverture sans trous, on a pris les paramètres suivants : $\epsilon = 1$, $\lambda = 5.1$ et $a = 2$, en moyenne sur 1000 configurations, l'algorithme de couverture a retiré 69.35% des sommets non critiques.

0.3.4 Propriétés mathématiques

La première propriété de notre algorithm est que la solution atteinte est optimale. Il est possible que ce ne soit pas la meilleure solution si il y a plusieurs optima, mais c'est un optimum local. En vocabulaire de théorie des jeux, cela veut dire que l'algorithme atteint un équilibre de Nash :

Theorem 1 (Equilibre de Nash)**.** *L'algorithme de réduction atteint un équilibre de Nash, défini dans [17] : chaque sommet du complexe simplicial final est nécessaire au maintien de son homologie.*

Proof Dans le complexe simplicial final, chaque sommet est d'indice strictement inférieur à k_0 dans le cas général. Par la définition des indices, on différencie alors deux types de sommets, ceux d'indice -1 et ceux d'indice 0.

Premièrement, les indices négatifs sont attribués aux sommets inenlevables. Soit un sommet a un indice négatif si c'est un sommet critique, auquel cas il doit rester dans le complexe par définition. Soit un sommet a un indice négatif car son retrait entraine un changement dans les nombres de Betti. Et il n'y a eu aucun retrait de sommets depuis, donc aucun changement dans le complexe simplicial, ainsi ce fait est toujours vrai.

Puis, un sommet d'indice nul est un sommet isolé. S'il est isolé au k_0-ième degré. Son retrait entraine une diminution de β_{k_0}. Par exemple, le retrait d'un sommet déconnecté décrémenterait β_0. Le retrait d'un sommet dans un trou entrainerait la réunion de deux ou plus de trous.

Finalement, si les données vérifient l'hypothèse de domaine entier, la preuve du Lemme 3 montre que l'algorithme atteint toujours un équilibre de Nash.

Dans un deuxième temps, nous avons trouvé une borne inférieure et une borne supérieure au nombre de sommets retirés par l'algorithme. Le nombre de sommets retirés est au moins un sommet par valeur d'indice, et au plus tous les sommets d'indices non minimum au début de l'algorithme :

Theorem 2 (Bornes supérieure et inférieure)**.** *Soit E_k l'ensemble des sommets d'indice k avant l'algorithme. Le nombre de sommets retirés M est borné par :*

$$\sum_{k=k_0+1}^{I_{\max}} \mathbb{1}_{[E_k \neq \emptyset]} \leq M \leq \sum_{k=k_0+1}^{I_{\max}} |E_k|.$$

où $|E_k|$ est le cardinal de E_k.

Proof On commence par prouver la borne supérieure. Le nombre maximum de sommets qui peuvent être retiréspar l'algorithme est le nombre de sommets dont l'indice est strictement supérieur à k_0. C'est une borne supérieure optimale atteinte dans le cas suivant :

Soit un k-simplexe, avec $k > k_0$, l'ensemble du complexe à réduire. , soit n_C le nombre de sommets critiques, alors $n_C \leq k + 1$. Les n_C sommets critiques ont un indice négatif, les $k + 1 - n_C$ autres sommets ont un indice de k, et ils sont tous retirés.

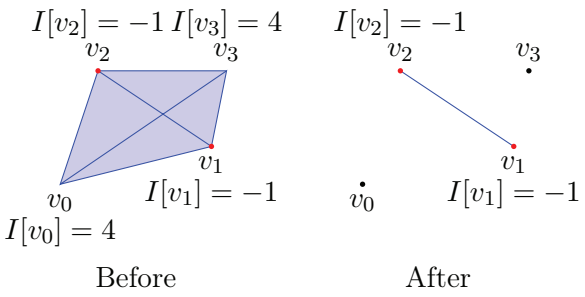


Figure 14: Exemple de ce cas avec $k = 3$ et $n_C = 2$, les deux sommets v_0 et v_3 sont retirés par l'algorithme.

Pour la borne inférieure, on a vu dans le Lemme 2 que le retrait d'un sommet d'indice $I_{\max}$ peut seulement modifier les indices des sommets d'indices $I_{\max}$. Dans le pire cas, le retrait diminue tous les indices qui étaient à $I_{\max}$, et la valeur de $I_{\max}$ change, pas nécessairement pour $I_{\max} - 1$ suivant la répartition des sommets critiques. Ainsi, au moins un sommet par valeur d'indice peut être retiré. D'où le résultat.

La borne inférieure est atteinte dans le cas précédent si $n_C = k$.

Remark 3. *Comme vu dans la preuve, les deux bornes supérieure et inférieure sont atteintes dans le cas du graphe complet où tous les sommets sauf un peuvent être retirés.*

Puis, nous avons trouvé quelques caractéristiques sur le complexe simplicial final dans le cas d'una application aux réseaux de capteurs sans fil. En utilisant la d'efinition suivante pour un ensemble couvrant :

Definition 14 (Ensemble couvrant)**.** *Comme défini [43], un ensemble $S \subseteq V$ de sommets d'un graphe $G = (V, E)$ est un ensemble couvrant si chaque sommet $v \in V$ est soit un élément de S ou adjacent à un élément de S.*

Sous l'hypothèse de domaine entier, l'ensemble de sommets conservés par l'algorithme est un ensemble couvrant de l'ensemble de sommets initial :

Theorem 3. *En deux dimensions pour l'algorithme de couverture appliqué à un complexe de Čech ou de Rips-Vietoris sous l'hypothèse de domaine entier, l'ensemble de sommets conservés est un ensemble couvrant de l'ensemble de sommets du complexe simplicial initial.*

Proof Sous l'hypothèse de domaine entier, les sommets initiaux sont tous dans le domaine géométrique défini par les sommets critiques. Pour l'algorithme de couverture, cela signifie que les sommets initiaux sont tous dans la zone définie par les sommets critiques. L'homologie du complexe n'est pas modifiée par l'algorithme, donc il n'y a aucun trou de couverture dans le complexe final. La zone est toujours couverte. Ainsi, chaque point de la zone est à l'intérieur d'un 2-simplexe. C'est vrai pour chaque sommet du complexe initial, qui ets alors adjacent à trois sommets restants.

Remark 4. *Dans le cas de l'application aux réseaux de capteurs sans fil, tous les capteurs en veille sont à un saut d'un capteur éveillé.*

Finalement, nous nous intéressons à la complexité de notre algorithme. Si on considère le nombre de sommets $n = s_0$ comme paramètre, alors la complexité de l'implémentation du complexe simplicial à n sommets est potentiellement exponentielle. En effet, le nombre de simplexes dans un complexe à n sommets est majoré par $2^n - 1$. Donc la complexité de l'implémentation des données est majorée par $O(2^n)$ par rapport au nombre de sommets n.

Theorem 4 (Complexité)**.** *La complexité de l'algorithme qui conserve k_0 nombre de Betti sur le complexe simplicial de s_k k-simplexes et $n = s_0$ sommets, est majorée par :*

$$n^2 s_{k_0} + (n + s_{k_0}) \sum_{k=0}^{C-1} s_k$$

avec C le cardinal de clique maximal du graphe sous-jacent.

Proof Soit $C - 1$ la taille du simplexe le plus grand dans le complexe initial, C étant connu comme le cardinal de clique maximum du graphe sous-jacent. Pour le calcul des degrés de tous les k_0-simplexes, l'algorithme parcourt au plus tous les k-simplexes pour $k_0 < k \leq C-1$ pour vérifier si le k_0-simplexe en est une face. Comme il y a s_k k-simplexes, le calcul des degrés est de complexité majorée par $s_{k_0} \sum_{k=k_0+1}^{C-1} s_k$.

Pour le calcul des indices, l'algorithme parcourt pour chacun des n sommets, ses k_0-cofaces, ce qui est au plus tous les k_0-simplexes. La complexité du calcul des indices a donc pour majorant ns_{k_0}.

Ces calculs sont faits au dbut de l'algorithme. Puis à chaque retrait de sommet, ce qui arrive au plus n fois :

- Les simplexes des sommets retirés sont détruits : complexité majorée par $\sum_{k=0}^{C-1} s_k$.
- Les nombres de Betti sont recalculés via les matrices d'adjacence qui existent déjà.
- Les degrés modifi/'es sont recalculés automatiquement avec la destruction des simplexes.
- Les au plus $n-1$ indices modifiés sont recalculés: complexité majorée par $(n-1)s_{k_0}$.

Corollary 5. *Quand n tend vers l'infini, la complexité de l'algorithme est $O(n^{k_0+1}2^n)$.*

Proof C'est une conséquence directe du Théorème 4 sachant que le nombre de k-simplexes s_k peut être majoré par $\binom{n}{k+1}$, k_0 est typiquement petit devant n (1 ou 2 en deux dimensions), et C est majoré par n.

Remark 5. *Comme l'implémentation des données est de complexité $O(2^n)$, la complexité de l'algorithme est polynomiale devant la complexité des données.*

On peut raffiner ces résultats sur la complexité de l'algorithme pour quelques complexes simpliciaux spécifiques.

Corollary 6. *Pour l'algorithme appliqué à un complexe de Čech $\mathcal{C}_\epsilon(\omega)$, défini avec la norme infinie, basé sur un processus de Poisson d'intensité λ sur un tore de côté a en dimension 2. La complexité de l'algorithme est majorée par $O((1 + (\frac{2\epsilon}{a})^2)^n)$ en moyenne lorsque n tend vers l'infini.*

Proof Selon [26], on a si $s_0 = n$:

$$\begin{aligned}\mathbf{E}\left[s_{k-1}\right] &= \binom{n}{k} k^2 \left(\frac{2\epsilon}{a}\right)^{2(k-1)} \text{ and,} \\ \mathrm{Cov}\left[s_{k-1}, s_{l-1}\right] &= \sum_{i=1}^{k\wedge l} \binom{n}{u_i}\binom{u_i}{k}\binom{k}{i}\left(u_i + 2\frac{(k-i)(l-i)}{i+1}\right)^2 \left(\frac{2\epsilon}{a}\right)^{2(u_i-1)},\end{aligned}$$

avec $u_i = k + l - i$.

L'utiliser dans la formule du Théorème 4 donne le résultat.

Corollary 7. *La complexité de l'algorithme appliqué à un complexe d'Erdös-Rényi basé sur le graphe $G(n,p)$ est de l'ordre de $O(n^{2(k_0+2)})$ en moyenne quand n tend vers l'infini.*

Proof Dans le complexe d'Erdös-Rényi,les espérances des nombres de k-simpleces sont données dans [14] :

$$\begin{aligned}\mathbf{E}\left[s_{k-1}\right] &= \binom{n}{k} p^{\binom{k}{2}}, \\ \mathbf{E}\left[s_{k-1}s_{l-1}\right] &= \sum_{i=0}^{k\wedge l} \binom{n}{k}\binom{k}{i}\binom{n-k}{l-i} p^{\binom{k}{2}+\binom{l}{2}-\binom{i}{2}}.\end{aligned}$$

Le résultat vient alors de la formule de la complexité du Théorème 4.

0.4 Autres résultats

Dans cette section, nous résumons les autres résultats présentés dans ce manuscrit.

0.4.1 Nombre de clique

En cherchant la complexité de notre algorithme de réduction, on est réduit à calculer le comportement de la taille du simplexe le plus grand dans un complexe simplicial aléatoire. En théorie des graphes, cette caractéristique est connue sous le nom de cardinal de la clique maximum. Le cardinal de clique maximum C est concrétement la taille de la plus grande clique dans un graphe. Dans le Chapitre 4, on trouve le comportement asymptotique presque sûr du cardinal de clique maximum pour le graphe sous-jacent au complexe de Rips-Vietoris sur un processus ponctuel binomial quand le nombre de points tend vers l'infini. Ce graphe est le graphe géométrique aléatoire de n sommets, tirés aléatoirement uniformément, incluant une arête entre deux sommets quand leur distance, prise avec la norme uniforme, est inférieure à un paramètre donné r. Les n sommets sont tirés uniformément sur le tore de côté a en dimension d. Le comportement du cardinal de clique maximum dépend du régime de percolation dans lequel se trouve le graphe. En posant $\theta = (\frac{r}{a})^d$, les régimes de eprcolation du graphe géométrique aléatoire sont définis selon les variations de $\frac{1}{n}$ devant θ. Dans le régime sous-critique, où $\theta = o(\frac{1}{n})$, nous donnons les intervalles de θ où C prend une valeur donnée asymptotiquement presque sûrement. Dans le régime critique, $\theta \sim \frac{1}{n}$, nous montrons que C croît légèrement moins vite que $\ln n$ asymptotiquement presque sûrement. Finalement, dans le régime sur-critique, $\frac{1}{n} = o(\theta)$, nous prouvons que C croît en $n\theta$ asymptotiquement presque sûrement. Nous nous intéressons aussi au comportement d'autres caractéristiques de graphe reliées : le nombre chromatique, le degré maximum, et la taille du stable maximum. Ce travail est le sujet de [28].

0.4.2 Applications aux réseaux cellulaires

Dans une deuxième partie, nous étendons les applications de la représentation par comlexe simplicial à de nouveaux types de réseaux sans fil : en particuler les réseaux cellulaires du futur.

0.4.2.1 Configuration

Dans le Chapitre 5, nous présentons un algorithme pour la planification automatique des fréquences, basé sur la représentation par complexes simpliciaux et utilisant l'algorithme de réduction présenté dans le Chapitre 3. Notre algorithme peut être appliqué aux réseaux sans fil aléatoires qui ne sont pas déployés selon un schéma régulier. L'algorithm minimise le nombre de fréquences planifiées tout en maximisant la couverture de chacune d'entre elles. Ainsi, un nombre minimum de ressources sont utilisées et ce de manière optimale. Cet algorithme de planification automatique de fréquences peut en particulier être utilisé pour la configuration automatique des réseaux cellulaires du futur. Nous présentons aussi sa performance et le comparons à l'algorithme glouton de coloriage.

0.4.2.2 Optimisation

Dans le Chapitre 6, nous proposons un algorithme pour l'optimisation automatique des réseaux cellulaires du futur pendant les heures creuses. Cet algorithme vient de modifications significatives de l'algorithme de réduction du Chapitre 3. En effet, nous proposons un algorithme de conservation d'énergie dont le but n'est plus de réduire le nombre de sommets sans modification de la topologie. L'algorithme de conservation d'énergie prend en compte non seulement la couverture, mais aussi le trafic, et est capable de satisfaire n'importe quelle demande de trafic. La performance de l'algorithme est aussi discutée et comparée à la solution optimal pas toujours atteignable.

0.4.2.3 Recouvrement

Dans le Chapitre 7, nous présentons un algorithme pour le recouvrement des réseaux sans fil après un désastre. On considère un réseau sans fil endommagé, avec des trous de couverture et/ou plusieurs composantes non connectés. Nous proposons un algorithme pour le recouvrement qiu répare le réseau. Il fournit la liste des positions où placer les nouveaux noeuds de réseau afin de colmater les trous de couverture, et recoller les composantes déconnectées. Afin de faire cela, on considère tout d'abord la représentation par complexe simplicial du réseau, puis l'algorithm ajoute de nouveaux sommets, en trop grand nombre, puis fait tourner l'algorithme de réduction du Chapitre 3 pour atteindre un résultat optimal. Une des nouveautés de ce travail réside dans la méthode proposée pour l'ajout de nouveaux sommets. Nous utilisons un processus ponctuel déterminantal : le processus de Ginibre qui crée une répulsion inhérente entre les sommets, et n'a jamais été simulé auparavent pour les réseaux sans fil. Nous comparons tout d'abord la méthode déterminantale avec des méthodes plus classqiues d'ajout de sommets. Puis nous comparons tout l'algorithme avec l'algorithme glouton pour le problème d'ensembles couvrants. Ce chapitre est l'objet de [87].

Chapter 1

Introduction

1.1 Motivation

Utilization of wireless sensor networks has notably grown in the past few years. Indeed they can be used in about every field where monitoring and observation play a role. These range from battlefield surveillance to target enumeration in agriculture, and include environmental monitoring. Moreover, miniaturization and decreasing costs of electronic circuits have allowed the creation of low-cost multifunctional sensors. Sensors are thus deployed in order to oversee or collect data about a given area. Therefore, the key factor for the quality of service of a wireless sensor network is its topology. In two dimensions, the topology of a wireless sensor network comprises both its connectivity and its coverage. For example entire connectivity would be needed if the network is set to oversee the crossings of a line of sensors, or the entries in a given area bounded by sensors. In a more general case, sensors need to be connected in order to convey data to a central data center since they do not have large memory capacity. Then, the coverage of a wireless sensor network defines the area being monitored; most of the time complete coverage of an area is required.

The first approach to wireless sensor network management is to discover its quality of service, that is consequently to discover its topology. To guarantee full knowledge of the topology, one solution is to deploy the sensors according to a regular pattern (hexagon, square grid, rhombus or equilateral triangle) as in [10]. However the target field does not always allow such a precise deployment. Furthermore, the topology may not be time-invariant: sensors could be destroyed, their batteries could die, or their communication could be disturbed by seasonal changes. Another approach is then to consider a random deployment that may create clusters of sensors, or on the contrary may leave holes of coverage. There is thus extensive research on the coverage problem for randomly deployed wireless sensor networks. Among popular methods, we can cite location-based methods, as in [33], which require exact location information for the sensors, and ranged-based methods, in [92], that compute coverage based on the distance between sensors. However, these two methods require geometrical information on the wireless sensor network which is not always available. Actually these methods encounter the same problems as deployment along a regular pattern: the target field may not allow precise location or distance measurement, and locations as well as distances may not be time-invariant.

That is why connectivity-based schemes seem of greater interest since they do not require such geometrical knowledge. In [37], Ghrist et al. introduced the so-called Vietoris-Rips complex, which is the clique complex built from the proximity graph between sensors, and determines the coverage via the homology of this complex. Coverage computation

via simplicial homology then boils down to simple linear algebra computations. It is used in [24], [67] and [94] as a tool for a network operator to evaluate the quality of a network. A distributed version of some of these algorithms is presented in [89] in order to detect coverage holes. Meanwhile, simplicial homology on random configurations has also attracted intensive mathematical research. Moments of simplicial homology characteristics can be obtained for a binomial point process, see [51], as well as for a Poisson point process, see [26].

1.2 Thesis contributions and outline

This dissertation is organized as follows. First in Chapter 2, we provide the simplicial homology background necessary for the understanding of this dissertation. We also included in this chapter the related work to establish some context.

In Chapter 3, we aim at reducing power consumption in wireless sensor networks by turning off supernumerary sensors. We use random simplicial complexes to provide an accurate and tractable representation of the topology of wireless sensor networks. Given a simplicial complex, we present an algorithm which reduces the number of its vertices, keeping its topology (i.e. connectivity, coverage) unchanged. We also give some simulation results for usual cases, especially coverage complexes simulating wireless sensor networks. We show that the algorithm reaches a Nash equilibrium. Moreover we find both a lower and an upper bounds for the number of vertices removed, the complexity of the algorithm, and the maximal order of the resulting complex for the coverage problem.

While computing the complexity of the reduction algorithm presented in Chapter 3, the computation comes down to the investigation of the behavior of the size of the largest simplex of the random geometric simplicial complex. Translating this to graph theory vocabulary, this means that we have to investigate the behavior of the clique number of the random geometric graph. In Chapter 4, we describe its asymptotic behavior when the number of vertices goes to infinity. This behavior depends on the percolation regimes of the graph, asymptotically almost sure behaviors are exhibited for the clique number in each regime. We also investigate the behavior of related graph characteristics: the chromatic number, the maximum vertex degree, and the independence number.

The simplicial complex representation is not only usable for wireless sensor networks, but also for any type of wireless networks where connectivity and coverage is a key factor. In particular, we choose to consider cellular networks, and in Chapters 5, 6 and 7, we apply the simplicial complex representation to cellular networks, and enhance our reduction algorithm for simplicial complexes in order to achieve new goals.

LTE cellular networks support some Self-Organizing Network (SON) features. In particular the 3GPP LTE Rel. 8 features the Automatic Neighbor Relation (ANR) detection: an eNode-B maintains a dynamic neighbor table. This feature strengthens even more the use of simplicial complex representation. In Chapter 5, we are interested in the implementation of SON self-configuration function for the planning of cells frequencies for future cellular networks. We propose and describe a frequency auto-planning algorithm which calls several instances of our reduction algorithm for simplicial complexes in order to minimize the number of needed frequencies while maximizing the coverage of each frequency.

In Chapter 6, our reduction algorithm for simplicial complexes is significantly modified for the self-optimization of cellular networks during off-peak hours. Indeed one of the self-optimization functions of SON is the ability to switch-off some of the base stations of a network during off-peak periods. However, our algorithm can not be directly applied

since the desired quality of service is not minimal coverage, as it can be the case in wireless sensor networks, but depends of the users traffic. The reduction algorithm is thus enhanced in order for a network to satisfy any required quality of service while using a minimum energy. We present our energy conservation algorithm and discuss its performance.

Then in Chapter 7, we provide an algorithm for the healing of any wireless network after a disaster. We consider a damaged wireless network presenting coverage holes, that we need to restore by patching the holes. We propose a disaster recovery algorithm which adds supernumerary vertices to cover the entire area, then run an improved reduction algorithm to reach an optimal result with a minimum number of added vertices. For the addition of new vertices, we propose the use of determinantal point processes which exhibit repulsion between vertices, and thus inherently facilitates the identification of coverage holes. We compare both this vertices addition method to other classic vertices addition methods, and the whole algorithm to the greedy algorithm for the set cover problem.

This dissertation is concluded in Chapter 8, where major contributions are reminded, and possible future work is discussed.

Finally I included two appendices on some related research work that does not directly fit within the scope of this dissertation. Malliavin calculus is applied in Appendix A in order to compute moments of characteristics of random geometric simplicial complexes based on Poisson point processes. In Appendix B, we propose a novel method for the simulation of Ginibre determinantal point processes with a given number of points in a compact set. Determinantal point processes are random point processes where the particules are not independent of each other, but repel each others.

Chapter 2

Related work and mathematical background

The work presented in this dissertation is at the intersection between different fields: we represent wireless sensor network using simplicial homology and stochastic geometry in order to write a reduction algorithm of which we derive properties and enhancements throughout the thesis. Therefore, we need to investigate in the first section of this chapter the research work done in the three domains: wireless sensor network, simplicial homology, and random configurations. In the second section, we introduce the definitions and theorems from simplicial homology theory which are necessary for the understanding of this dissertation.

2.1 Related work

2.1.1 Wireless sensor network representation

Many approaches exist for the representation of wireless sensor networks. Since the connectivity and the coverage of a wireless sensor network define its quality of service, the topology have a key role in the choice of representation. The first kind of traditional approach we can mention is the regular pattern display. With this approach the sensors are considered to be deployed along a hexagonal, square or triangular lattice. In [10], the authors propose an optimal deployment pattern in order to achieve coverage and redundant connectivity.

When sensors positions are not following a regular pattern, one has to consider a way to discover the sensor network topology. We can differentiate three different points of view. The first one is a location-based approach. The exact positions of the sensors are known and the Voronoi diagram is build. Considering the set of vertices of the sensor positions, for each vertex v, there is a corresponding region, called the Voronoi cell of vertex v, consisting of all points closer to v than to any other vertex. The Voronoi cells can thus be build with the bisectors between every two vertices. This approach is used in [33, 88] to compute coverage. Indeed one only has to check if every point of a Voronoi cell is covered by the sensor located in v, for every cells. However, the Voronoi diagram is not locally computable in general. But in [93] the authors propose a way for each node to compute its localized Voronoi cell.

The second approach to coverage discovery for a random deployment of sensors is range-based. That is to say that the exact positions of the sensors is not needed, whereas

the distance between neighboring sensors is used to identify coverage. In [92], the authors use the localized Voronoi cell computation to discover coverage as previously. Indeed the Voronoi diagram can be build by using distance information instead of position information. Another way to check the coverage of an area is for each sensor to check if the perimeter of its coverage disk is covered by its neighbors. This is used in [11] in a coverage verification algorithm. This is improved by the same author in [12]: considering the neighboring radius to be large enough compared to the coverage radius, each sensor can determine the relative locations of its neighbors.

In recent works, homology is used to discover the coverage of a wireless sensor network. This is what we can call a connectivity-based approach since only connectivity information between sensors is needed. Ghrist et al., [25, 37], introduced the Čech simplicial complex which gives an exact mathematical representation of a network topology via homology groups.

One can easily spot the differences between the approaches presented here. In the regular pattern approach, sensors need to be deployed along a given lattice. This is not always achievable in a real life scenario, that is why we prefer random deployments representations. Considering a random deployment for sensors, there is three ways to compute the coverage of the sensor network. Two of them require geometrical information: either exact position information for the first one, or distance between sensors for the second one. This geometrical information is not generally easy to obtain. The last approach doest not require geometrical information, but only connectivity information that sensors can send themselves. This deficit of information is counterbalanced by an increase in complexity. Yet, this method provides an exact and tractable mathematical description of the topology of a network.

2.1.2 Simplicial homology

Simplicial homology lies in the mathematical area of algebraic topology and has been studied prior to its application to wireless network representation. For a further general reading about simplicial homology we refer to the book of Hatcher [40].

As a pioneer work, Ghrist et al. introduced the use of simplicial homology to compute the coverage of a wireless sensor network in [37]. They first present the Čech complex, a combinatorial object defined in the next section, which characterizes the topology of a wireless sensor network. Since this object is complex to build, they also introduce the Vietoris-Rips complex, which is an approximation to the Čech complex.The Vietoris-Rips complex can be build with only the graph description and provide an exact connectivity description and an approximate coverage description. In [37], then in [24], they use a homology criterion for coverage.

There exist many ways to compute the homology of simplicial complexes. First we can cite persistent homology in which the simplicial complex is build progressively, simplex by simplex. An algorithm for computing simplicial homology is given in [94], moreover the persistence of geometric complexes, such as the Vietoris-Rips we are interested in, is studied in [18]. Persistent homology is applied to the computation of coverage of sensor networks in [25]. Homology can also be computed in a distributed way, as in [66]. In this article, the authors use the combinatorial Laplacians in order to detect the absence or presence of a coverage hole. Then, in [67], it is possible to distinguish between different coverage holes thanks to a gossip like algorithm. In [89], a distributed algorithm for hole detection in wireless sensor network is proposed, of which the accuracy is computed in [90].

Finally there is also some work combining both simplicial homology and random configurations. The expected topological properties of a Čech or a Vietoris-Rips complex are studied in [51]. Particularly, the author gives the asymptotic behavior for the expectation of the Betti numbers, which as seen in the next section, count the number of holes. In [26], the first moments of the number of complete components, called simplices, of the Čech and Vietoris-Rips complexes are computed.

2.1.3 Random configurations

In the above mentioned work of Kahle [51], the geometric complex is build on a binomial point process. In a binomial point process, a given number of vertices is sampled uniformly on a compact set. From the Čech and the Vietoris-Rips complex, we can recover the underlying graph. With a set of vertices following a binomial point process, this graph is called the random geometric graph. Random geometric graphs are thoroughly studied in Penrose's book [73], where it is shown that their behavior can be studied along percolation regimes. Particularly weak laws of large number and central limit theorems can be applied in the subcritical regime [75–77]. While strong las of large numbers are used in [8] to find some of the graph characteristics behaviors.

Then in [26], the random point process used is the Poisson point process. In this process, not only the vertices positions are random but also the number of vertices. Indeed, the number of vertices follow a Poisson distribution, and their positions are uniformly sampled on the compact set. Poisson point processes have very useful properties, as described in [78], which allow the authors of [26] to obtain their results. For example, a concentration inequality exists, and Malliavin calculus can be applied.

2.2 Simplicial homology background

2.2.1 Definitions

When representing a wireless sensor network, one's first idea will be a geometric graph, where sensors are represented by vertices, and an edge is drawn whenever two sensors can communicate with each other. However, the graph representation has some limitations; first of all there is no notion of coverage. Graphs can be generalized to more generic combinatorial objects known as simplicial complexes. While graphs model binary relations, simplicial complexes represent higher order relations. A simplicial complex is a combinatorial object made up of vertices, edges, triangles, tetrahedra, and their n-dimensional counterparts. Given a set of vertices V and an integer k, a k-simplex is an unordered subset of $k+1$ vertices $[v_0, v_1 \ldots, v_k]$ where $v_i \in V$ and $v_i \neq v_j$ for all $i \neq j$. Thus, a 0-simplex is a vertex, a 1-simplex an edge, a 2-simplex a triangle, a 3-simplex a tetrahedron, etc. See Figure 2.1 for instance.

Any subset of vertices included in the set of the $k+1$ vertices of a k-simplex is a face of this k-simplex. Thus, a k-simplex has exactly $k+1$ $(k-1)$-faces, which are $(k-1)$-simplices. For example, a tetrahedron has four 3-faces which are triangles. The inverse notion of face is coface: if a simplex S_1 is a face of a larger simplex S_2, then S_2 is a coface of S_1. A simplicial complex is a collection of simplices which is closed with respect to the inclusion of faces, i.e. all faces of a simplex are in the set of simplices, and whenever two simplices intersect, they do so on a common face. An abstract simplicial complex is a purely combinatorial description of the geometric simplicial complex and therefore does

0-simplex 1-simplex 2-simplex

Figure 2.1: Example of k-simplices.

not need the property of intersection of faces. For details about algebraic topology, we refer to [40].

For the remainder of this thesis, the "abstract" adjective of abstract simplicial complex might be dropped for a smoother reading. However all the simplicial complexes in this thesis are actually abstract simplicial complexes.

One can define an orientation for a simplex, where a change in the orientation corresponds to a change in the sign of the coefficient. For instance if one swaps vertices v_i and v_j:

$$[v_0, \dots, v_i, \dots, v_j, \dots, v_k] = -[v_0, \dots, v_j, \dots, v_i, \dots, v_k].$$

Then let us define vector spaces of k-simplices and a boundary map on them:

Definition 15. *Given an abstract simplicial complex X, for each integer k, $C_k(X)$ is the vector space spanned by the set of oriented k-simplices of X.*

Definition 16. *The boundary map ∂_k is defined to be the linear transformation $\partial_k : C_k \to C_{k-1}$ which acts on basis elements $[v_0, \dots, v_k]$ of C_k via*

$$\partial_k[v_0, \dots, v_k] = \sum_{i=0}^{k} (-1)^i [v_0, \dots, v_{i-1}, v_{i+1}, \dots, v_k].$$

The boundary map on any k-simplex, is the cycle of its $(k-1)$-faces. This map gives rise to a chain complex: a sequence of vector spaces and linear transformations.

$$\dots \xrightarrow{\partial_{k+2}} C_{k+1} \xrightarrow{\partial_{k+1}} C_k \xrightarrow{\partial_k} C_{k-1} \xrightarrow{\partial_{k-1}} \dots \xrightarrow{\partial_1} C_0 \xrightarrow{\partial_0} 0.$$

Finally let us define:

Definition 17. *The k-th boundary group of X is $B_k(X) = \operatorname{im} \partial_{k+1}$.*

Definition 18. *The k-th cycle group of X is $Z_k(X) = \ker \partial_k$.*

The boundary map applied to a cycle gives the cycle of this cycle which is zero as can be seen in Figure 2.2. Therefore a standard result asserts that for any integer k,

$$\partial_k \circ \partial_{k+1} = 0.$$

It follows that $B_k \subset Z_k$.

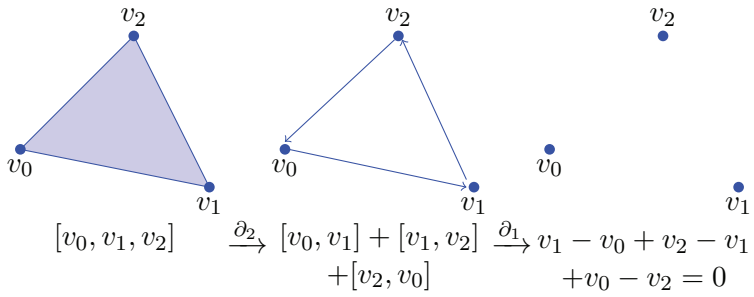


Figure 2.2: Application of the boundary map to a 2-simplex.

We are now able to define the k-th homology group and its dimension:

Definition 19. *The k-th homology group of X is the quotient vector space:*

$$H_k(X) = \frac{Z_k(X)}{B_k(X)}.$$

Definition 20. *The k-th Betti number of X is the dimension:*

$$\beta_k = \dim H_k = \dim Z_k - \dim B_k.$$

The Betti numbers are computed via the boundary maps matrices. These matrices are obtained directly from the simplicial complex description: the matrix of ∂_k is equivalent to the list of the $k-1$-faces of the k-simplices of the complex. Since we consider *abstract* simplicial complexes, it is not possible to use the incremental algorithm for the Betti numbers computation.

We can compute the Betti numbers in a simple case as an example. Let X be the simplicial complex made up of 5 vertices $[v_0], \ldots, [v_4]$, 6 edges $[v_0, v_1]$, $[v_0, v_2]$, $[v_1, v_2]$, $[v_1, v_4]$, $[v_2, v_3]$ and $[v_3, v_4]$, and one triangle $[v_0, v_1, v_2]$ represented in Figure 2.3.

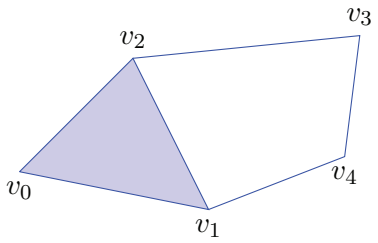


Figure 2.3: A geometric representation of X.

The boundary maps associated to the simplicial complex X are easy to obtain in matrix form:

$$\partial_1 = \begin{array}{c} \\ [v_0] \\ [v_1] \\ [v_2] \\ [v_3] \\ [v_4] \end{array} \begin{array}{c} \begin{array}{cccccc} [v_0v_1] & [v_0v_2] & [v_1v_2] & [v_1v_4] & [v_2v_3] & [v_3v_4] \end{array} \\ \begin{pmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & -1 & 0 & 0 \\ 0 & 1 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array},$$

$$\partial_2 = \begin{array}{c} \\ [v_0,v_1] \\ [v_0,v_2] \\ [v_1,v_2] \\ [v_1,v_4] \\ [v_2,v_3] \\ [v_3,v_4] \end{array} \begin{array}{c} [v_0,v_1,v_2] \\ \begin{pmatrix} 1 \\ -1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \end{array}.$$

The boundary map ∂_0 is the null function on the set of vertices. Then we can compute the Betti numbers:

$$\begin{aligned} \beta_0(X) &= \dim\ker\partial_0 - \dim\operatorname{im}\partial_1 \\ &= 5-4 \\ &= 1 \\ \beta_1(X) &= \dim\ker\partial_1 - \dim\operatorname{im}\partial_2 \\ &= 2-1 \\ &= 1 \end{aligned}$$

2.2.2 Abstract simplicial complexes

There are several famous types of abstract simplicial complexes, here we focus on two particular abstract simplicial complexes.

Definition 21 (Čech complex)**.** *Given (X,d) a metric space, ω a finite set of points in X, and ϵ a real positive number. The Čech complex of parameter ϵ of ω, denoted $\mathcal{C}_\epsilon(\omega)$, is the abstract simplicial complex whose k-simplices correspond to unordered $(k+1)$-tuples of vertices in ω for which the intersection of the $k+1$ balls of radius ϵ centered at the $k+1$ vertices is non-empty.*

Thus the Čech complex characterizes the coverage of a domain, this is the representation that we will use for wireless sensor network.

Although the Čech complex is hard to compute, there exists an approximation:

Definition 22 (Vietoris-Rips complex)**.** *Given (X,d) a metric space, ω a finite set of points in X, and ϵ a real positive number. The Vietoris-Rips complex of parameter ϵ of ω, denoted $\mathcal{R}_\epsilon(\omega)$, is the abstract simplicial complex whose k-simplices correspond to unordered $(k+1)$-tuples of vertices in ω which are pairwise within distance less than ϵ of each other.*

We can see an example of wireless sensor network representation by a Vietoris-Rips complex in Figure 2.4.

Only graph information is required to build a Vietoris-Rips complex. In the same way we can build an abstract simplicial complex from any graph: each k-simplex is included in the complex if every one of its $(k-1)$-faces already are. That abstract simplicial complex is called the clique complex of the graph.

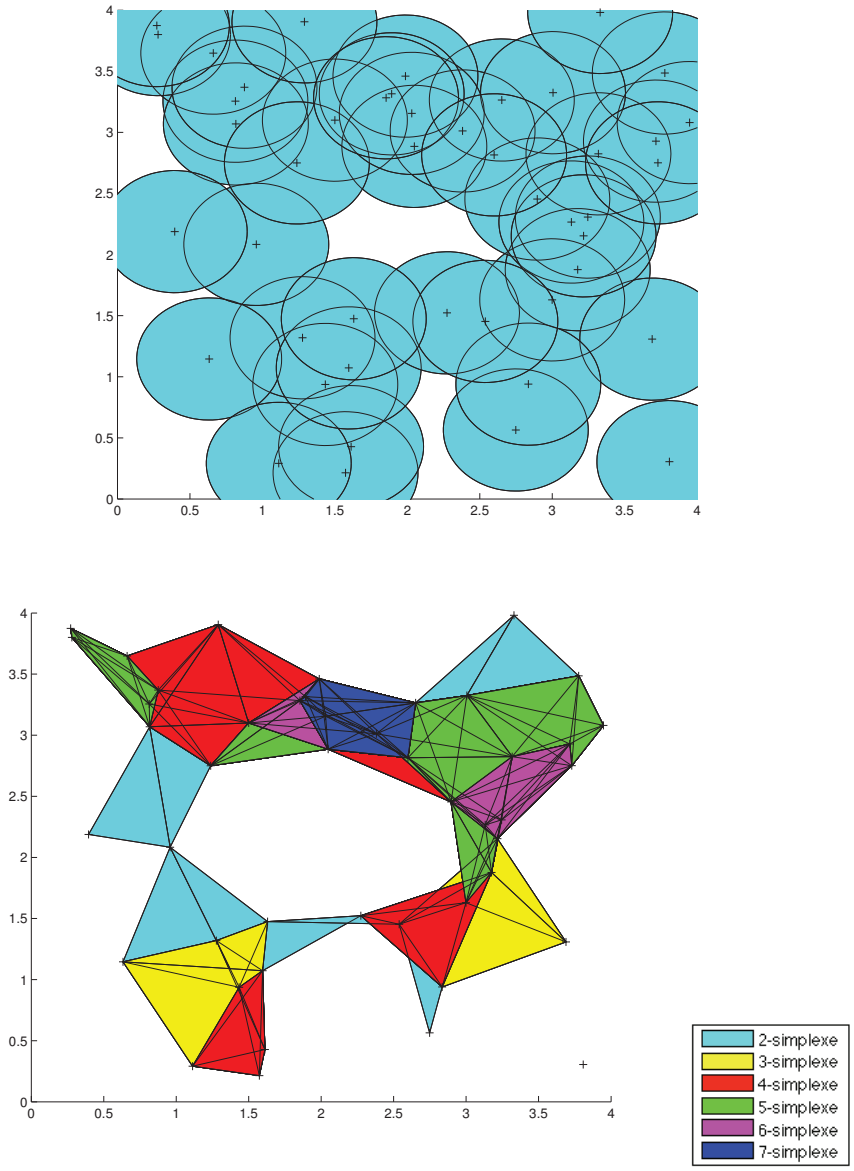


Figure 2.4: A wireless sensor network and its associated Vietoris-Rips complex.

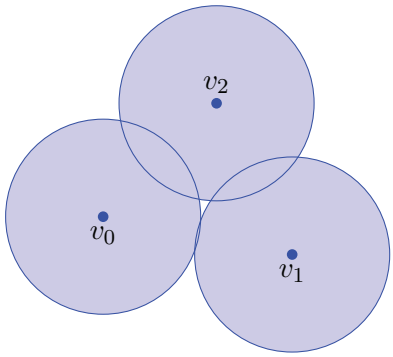
In general, unlike the Čech one, Vietoris-Rips complexes are not topologically equivalent to the coverage of an area. However, the following gives us the relation between the Čech and the Vietoris-Rips complexes:

Lemma 5. *On $\mathbb{R}^2$, given ω a finite set of points in X, and ϵ a real positive number,*

$$\mathcal{R}_{\sqrt{3}\epsilon}(\omega) \subset \mathcal{C}_\epsilon(\omega) \subset \mathcal{R}_{2\epsilon}(\omega).$$

A proof of this lemma can be found in [25].

One can easily verify that the Čech complex $\mathcal{C}_\epsilon(\omega)$ and the Vietoris-Rips complex $\mathcal{R}_{2\epsilon}(\omega)$ only differ on specific triangles. For example, we can consider this set of 3 vertices with their communication disks of radii ϵ:



Then its Čech complex representation is three 1-simplices with no 2-simplex:

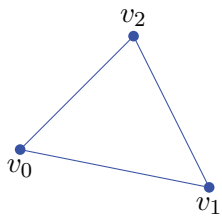


Figure 2.5: Čech complex $\mathcal{C}_\epsilon(\omega)$

However, since the Vietoris-Rips complex is entirely build on graph description, it contains a 2-simplex as soon as there are three 1-simplices linking three 0-simplices:

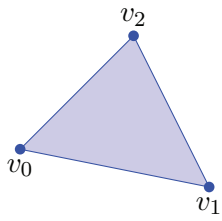


Figure 2.6: Vietoris-Rips complex $\mathcal{R}_{2\epsilon}(\omega)$

But the absence of 2-simplex in the Čech complex can be seen by building the Vietoris-Rips complex $\mathcal{R}_{\sqrt{3}\epsilon}(\omega)$:

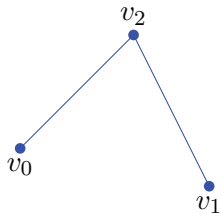


Figure 2.7: Vietoris-Rips complex $\mathcal{R}_{\sqrt{3}\epsilon}(\omega)$

For coverage simplicial complexes, as the Čech and the Vietoris-Rips complex, the Betti numbers represent the number of k-dimensional holes. Indeed the k-th Betti number β_k

counts the number of cycles of k-simplices which are not filled with $(k+1)$-simplices. For example, β_0 counts the number of 0-dimensional holes, that is the number of connected components. And β_1 counts the number of holes in the plane, then β_2 is the number of voids inside a given 3-D surface. If we are in dimension d, the k-th Betti number for $k \geq d$ has no geometric meaning.

Part I

Simplicial complexes reduction algorithm

Chapter 3

Reduction algorithm for energy savings in wireless sensor networks

In this chapter, we present the central result of this thesis: a reduction algorithm for simplicial complexes. The simplicial complex representation is used to provide a mathematical description of the topology of a wireless sensor network. Then our reduction algorithm aims at reducing the number of sensors, i.e. vertices, without modifying the topology of the network, i.e. the Betti numbers of the simplicial complex. We provide a full description of our algorithm, as well as simulations results. In the last section, some mathematical properties of the reduction algorithm are presented. This work has been published in [86].

3.1 Introduction

Sensors are autonomous systems: they are neither plugged in nor physically connected to each other. Battery life is thus a key problem and energy savings a crucial point in wireless sensor networks management. There even exists a great number of different definitions for a wireless sensor network lifetime as surveyed in [29]. Our approach to network lifetime is rather naive: we consider a static image of the network. To overcome a sensor network sensitivity to coverage holes or disconnected components, a well-known solution is to deploy an excessive number of sensors. By using more sensors than needed to cover an area or to connect a network, one ensures redundant coverage or full connectivity. However, doing so has a cost in hardware, in maintenance, as well as in battery life. Therefore, a first approach to expand the lifespan of a wireless sensor network and to reduce power consumption would logically be to put some of the sensors on standby, since there are too many of them. However, by doing it randomly, one could modify the topology of the sensor network by creating a coverage hole or breaking the connectivity.

This is why we propose here an algorithm which returns the set of sensors that can be put in standby without modifying the topology of the network. Given a simplicial complex, our algorithm removes vertices in an optimized order, keeping the topology of the complex unchanged. An example of an execution of this algorithm is shown in Figure 3.1.

We show that the algorithm reaches a Nash equilibrium: every vertex in the final simplicial complex is needed to maintain the homology. This means that the algorithm

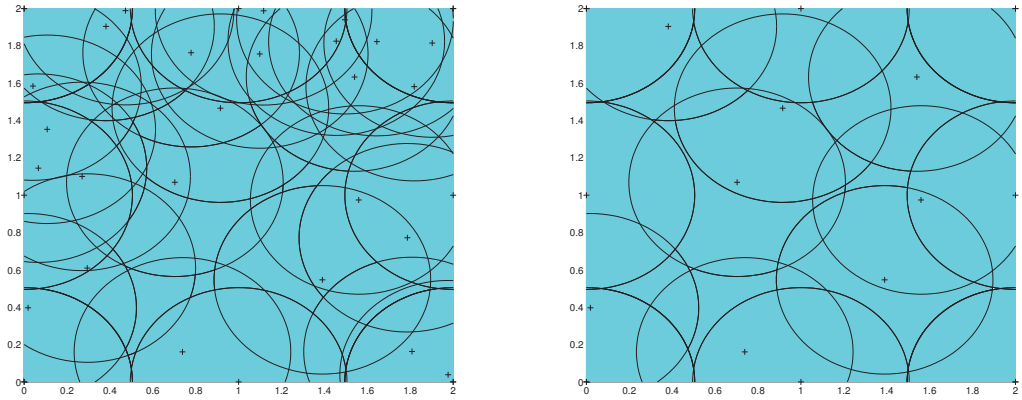


Figure 3.1: A sensor network before and after the execution of the coverage algorithm.

reaches a local optimum. We evaluate a lower and an upper bound for the number of removed vertices. The average complexity of the algorithm is analyzed for two types of random simplicial complexes: Erdös-Rényi complexes and Poisson random geometric complexes. We show that this complexity is polynomial for the former and (slightly) exponential for the latter, on which we will elaborate in the next chapter. We also provide some characteristics for the resulting complex in the wireless sensor network coverage problem.

This is the first reduction algorithm based on simplicial complex representation using homology aimed at energy savings in wireless sensor networks. A usual approach to power management in networks is the usage of the connectivity graph, such as in the dominating graph problem [43]. However, graphs are 2-dimensional objects. One vertex has full knowledge of its neighbors, but there is no representation of the interaction between these neighbors. Therefore, there is no notion of coverage in graphs. Simplicial complexes allow us to represent higher order relations, and are thus more convenient for the representation of wireless sensor networks. Several works may seem at first glance related to our approach but they do not exactly fit our purposes. In [30, 50], the authors use reduction of chain complexes to compute homology, reducing the covered domain, which make it inapplicable to a coverage problem. Witness complexes reduction, which is a reduction to a chosen number of vertices, is used in [23] to compute topological invariants. In the latter paper, as in the articles on reduction of chains complexes, the authors use reduction to compute the homology, whereas we use homology results to reduce optimally a simplicial complex. Finally, the authors of [17] present a game-theoretic approach to power management where they define a coverage function. However this method requires precise location information as well as coverage knowledge. Moreover, authors aim at identifying sub-optimal solutions that do not guarantee an unmodified coverage.

This chapter is organized as follows: Section 3.2 is devoted to the description of our reduction algorithm. Some simulation results are given in Section 3.3. Finally in Section 3.4, we discuss the mathematical properties of the algorithm.

3.2 Reduction algorithm

In this section, we present the reduction algorithm which provides which sensors may be put on standby in a wireless sensor network without altering the network's topology. In this algorithm we use simplicial homology to represent the wireless sensor network without location information, and to characterize its topology. But we also use information from the simplicial complex representation to identify the non-needed sensors, the main idea being to put on standby sensors which are represented by vertices belonging to the largest simplices.

The algorithm needs two inputs. First it needs the abstract simplicial complex, with all its simplices specified. With only an abstract simplicial complex, the optimal reduction with no modification in the homology will always be to reduce the complex to a simple vertex. That is why the algorithm needs as a second input a list of vertices of the abstract simplicial complex that have to be kept by the reduction algorithm. We call these vertices *critical* vertices. Then the algorithm removes non-critical vertices and their faces one by one from the simplicial complex without changing its homology, i.e. its Betti numbers. At the end, we obtain a *final* simplicial complex, and a list of removed vertices.

There are as many nonzero homology groups, hence nonzero Betti numbers, as sizes of simplices in the abstract simplicial complex. Therefore it is possible to define different algorithms following the number of Betti numbers that have to be kept unchanged. We will denote by k_0 the number of Betti numbers that the algorithm takes into account.

In the wireless sensor network case, the abstract simplicial complex will typically be a Čech complex or a Vietoris-Rips complex in two dimensions. The only Betti numbers of a Čech or a Vietoris-Rips complex that have geometric meaning are β_0, and β_1 in two dimensions. We thus consider two algorithms:

- The first algorithm, henceforth referred to as the connectivity algorithm, maintains connectivity within the simplicial complex, and does not take coverage into account, i.e. it only maintains the number of connected components β_0 and $k_0 = 1$.
- The second algorithm is the coverage algorithm. It takes into account both connectivity and coverage, i.e. it maintains the number of connected components β_0 and the number of coverage holes β_1, and $k_0 = 2$. It therefore constitutes the general case in two dimensions.

The list of critical vertices can be viewed as a list of active sensors that have to stay connected as they are, or extremity sensors of a line-shaped network for the connectivity algorithm. In the coverage algorithm, the critical vertices will be the sensors lying on the boundary of the area that is to stay covered, that includes both the external boundary and the holes boundary. We need all the *boundary* vertices in order to not shrink the area, nor enlarge coverage holes. While the external boundary vertices are quite easy to discover: using the convex hull, or directly defined by the network manager; the hole boundary vertices are more tricky to obtain. the authors of [89] propose an algorithm in order to find them. But in the main application of our algorithm: power consumption reduction in wireless sensor networks, we consider that there are too many sensors to cover an area that we want to reduce the number: therefore we consider that there is no coverage hole. So the discovery of the hole boundary vertices is not a problem.

Let us now define the full domain hypothesis for the wireless sensor network application case, i.e. when the reduction algorithm is applied to a Čech or a Vietoris-Rips complex in two or less dimensions:

Definition 23 (Full domain hypothesis)**.** *In dimension $d \leq 2$, we define if $k_0 = d$, for a Čech or a Vietoris-Rips complex with $\beta_0 = 1$ and, if $k_0 = 2$, $\beta_1 = 0$, the full domain hypothesis which holds when all the vertices of the abstract simplicial complex lie within the same geometric domain defined by the critical vertices.*

For the connectivity algorithm in one dimension, this means that the critical vertices are two extremity vertices and all other vertices have to be in the same path linking the two critical vertices.

For the coverage algorithm in two dimensions, this means that the critical vertices are boundary vertices and all other vertices have to lie in the area defined by the critical vertices.

We can note that it is always possible to satisfy the full domain hypothesis by removing, before the reduction algorithm, all vertices not in the path defined by the critical vertices for the connectivity algorithm in one dimension, or not lying in the convex hull of the critical vertices in two dimensions.

3.2.1 Degree calculation

The first step of the algorithm is the calculation of a number, which we call the *degree*, that we define for every k_0-simplex, where k_0 is the number of Betti numbers to be kept unchanged. To connect vertices, we only need 1-simplices, and to cover an area, we only need 2-simplices. Generalizing this idea, we have that we only need k_0-simplices to maintain k_0 Betti numbers. And larger simplices, i.e. simplices with more than $k_0 + 1$ vertices, are superflous for the k_0-th homology (first k_0 Betti numbers). We intend to characterize the superfluousness of k_0-simplices with the following definition of degree:

Definition 24. *For k_0 an integer, the degree of a k_0-simplex $[v_0, v_1, \ldots, v_{k_0}]$ is the size of its largest coface:*

$$D[v_0, v_1, \ldots, v_{k_0}] = \max\{d \mid [v_0, v_1, \ldots, v_{k_0}] \subset d\text{-}simplex\}.$$

By definition we have $D[v_0, v_1, \ldots, v_{k_0}] \geq k_0$.

For the remainder of the thesis, let $s_k(X)$, or s_k, be the number of k-simplices of the simplicial complex X. Let $D_1, \ldots, D_{s_{k_0}}$ denote the s_{k_0} degrees of a simplicial complex; they are computed according to Algorithm 4.

Algorithm 4 Degree calculation

```
for i = 1 → s_{k_0} do
    Get (v_0, ..., v_{k_0}) the vertices of the i-th k_0-simplex
    k = k_0
    while (v_0, ..., v_{k_0}) are vertices of a (k+1)-simplex do
        k = k + 1
    end while
    D_i = k
end for
return D_1, ..., D_{s_{k_0}}
```

We can see an example of values of the degree for 2-simplices in Figure 3.2: The degree of an alone 2-simplex is 2, when it is the face of 3-simplex it becomes 3.

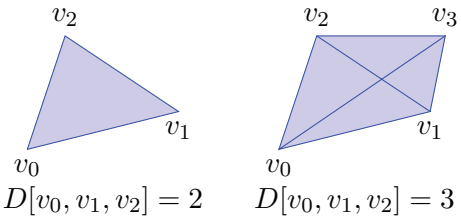


Figure 3.2: Example of values of the degree for 2-simplices

3.2.2 Index computation

The goal of the reduction algorithm is to remove vertices and not k_0-simplices, so we bring the superfluousness information of the k_0-simplices down to the vertices with an *index*. We consider a vertex sensitive if its removal leads to a change in the Betti numbers of the simplicial complex. Since a vertex is as sensitive as its most sensitive k_0-simplex, the index of a vertex is the minimum of the degrees of the k_0-simplices it is a vertex of:

Definition 25. *The index of a vertex v is the minimum of the degrees of the k_0-simplices it is a vertex of:*

$$I[v] = \min\{D[v_0, v_1, \ldots, v_{k_0}] \mid v \in [v_0, v_1, \ldots, v_{k_0}]\},$$

If a vertex v is not a vertex of any k_0-simplex then $I[v] = 0$.

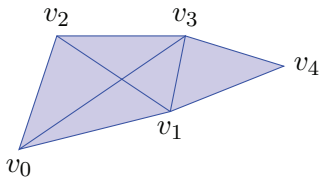
Let $v_1, v_2, \ldots, v_{s_0}$ be the vertices of the simplicial complex, the computation of the s_0 indices is done as shown in Algorithm 5.

Algorithm 5 Indices computation

```
for i = 1 → s_0 do
  I[v_i] = 0
  for j = 1 → s_{k_0} do
    if v_i is vertex of k_0-simplex j then
      if I[v_i] == 0 then
        I[v_i] = D_j
      else
        I[v_i] = min{I[v_i], D_j}
      end if
    end if
  end for
end for
return I[v_1], ..., I[v_{s_0}]
```

We can see in Figure 3.3 an example of index values in a simplicial complex. Vertices of a k_0-simplex are more sensitive than vertices of only largest simplices.

The index of a vertex is then an indicator of the density of vertices around the vertex: an index of k_0 indicates that at least one of its k_0-coface has no $(k_0 + 1)$-cofaces, whereas



$$D[v_0, v_1, v_2] = D[v_0, v_1, v_3] = D[v_0, v_2, v_3] = D[v_1, v_2, v_3] = 3$$
$$D[v_1, v_3, v_4] = 2$$
$$I[v_0] = I[v_2] = 3 \text{ and } I[v_1] = I[v_3] = I[v_4] = 2$$

Figure 3.3: Example of values of the indices of vertices in a simplicial complex

a higher index indicates that each one of its k_0-cofaces is the face of at least one larger simplex. The main idea of the algorithm is thus to remove the vertices with the greatest indices.

Remark 6. *An index of* 0 *indicates that the vertex has no k_0-coface: the vertex is isolated up to the k_0-th degree. For $k_0 = 1$, that means that the vertex is disconnected from any other vertices. For $k_0 = 2$, the vertex is only linked to other vertices by edges, therefore it is inside a coverage hole. Under the full domain hypothesis, these vertices should not exist.*

3.2.3 Optimized order for the removal of vertices

The algorithm now removes vertices from the initial abstract simplicial complex following an optimized order. We begin by computing the first k_0 Betti numbers via linear algebra. Then the degrees of all the k_0-simplices and the indices of all the vertices are computed as explained in the previous section. The critical vertices of the input list are given a negative index, to flag them as unremovable. Then the indices give us an order for the removal of vertices: the greater the index of a vertex, the more likely it is superfluous for the k_0-th homology of its simplicial complex. Therefore, the vertices with the greatest index are candidates for removal: one is chosen randomly. The removal of a vertex leads to the movaln of all its k-faces for every integer $k \geq 1$.

At every vertex removal, we need to ensure that the homology is unchanged. We compute the first k_0 Betti numbers thanks to the boundary maps every time a vertex is removed. This computation is instantaneous since the complex is already built, and only adjacency matrices defining the complex are needed. If the removal changes the homology, the vertex is put back in the simplicial complex. Moreover its index is assigned a temporary negative value so that the vertex is not candidate for the following draw. The temporary values are recalculated at the next effective removal of a vertex.

Otherwise, if the removal does not change the homology, the removal is confirmed. And the modified degrees of the k_0-simplices and the indices of the vertices are recalculated. We can note that only the vertices of maximum index can have their indices changed, as explained in Lemma 6. Moreover, in order to improve the algorithm performance it is possible to only compute the modified degrees of k_0-simplices. It suffices to flag the k-simplices, $k > k_0$, which are the largest cofaces of k_0-simplices. Then when one of them is deleted, the degrees of its k_0-faces have to be recalculated.

Lemma 6. *When a vertex of index $I_{\max}$ is removed, only the vertices sharing an $I_{\max}$-simplex with it, and of index $I_{\max}$ can have their index changed.*

Proof Let w be the removed vertex of index $I_{\max}$, and v any vertex of the simplicial complex.

If v does not share any simplex with w, none of the degrees of its k_0-simplex will change, and neither will its index.

Thus let us consider that the largest common simplex of v and w is a k-simplex, $k > 0$. If $k < k_0$, then the removal of w and this k-simplex has no incidence at all on the index of v by definition. Next, if $k_0 \leq k < I_{\max}$ then w has an index $k < I_{\max}$, which is absurd. We can thus assume that $k \geq I_{\max}$. Either the index of v is strictly less than $I_{\max}$ and thus comes from a simplex not shared with w, therefore it is unmodified by the removal of w. Or if the index of v is $I_{\max}$, it either still comes from a $I_{\max}$-simplex not shared with w and remains unmodified, or it comes from a common $I_{\max}$-simplex. This latter case is the only case where the index of v can change.

The algorithm goes on removing vertices until every remaining vertex is unremovable, thus achieving optimal result. Every vertex is unremovable when all indices are strictly below k_0. By the definition of indices, that means when all indices are equal to zero or negative (temporarily or not).

Some things can be refined with the full domain hypothesis in the wireless sensor network application case. First the stopping condition can be improved to $I_{\max}$ being below or equal to k_0:

Lemma 7. *Under the full domain hypothesis, the algorithm can stop when all vertex indices are below or equal to k_0.*

Proof Let us suppose the input data satisfies the full domain hypothesis. Let v be a vertex of index $I(v) = k_0$, which means that at least one of its k_0-cofaces has no $(k_0 + 1)$-coface. The removal of this vertex would lead to the removal of this particular k_0-simplex. Since we need to maintain the homology on the entire geometric domain that is defined by the critical vertices without shrinking it, this would lead to a k_0-dimensional hole, and an increment of β_{k_0-1}.

For the connectivity algorithm, the removal of an edge which is not the side of a triangle leads to a disconnectivity in the path linking the critical vertices. For the coverage algorithm, the removal of a triangle which is not the face of a tetrahedron leads to the creation of a coverage hole inside the area defined by the critical vertices.

Then under the full domain hypothesis in the wireless sensor network application, all temporarily unremovable vertices (negative index vertices) stay unremovable:

Lemma 8. *Under the full domain hypothesis, when the removal of a vertex modify the homology, it will always modify it.*

Proof In the full domain hypothesis, the distance between the critical vertices can not be decreased, or the covered domain can not have its area decreased. Since the domain size is unchanged, as in the proof of Lemma 7, the removal of a vertex that has led to a change in a Betti number will always lead to the same change.

Remark 7. *In the wireless sensor network application case, for the coverage algorithm, and under the full domain hypothesis, it is possible to omit the optimized order of vertices, removing every vertex of index strictly greater than k_0 when the homology is unchanged. Then the degree calculation and indices computation can be limited to the choice greater*

than k_0 or not. Doing that, one loses the optimized order for the vertex removal. In this case, the algorithm can be distributed, every node can compute a decentralized algorithm with only the following information: its neighbors and their relations. This has been done in [91]. this decentralized algorithm is only possible for the coverage algorithm under the full domain hypothesis, otherwise the reached solution is non optimal.

We give in Algorithm 6 the whole algorithm for the conservation of the first k_0 Betti numbers.

Algorithm 6 Reduction algorithm

Require: Simplicial complex X, list L_C of critical vertices.
  Computation of $\beta_0(X), \dots, \beta_{k_0-1}(X)$
  Computation of $D_1(X), \dots, D_{s_{k_0}}(X)$
  Computation of $I[v_1(X)], \dots, I[v_{s_0}(X)]$
  for all $v \in L_C$ **do**
    $I[v] = -1$
  end for
  $I_{\max} = \max\{I[v_1(X)], \dots, I[v_{s_0}(X)]\}$
  while $I_{\max} \geq k_0$ **do**
    Draw w a vertex of index $I_{\max}$
    $X' = X \backslash \{w\}$
    Computation of $\beta_0(X'), \dots, \beta_{k_0-1}(X')$
    if $\beta_i(X') \neq \beta_i(X)$ for some $i = 0, \dots, k_0 - 1$ **then**
      $I[w] = -1$
    else
      Computation of $D_1(X'), \dots, D_{s'_{k_0}}(X')$
      for $i = 1 \to s'_0$ **do**
        if $I[v_i(X')] == I_{\max}$ **then**
          Recomputation of $I[v_i(X')]$
        end if
        if $I[v_i(X')] == -1$ && $v_i \notin L_C$ **then**
          Recomputation of $I[v_i(X')]$
        end if
      end for
      $X = X'$
    end if
    $I_{\max} = \max\{I[v_1(X)], \dots, I[v_{s_0}(X)]\}$
  end while
  return X

3.3 Simulation results

Our simulations aim to illustrate our algorithm. The results are highly dependent on the chosen parameters: for the connectivity algorithm the percentage of removed vertices is linked to the fact that the critical vertices are connected without any intermediary in the simplicial complex or not and how; for the coverage algorithm, it is linked to the ratio of the initial number of vertices to the number of vertices needed for the coverage of the area defined by the critical vertices. We simulated our algorithm on two types of complexes.

First we consider the Erdös-Rényi clique complex of the eponymous random graph:

Definition 26 (Erdös-Rényi complex)**.** *Given n an integer and p a real number in $[0, 1]$, the Erdös-Rényi complex of parameters n and p, denoted $G(n, p)$, is an abstract simplicial complex with n vertices. Then each edge is included with probability p independent from every other edge. Then a k-simplex, for $k \geq 2$, is included if and only if all of its faces already are.*

We can see in Figure 3.4 one realization of the reduction algorithm keeping the number of connected components β_0 unchanged on a Erdös-Rényi complex of parameter $n = 15$ and $p = 0.3$. The critical vertices are chosen at random: a vertex is critical with probability $p_c = 0.5$ independently from every other vertices. Critical vertices are circled, and non-critical vertices which are kept to maintain the connectivity between critical vertices are starred.

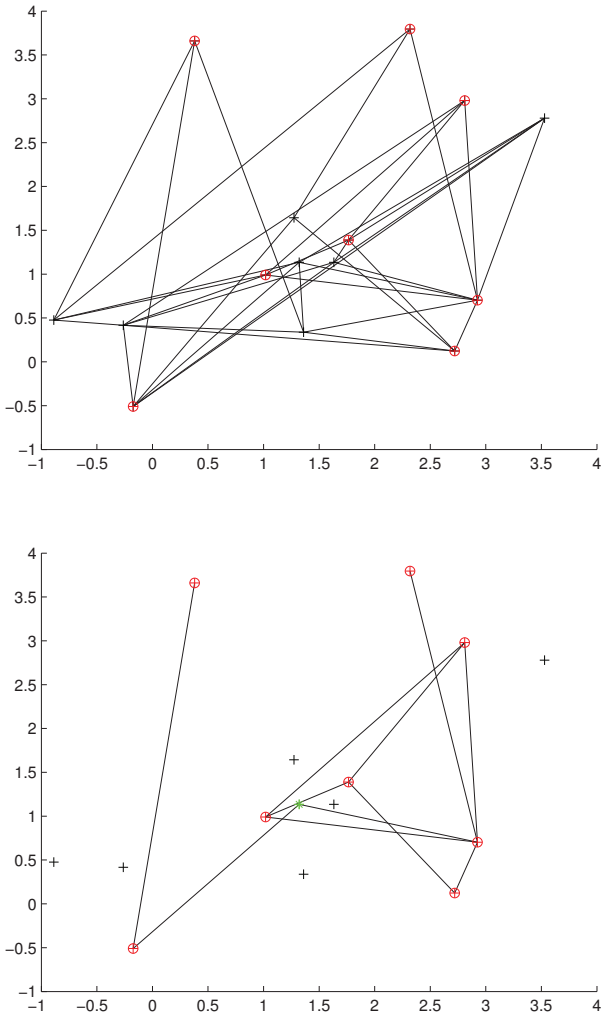


Figure 3.4: A Erdös-Rényi complex before and after the reduction algorithm for $k_0 = 1$.

With the parameters $n = 60$ vertices, and $p = 0.2$, on average for 1000 configurations with one connected component, with p_c taking values between 0.1 and 0.5 (200 configurations each), the algorithm removed 98% of the non-critical vertices:

p_c	Percentage of removed vertices
0.1	94.96%
0.2	97.14%
0.3	98.59%
0.4	99.43%
0.5	99.87%

Then we are interested in illustrating the wireless sensor network application with a Vietoris-Rips complex in two dimensions. We simulate the set of vertices with a Poisson point process:

Definition 27. *A Poisson point process ω with intensity λ on a Borel set X is defined by:*

i) For any $A \in \mathcal{B}(X)$, the number of vertices in A, $\omega(A)$, is a random variable following a Poisson law of parameter $\lambda S(A)$, $\Pr(\omega(A) = k) = e^{\lambda S(A)} \frac{(\lambda S(A))^k}{k!}$.

ii) For any disjoint A, $A' \in \mathcal{B}(X)$, the random variables $\omega(A)$ and $\omega(A')$ are independent.

We can see in Figure 3.5 one realization of the connectivity algorithm on a Vietoris-Rips complex of parameter $\epsilon = 1$ based on a Poisson point process of intensity $\lambda = 4$ on a square of side length 4, with random critical vertices. A vertex is critical with probability $p_c = 0.5$ independently from every other vertices.

For this parameters configuration with one connected component, on average for 1000 runs, the connectivity algorithm removed 96.01% of the non-critical vertices.

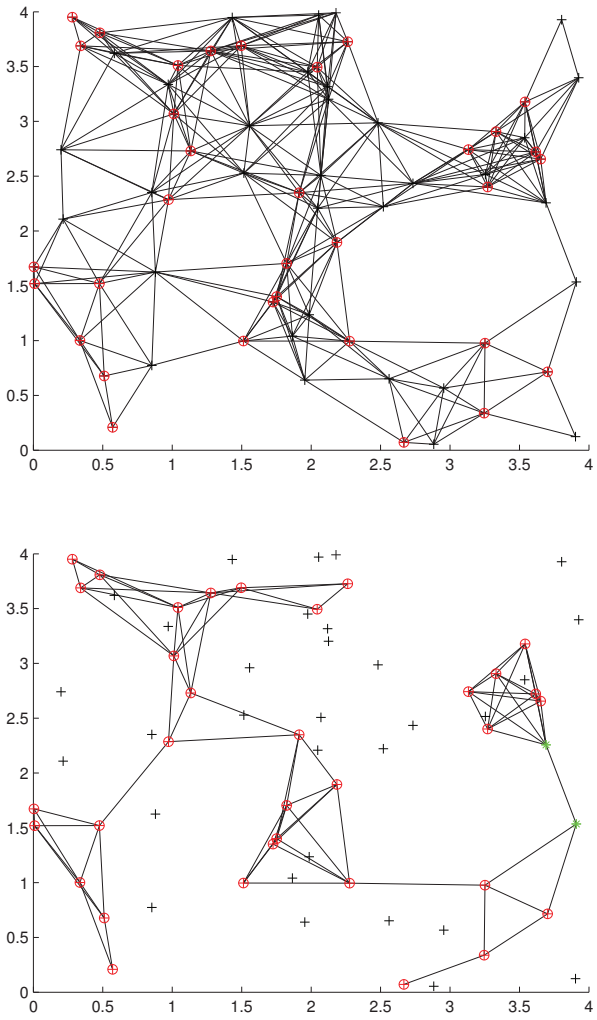


Figure 3.5: A Vietoris-Rips complex before and after the connectivity algorithm.

We can see in Figure 3.6 one realization of the coverage algorithm on a Vietoris-Rips complex of parameter $\epsilon = 1$ based on a Poisson point process of intensity $\lambda = 4.2$ on a square of side length $a = 2$, with a fixed boundary of vertices on the square perimeter. The critical vertices are the boundary vertices, thus satisfying the full domain hypothesis for the coverage algorithm. They are circled on the figure.

In order to ensure perfect coverage, we chose the following parameters: $\epsilon = 1$, $\lambda = 5.1$ and $a = 2$, on average for 1000 configurations, the coverage algorithm removed 69.35% of the non-critical vertices.

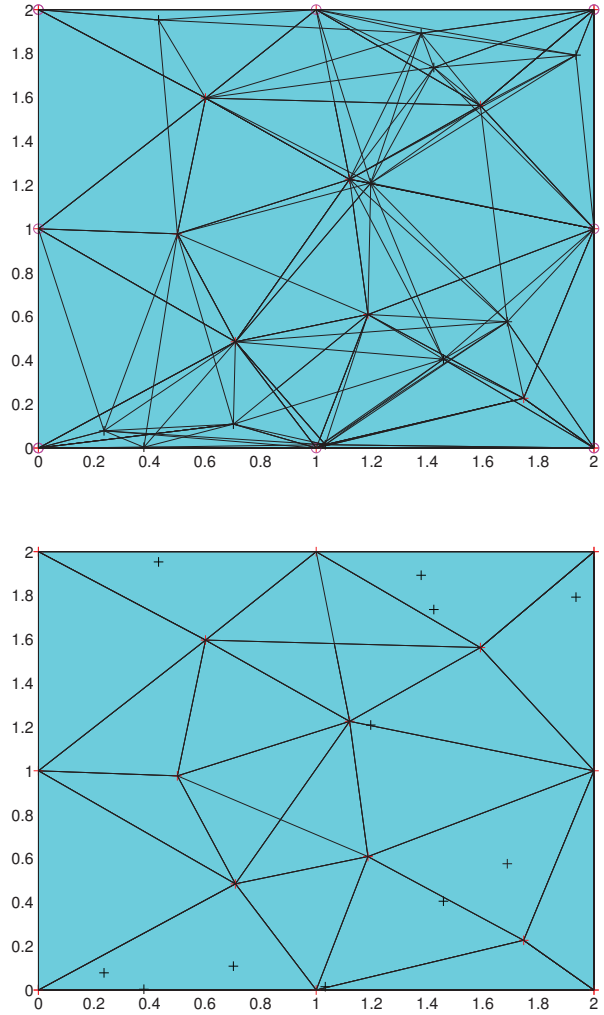


Figure 3.6: A Vietoris-Rips complex before and after the coverage algorithm.

3.4 Mathematical properties

The first property of our algorithm is that the reached solution is optimal. It may not be the optimum solution if there are multiple optima but it is a local optimum. In game theory vocabulary that means that the algorithm reaches a Nash equilibrium as defined in [17].

Theorem 8 (Maximal solution)**.** *The reduction algorithm reaches a maximal number for the number of removed vertices: every vertex in the final simplicial complex is needed to maintain its homology. No further vertex can be removed without violating the constraints on the Betti numbers*

Proof In the final simplicial complex, every vertex is of index strictly smaller than k_0 in the general case. By the definition of the indices, we then differentiate two types of vertices: vertices of index -1 and 0.

First, negative indices are given to vertices to flag them as unremovable. Either a vertex is of negative index because it is a critical vertex, in which case it is required to stay in the complex. Or a vertex is of negative index if its removal leads to a change in the Betti numbers. Since there has been no other removal that has changed the complex, that fact is still true.

Then a vertex of null index is an isolated vertex. If it is isolated up to the k_0-th degree, its removal will decrease the k_0-th Betti number. For example, the removal of a disconnected vertex would decrement β_0. As well the removal of a vertex inside of a hole would lead to the union of 2 or more holes.

Finally, if the input data satusfies the full domain hypothesis, the proof of Lemma 7 shows that the algorithm still reaches a Nash equilibrium.

Secondly, we are able to find both a lower and an upper bound the number of removed vertices. The number of removed vertices is at least one vertex removed by index value, and at most every vertex of non minimal index before the algorithm:

Theorem 9 (Upper and lower bounds)**.** *Let E_k be the set of vertices that have indices k before the algorithm. The number of removed vertices M is bounded by:*

$$\sum_{k=k_0+1}^{I_{\max}} \mathbb{1}_{[E_k \neq \emptyset]} \leq M \leq \sum_{k=k_0+1}^{I_{\max}} |E_k|.$$

with $|E_k|$ being the cardinality of E_k.

Proof Let us begin with the upper bound. The maximum number of vertices the can be removed by the algorithm is the number of vertices that initially have their index strictly greater than k_0. This is an optimal upper bound since this number of removed vertices is reached in the following case:

Let a k-simplex, with $k > k_0$, be the initial simplicial complex, and n_C of its vertices be the initial critical vertices, necessarily $n_C \leq k+1$. The n_C critical vertices have negative indices, the $k+1-n_C$ other vertices have an index of k, and they are all removed.

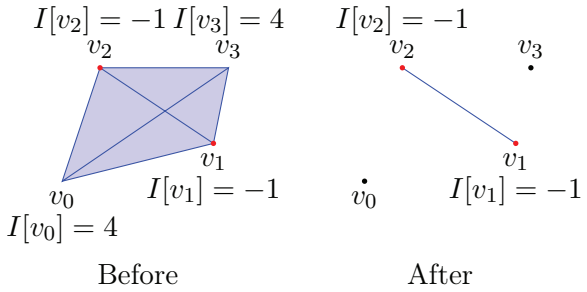


Figure 3.7: Example of this case with $k = 3$ and $n_C = 2$, the two vertices v_0 and v_3 are removed by the algorithm.

For the lower bound, we have seen in Lemma 6 that the removal of a vertex of index $I_{\max}$ can only change the index of vertices of index $I_{\max}$. In the worst case, it decreases all

indices $I_{\max}$ and the value of $I_{\max}$ changes, not necessarily to $I_{\max} - 1$ depending on the critical vertices. Thus we can see, that at least one vertex per index value can be removed, hence the result.

The lower bound is reached in the previous case if $n_C = k$.

Remark 8. *As seen in the proof, both upper and lower bounds are reached and equal in the case of the complete graph where all but one vertices are removed.*

Then, we are able to find some characteristics about the final complex in the wireless sensor network application case.

If we use the following definition of a dominating set:

Definition 28 (Dominating set)**.** *As defined in [43], a set $S \subseteq V$ of vertices of a graph $G = (V, E)$ is called a dominating set if every vertex $v \in V$ is either an element of S or is adjacent to an element of S.*

Under the full domain hypothesis, the final set of vertices is a dominating set of the initial set:

Theorem 10. *In two dimensions for the coverage algorithm applied to a Čech or a Vietoris-Rips complex under the full domain hypothesis, the set of remaining vertices of the final complex is a dominating set of the set of vertices of the initial complex.*

Proof Under the full domain hypothesis, initial vertices are all in the geometric domain defined by the critical vertices. For the coverage algorithm that means that initial vertices lie in the area defined by the critical vertices. The homology of the complex is unmodified by the algorithm therefore there is no coverage hole in the final complex. The area is still covered. Then, each point of this area is inside a 2-simplex. This is true for every vertex of the initial complex, it is thus adjacent to three remaining vertices.

Remark 9. *In the wireless sensor network application, that means that every standby sensor is one-hop away from an awake sensor.*

Finally we investigate the complexity of our algorithm. If we consider the number of vertices $n = s_0$ to be the parameter, then the complexity of the implementation of the simplicial complex is potentially exponential. Indeed, the number of simplices in a simplicial complex of n vertices is at most $2^n - 1$. Therefore the data implementation is at most of complexity $O(2^n)$ compared to the number of vertices n.

Theorem 11 (Complexity)**.** *The complexity of the algorithm to keep k_0 Betti numbers for a simplicial complex of s_k k-simplices and $n = s_0$ vertices, is upper bounded by:*

$$n^2 s_{k_0} + (n + s_{k_0}) \sum_{k=0}^{C-1} s_k$$

with C being the clique number of the underlying graph.

Proof Let $C - 1$ be the size of the largest simplex in the simplicial complex, C being known as the clique number of the underlying graph. For the computation of the degrees of every k_0-simplex, the algorithm traverses at most all the k-simplices for $k_0 < k \leq C - 1$

to check if the k_0-simplex is included in it. Since there is s_k k-simplices, that means that the degrees computation complexity is upper bounded by $s_{k_0} \sum_{k=k_0+1}^{C-1} s_k$.

For the computation of the indices, the algorithm traverses, for every one of the n vertices, its k_0-cofaces, which is at most all the k_0-simplices. The complexity of the computation of the indices has therefore an upper bound of ns_{k_0}.

These computations are done at the beginning of the algorithm. Then at each removal of vertices, at most n removed vertices:

- The simplices of the removed vertex are deleted: complexity which upper bound is $\sum_{k=0}^{C-1} s_k$.
- The Betti numbers are recomputed via the adjacency matrices which already exist.
- The modified degrees are recomputed automatically with the deletions of simplices.
- The at most $n-1$ modified indices are recomputed: complexity which upper bound is $(n-1)s_{k_0}$.

Corollary 12. *When n goes to infinity the complexity of the algorithm is $O(n^{k_0+1}2^n)$.*

Proof This is a direct consequence of Theorem 11 since the number of k-simplices s_k can be upper bounded by $\binom{n}{k+1}$, k_0 is typically small relative to n (equal to 1 or 2 in two dimensions), and C has n for upper bound.

Remark 10. *Since the data size is of complexity $O(2^n)$, the complexity of the algorithm is polynomial relative to the complexity of the data.*

We are able to refine the results regarding the complexity of our algorithm for some specific simplicial complexes.

Corollary 13. *If we apply our algorithm to the Čech complex $\mathcal{C}_\epsilon(\omega)$, defined with the maximum norm, based on a Poisson point process of parameter λ on a torus of side a in dimension 2. The complexity of the algorithm has for upper bound: $O((1+(\frac{2\epsilon}{a})^2)^n)$ on average when n goes to infinity.*

Proof According to [26], we have that if $s_0 = n$:

$$\begin{aligned}
\mathbf{E}\left[s_{k-1}\right] &= \binom{n}{k} k^2 \left(\frac{2\epsilon}{a}\right)^{2(k-1)} \text{ and,} \\
\operatorname{Cov}\left[s_{k-1}, s_{l-1}\right] &= \sum_{i=1}^{k\wedge l} \binom{n}{u_i}\binom{u_i}{k}\binom{k}{i}\left(u_i + 2\frac{(k-i)(l-i)}{i+1}\right)^2 \left(\frac{2\epsilon}{a}\right)^{2(u_i-1)},
\end{aligned}$$

with $u_i = k + l - i$.

Plugging this into the complexity formula of Theorem 11 gives the result.

Corollary 14. *The complexity of the algorithm for the Erdös-Rényi complex based on the graph $G(n,p)$ is of the order of $O(n^{2(k_0+2)})$ on average when n goes to infinity.*

Proof In the Erdös-Rényi complex, the expected values of the number of k-simplices and of the product of numbers of k-simplices are given in [14]:

$$\begin{aligned}\mathbf{E}\left[s_{k-1}\right] &= \binom{n}{k} p^{\binom{k}{2}},\\ \mathbf{E}\left[s_{k-1}s_{l-1}\right] &= \sum_{i=0}^{k\wedge l} \binom{n}{k}\binom{k}{i}\binom{n-k}{l-i} p^{\binom{k}{2}+\binom{l}{2}-\binom{i}{2}}.\end{aligned}$$

The results then comes from the complexity formula of Theorem 11.

We can see that there are great differences in the complexity of our algorithm depending on which random simplicial complex it is applied to. This can be explained by the fact that the two simplicial complexes investigated here have very different behaviors when n goes to infinity. Indeed, the Erdös-Rényi complex is made of disconnected components when n tends to infinity with a fixed probability p. However the Čech complex can percolate depending on parameter ϵ. We can see in the complexity formula of Theorem 11 that the size of the largest simplex, i.e. the clique number, is of great importance in the complexity: it can make the latter go from polynomial to exponential. That is why we investigate its behavior in the next chapter.

Chapter 4

Clique number of random geometric graphs

While investigating the complexity of our reduction algorithm, we are reduced to compute the behavior of the size of the largest simplex in the random simplicial complex. In graph theory, this characteristic is known as the clique number. The clique number C of a graph is namely the largest clique size in the graph. In Chapter 4, we find the almost sure asymptotic behavior of the clique number for the underlying graph of the Vietoris-Rips complex on a binomial point process as the number of vertices goes to infinity. This graph is the random geometric graph of n vertices, taken uniformly at random, including an edge beween two vertices if their distance, taken with the uniform norm, is less than a parameter r. The n vertices are uniformly sampled on the torus of side a in dimension d. The behavior of the clique number depends on the percolation regime of the graph. Setting $\theta = (\frac{r}{a})^d$, the percolation regimes of the random geometric graph are defined along variation of $\frac{1}{n}$ before θ. In the subcritical regime where $\theta = o(\frac{1}{n})$, we exhibit the intervals of θ where C takes the same value asymptotically almost surely. In the critical regime, $\theta \sim \frac{1}{n}$, we show that C is growing slightly slower than $\ln n$ asymptotically almost surely. Finally, in the supercritical regime, $\frac{1}{n} = o(\theta)$, we prove that C grows as $n\theta$ asymptotically almost surely. We also investigate the behavior of related graph characteristics: the chromatic number, the maximum vertex degree, and the independence number. This work is the subject of [28].

4.1 Introduction

In this chapter we switch from simplicial homology vocabulary to graph theory vocabulary for a smoother reading.

In graph theory, a clique in a graph is a subset of its vertices where all vertices are connected to each other by an edge, this is the corresponding term to simplex from simplicial homology. Thus a vertex is a itself a clique of size 1, an edge is a 2-clique and a triangle is a 3-clique. Cliques are one of the basic concepts of graph theory and are the subjects of many research articles since the middle of the twentieth century [58]. A maximum clique is a clique of the largest possible size in a given graph. The clique number of a graph is then the number of vertices in a maximum clique of this graph. The clique number is a well known graph characteristic and occurs in various problems. For instance, the NP-complete clique problem in computer science, which consists in finding either a particular clique such as a maximum clique or whether there exists a clique larger than a given size in a graph,

has been highly documented, see [84] for example.

However, the clique number also appears in problems not directly related to graph theory. As we have seen in the previous chapter, if one considers the simplicial complex representation for sensor networks, the wireless sensor network is then represented by a coverage simplicial complex, the Čech complex, in order to have a description of its coverage and coverage holes. If one wants to implement the full description of the complex, one needs to implement all of its simplices. The complexity of the data implementation obviously depends on the size of the largest simplex. As one can easily notice, with a change of domain vocabulary the size of the largest simplex in a simplicial complex is the clique number of the underlying graph. So the clique number appears in the complexity of a simplicial complex implementation. We also encountered it while computing the complexity of our reduction algorithm presented in the previous chapter. As can be seen in Theorem 11, the clique number is highly critical there since it can make the complexity of our algorithm go from polynomial to exponential.

One of the most famous random graph is the Erdös-Rényi model $G(n,p)$ with n vertices where each edge occurs with probability p. In this model, the probability distribution of the clique number is known when the number of vertices n goes to infinity and p is fixed [14]. This result was derived from the investigation of Matula in [61], who proved the 2-point concentration of the clique number distribution. However such results are not yet available for the random geometric graph $G(n,r)$ of n vertices sampled uniformly and including an edge between two vertices if their distance is less than r. Indeed, it is necessary to consider the variations of r depending on n, thus consider different regimes, incrementing the complexity of the problem. Regimes were also investigated for the Erdös-Rényi model, considering the variations of p depending on n. In [55], the authors proved the phase transition at $p = \frac{1}{n}$ for the connectivity. For the random geometric graph case, Penrose in [73] distinguished two types of regime transition. One can consider regimes transitions through percolation, or through connectivity. In [51], Kahle studied the homology of the clique complex of the random geometric graph in the three regimes delimited by percolation (subcritical, critical and supercritical), and in the connected regime.

In this chapter, we find the asymptotic behavior of the clique number of a random geometric graph $G(n,r)$ with the uniform norm when n goes to infinity. Derived from the stochastic analysis made in [78], the authors of [26] give the explicit moments of the number of cliques of size k for the random geometric graph on the torus. Thanks to these results, we are able to find the asymptotic behavior of the clique number for the three regimes defined by percolation. In the subcritical regime, we find the intervals where the clique number takes asymptotically almost surely a given finite value. In the critical regime, we show that the clique number grows slightly slower than $\ln n$ asymptotically almost surely. Finally in the supercritical regime, we prove the growth of the clique number asymptotically almost surely. We also investigate the behaviors of the related quantities: the chromatic number, the maximum vertex degree, and the independence number.

To our knowledge, this is the first work including the asymptotical behavior of the clique number of a random geometric graph for all three regimes. In [39], the authors proved that monotone properties of random geometric graphs, such as the connectivity of the graph, have sharp thresholds. In [74], then in [68] for a weaker assumption, the authors proved a conjecture of Penrose [73] stating that, in the subcritical regime, the clique number becomes concentrated on two consecutive integers, as in the Erdös-Rényi model [61]. Moreover, in the subcritical regime, weak laws of large numbers and central limit theorems [75–77] have been found by Penrose and Yukich for some functionals in random geometric graphs,

including the clique number. Finally in the supercritical regime, using the uniform norm, Appel and Russo [8], were able to find strong laws for the maximum vertex degree and for cliques. In particular, they found the asymptotical behavior of the clique number in the supercritical regime via the behavior of the maximum vertex degree. We propose here the opposite approach and find the same results.

The remainder of this chapter is organized as follows. First in the following section, we present our model with the definitions necessary to its construction, and the previous results that have allowed us to write this article. Then Section 4.3 is devoted to the results we found for the asymptotical behavior of the clique number of a random geometric graph, separated in three subsections depending on the regime where the graph is lying. Finally, Section 4.4 is devoted to the investigation on the quantities related to the clique number in graph theory.

4.2 Model

In order to describe our model of a random geometric graph we need some preliminary definitions from graph theory and probability. Let us first denote by $\mathbb{T}_a^d$ the torus of side a in dimension d.

Definition 29. *Let f be the uniform probability density function on the torus $\mathbb{T}_a^d$, let $x_1, x_2, \dots$ be a set of independent and identically distributed d-dimensional random variables with common density f, and let $X_n = \{x_1, \dots, x_n\}$. For the set of n points X_n and the positive distance r, let us define the random geometric graph $G(n, r)$ as the graph with n vertices $V(G) = X_n$ and edges $E(G) = \{[x, y] \mid d(x, y) \leq r\}$.*

Definition 30. *In a graph, a clique is a subset of vertices such that every two vertices in the clique are connected by an edge.*

Definition 31. *The clique number of a graph, that we call C, is the number of vertices in a clique of the largest possible size in the graph.*

Definition 32. *We say that $G(n, r)$ asymptotically almost surely has property P if $\mathbf{P}[G(n, r) \in P] \to 1$ when n tends to infinity.*

For the remainder of this chapter, we consider a random geometric graph $G(n, r)$ of n vertices sampled following a uniform distribution on the torus $\mathbb{T}_a^d$ of side a in dimension d, and including an edge between two vertices if their distance is less than r. Taking the torus $\mathbb{T}_a^d$ instead of the cube $[0, a]^d$ allows us to not consider boundary effects. Indeed, if one samples the vertices on the cube, vertices in the corner do not have the same behavior as vertices in the center of the cube. Sampling on the torus eliminates the different vertices behaviors. We denote C its clique number.

In [26], the authors provide expressions for moments of random variables of the Čech complex by means of Malliavin calculus. Thanks to their use of the uniform norm the so-called Čech complex is the exact same as the Vietoris-rips complex, which is the clique complex of the random geometric graph. Indeed in simplicial homology, instead of considering only vertices and edges as in graph theory, cliques of n connected vertices are also

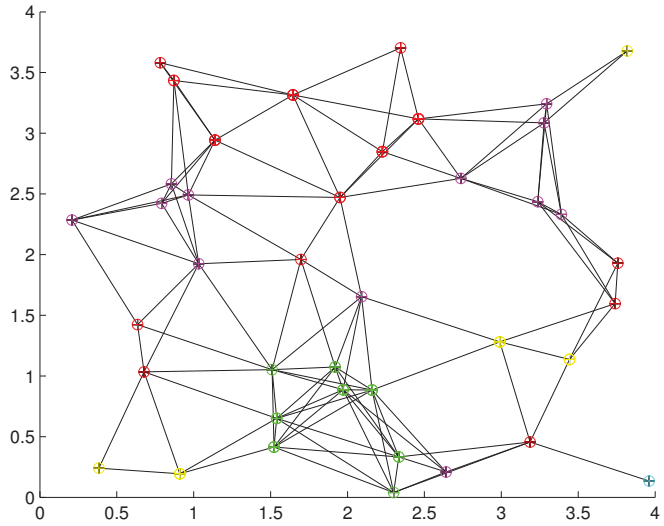


Figure 4.1: Example of a random geometric graph whose vertices are colored following the size of their largest clique.

considered and called simplices. And the Vietoris-Rips simplicial complex is the complex whose simplices are the cliques of the random geometric graph. Then we are able to apply the results of [26] to random geometric graphs, especially in the expressions of the expectation and the variance of the number of cliques of size k.

In order to use the results from [26], we first need to make a few assumptions:

1. First, we use the uniform norm to calculate the distance between two vertices. Note that [26] also provides the needed results for the Euclidean norm, however their expressions are not as tractable as the ones for the uniform norm.

2. Let us take $\theta = \left(\frac{r}{a}\right)^d$, then we must have $\theta \leq \left(\frac{1}{2}\right)^d$. This insures that the graph is small on the torus, and that no vertex is its own neighbor.

We can note that the parameter θ acts as a coverage parameter: if one puts vertices along a regular grid of side r, one needs $\frac{1}{\theta}$ vertices to cover the area of $\mathbb{T}_a^d$. These assumptions hold for the remainder of the chapter.

Thus, let N_k denote the number of cliques of k vertices. Then $N_1 = n$ is the number of vertices, and the results of [26] state that:

Theorem 15 ([26]). *The expectation and variance of the number of k-vertex cliques, for $k > 1$, in a random geometric graph $G(n, r)$ on a torus $\mathbb{T}_a^d$ are given by:*

$$\mathbf{E}\left[N_k\right] = \binom{n}{k} k^d \theta^{k-1}, \tag{4.1}$$

and,

$$\mathbf{V}\left[N_k\right] = \sum_{i=1}^{k} \binom{n}{2k-i} \binom{2k-i}{k} \binom{k}{i} \theta^{2k-i-1} \left(2k - i + 2\frac{(k-i)^2}{i+1}\right)^d. \tag{4.2}$$

This result and other results of [26] are given in Appendix A.

In this chapter, we are interested in the asymptotic behavior of the clique number as n tends to infinity. Throughout the article, we use Bachman-Landau notations. For non-negative functions f and g we write as n tends to infinity:

- $f(n) = o(g(n))$ if for every $\epsilon > 0$ there exists n_0 such that for $n \geq n_0$, we have $f(n) \leq \epsilon g(n)$. We say that f is dominated by g asymptotically.
- $f(n) = O(g(n))$ if there exists $k > 0$ and n_0 such that for $n \geq n_0$, we have $f(n) \leq kg(n)$. We say that f is bounded by g asymptotically.
- $f(n) \sim g(n)$ if $f(n) = O(g(n))$ and $g(n) = O(f(n))$. We say that f and g are equal asymptotically.
- $f(n) \ll g(n)$ if $\frac{f(n)}{g(n)} = o(1)$. We say that f is small compared to g asymptotically.

4.3 Asymptotical behavior

We studied the asymptotical behavior of the clique number under percolation regimes: these regimes are defined following the variations of θ compared to $\frac{1}{n}$.

4.3.1 Subcritical regime

In this subsection, we consider that $\theta = o(\frac{1}{n})$. In the subcritical regime for the percolation, the random geometric graph mainly consists of disconnected components as one can see in Figure 4.2. The clique number is small compared to the number of vertices n when the latter tends to infinity. Therefore we can focus on k-vertex cliques for k asymptotically small compared to n.

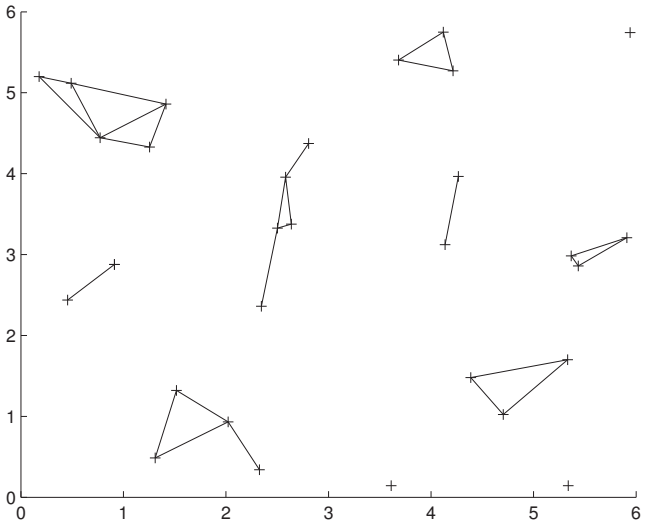


Figure 4.2: Subcritical regime

From the exact expressions we had from [26], we can derive asymptotic expressions for the moments of the number of k-cliques:

Lemma 9. *For $k \geq 1$ small compared to n, according to Theorem 15,*

$$\mathbf{E}[N_k] \sim n^k \theta^{k-1} k^d, \text{ and } \mathbf{V}[N_k] \sim n^k \theta^{k-1} k^d.$$

Proof This is a direct consequence of the subcritical regime hypothesis applied to Equations 4.1 and 4.2.

In this regime, the random geometric graph $G(n, r)$ is similar to the Erdös-Rényi graph $G(n, p)$ which we define as follows:

Definition 33. *Let n be an integer, and $0 < p < 1$. Then the Erdös-Rényi random graph $G(n, p)$ is the graph of n vertices where each edge occurs with probability p.*

When n tends to infinity with p unchanged, the graph $G(n, p)$ is composed of many disconnected components like the random geometric graph in the subcritical regime. Therefore our investigation of the asymptotic behavior of the clique number in this regime is similar to the one exposed in [14] by Bollobás and Erdös. The authors first define n_k and n'_k such that the expectation of the number of cliques of size k is respectively upper bounded by $k^{-(1+\epsilon)}$ and lower bounded by $k^{1+\epsilon}$ for $\epsilon > 0$. The values of n_k and n'_k are approximated by $p^{-k/2}$ and $(1 + \frac{3 \log k}{k}) p^{-k/2}$ respectively. This leads to the result:

Theorem 16 ([14]). *For almost every graph $G(n, p)$ there is a constant c such that if $n'_k \leq n \leq n_{k+1}$ for some $k > c$ then the clique number C is $C = k$.*

In the random geometric graph $G(n, r)$ case, we observe the variations of $\theta = (\frac{r}{a})^d$ instead of n, when n tends to infinity. Indeed θ varies with n, and it makes more sense to find the intervals of θ where the clique number takes a finite value. The first step is to find the regimes of θ where we can bind the expectation of the number of cliques of size k, integer, in a similar way:

Definition 34. *We define for $\eta > 0$ and for $k \geq 1$:*

$$\theta'_k = \frac{k^{\frac{1+\eta-d}{k-1}}}{n^{\frac{k}{k-1}}}, \text{ and } \theta_k = \frac{k^{-\frac{1+\eta+d}{k-1}}}{n^{\frac{k}{k-1}}}.$$

Then for $\theta > \theta'_k$, thanks to the approximations of Lemma 9 we have:

$$\mathbf{E}[N_k] \geq n^k \left(\frac{k^{\frac{1+\eta-d}{k-1}}}{n^{\frac{k}{k-1}}} \right)^{k-1} k^d \geq k^{1+\eta}.$$

And for $\theta < \theta_k$,

$$\mathbf{E}[N_k] \leq n^k \left(\frac{k^{-\frac{1+\eta+d}{k-1}}}{n^{\frac{k}{k-1}}} \right)^{k-1} k^d \leq k^{-(1+\eta)}$$

This leads to our main theorem for the subcritical regime:

Theorem 17. *In the subcritical regime, for $k \geq 1$ small compared to n, and $\theta'_k < \theta < \theta_{k+1}$, the clique number is asymptotically almost surely $C = k$.*

Proof For $k \geq 1$ small compared to n when n tends to infinity, we can easily check that if $n > k^{2(1+\eta)k}$. This holds in particular when n tends to infinity and k is fixed. Then we have that $\theta_{k+1} > \theta'_k$.

We can now consider θ such that $\theta'_k < \theta < \theta_{k+1}$. Thanks to the approximations of Lemma 9, we can, on one hand, find an upper bound for the probability of the non-existence of cliques of size k:

$$\mathbf{P}[N_k = 0, \theta > \theta'_k] \leq \frac{\mathbf{V}[N_k]}{\mathbf{E}[N_k]^2} \sim \frac{1}{\mathbf{E}[N_k]} \leq \frac{1}{k^{1+\eta}}.$$

On the other hand, we can find an upper bound for the probability of existence of cliques of size $k+1$:

$$\mathbf{P}[N_{k+1} > 0, \theta < \theta_{k+1}] \leq \mathbf{E}[N_{k+1}] \leq \frac{1}{(k+1)^{1+\eta}}.$$

Finally we have:

$$\mathbf{P}[\exists \theta, \theta'_k < \theta < \theta_{k+1}, C \neq k] < \frac{1}{k^{1+\eta}} + \frac{1}{(k+1)^{1+\eta}}.$$

As the sum $\sum_{k=1}^{\infty} k^{1+\eta}$ converges, the Borel-Cantelli theorem implies that with the exception of finitely many k's, for all θ such that $\theta'_k < \theta < \theta_{k+1}$, one has $C = k$. Then when n goes to infinity, we have asymptotically almost surely that $C = k$:

$$\mathbf{P}[C = k, \theta'_k < \theta < \theta_{k+1}] \to 1,$$

concluding the proof.

4.3.2 Critical regime

In the critical regime, where $\theta \sim \frac{1}{n}$, percolation occurs: disconnected components of the random geometric graph begin to connect into one sole connected component as seen in Figure 4.3. The clique number is still rather small compared to n, allowing us to consider only the k-vertex cliques for $k = O(n)$ when n goes to infinity.

In this regime we have a direct approximation of our variable θ, allowing us to compute an approximation of the expected number of cliques of size k integer from the Equation 4.1.

Lemma 10. *For $k = O(n)$, and according to Theorem 15, we have:*

$$\mathbf{E}[N_k] \sim \frac{1}{\sqrt{2\pi}} n k^{d-k-\frac{1}{2}}, \text{ and } \mathbf{V}[N_k] \sim \frac{1}{\sqrt{2\pi}} n k^{d-k-\frac{1}{2}}.$$

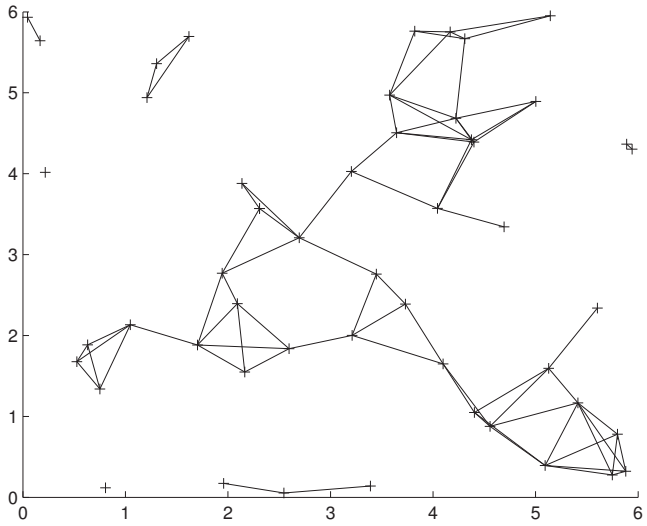


Figure 4.3: Critical regime

Proof We have from Equation 4.1 that $\mathbf{E}[N_k] = \binom{n}{k}\theta^{k-1}k^d$.

After some calculations via the Stirling's approximation $n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$, we obtain the result for the expectation.

Thanks to the fact that $k = O(n)$, we can still approximate the variance of N_k by its dominating term in $i = k$, and $\mathbf{V}[N_k] \sim \mathbf{E}[N_k]$.

From that approximation and using the same process as in the previous subsection, we can write the main theorem of this subsection:

Theorem 18. *In the critical regime, the clique number grows asymptotically almost surely slower than* $\ln n$ *with an arbitrarily small distance:*

$$(\ln n)^{1-\eta} < C < \ln n, \quad \forall \eta > 0.$$

Proof First, for $k > \ln n$, we can find an upper bound for the expectation approximation of Lemma 10 by:

$$\mathbf{E}[N_k] < n(\ln n)^{d-\frac{1}{2}-\ln n}.$$

One can easily check that $n(\ln n)^{d-\frac{1}{2}-\ln n} \to 0$. Since $\mathbf{P}[N_k > 0] \leq \mathbf{E}[N_k]$, the probability that there exists k-vertex cliques tends to 0 and:

$$\mathbf{P}[C > k] = \mathbf{P}[N_k > 0] \to 0 \quad \forall k > \ln n,$$

and $C < \ln n$ asymptotically almost surely.

Then, for $k < (\ln n)^{1-\eta}$ with $\eta > 0$, we can now find a lower bound for the expectation approximation by:

$$\mathbf{E}[N_k] > n(\ln n)^{(1-\eta)(d-\frac{1}{2}-(\ln n)^{1-\eta})}$$

And one can check that $n(\ln n)^{(1-\eta)(d-\frac{1}{2}-(\ln n)^{1-\eta})} \to +\infty$. Then, thanks to the asymptotic equivalence of the variance and the expectation of N_k, we have $\mathbf{P}[N_k = 0] \leq \frac{1}{\mathbf{E}[N_k]}$: the probability that there exists no k-vertex cliques tends to 0, and:

$$P[C < k] = \mathbf{P}[N_k = 0] \to 0 \quad \forall k < (\ln n)^{1-\eta}.$$

Thus, $C > (\ln n)^{1-\eta}$ asymptotically almost surely.

4.3.3 Supercritical regime

In the supercritical regime, $\frac{1}{n} = o(\theta)$, the random geometric graph $G(n, r)$ is connected and tends to become the complete graph as we can see in the example of Figure 4.4. The asymptotic behavior of the clique number has already been studied in this regime in [8] by Appel and Russo. They first find the almost sure asymptotic rate for the maximum vertex degree. Then by squeezing the clique number between two values of the maximum vertex degree, they obtain its asymptotic behavior. We propose here an alternative approach, where the clique number asymptotic rate is used to squeeze the other related quantities such as the maximum vertex degree.

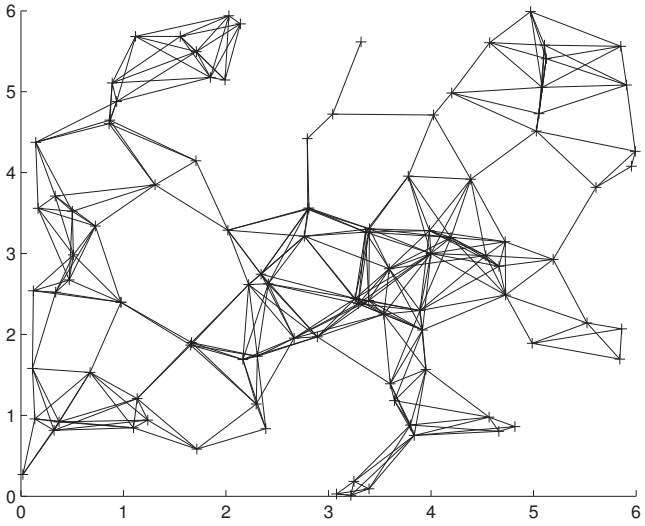


Figure 4.4: Supercritical regime

In this regime, percolation has occurred, that is to say that the graph is connected and the number of k-cliques is not asymptotically small anymore compared to n. Therefore an upper bound via the expected number of k-vertex cliques is not a good enough approach anymore. Instead, we came back to the definition of the random geometric graph.

For the first step of our exploration, we use a similar argument as in [8]. To cover the torus $\mathbb{T}_a^d$, one needs at least $\lceil \frac{1}{\theta} \rceil$ balls of diameter r in dimension d. If one places these balls along a lattice square grid with spacing r, one can denote B_i, for $1 \leq i \leq \lceil \frac{1}{\theta} \rceil$, the

$\lceil \frac{1}{\theta} \rceil$ needed balls centered on the points of the grid and of radius $\frac{r}{2}$. Then the number of vertices n is smaller than the sum of the number of vertices in each ball B_i:

$$n < \sum_{i=1}^{\lceil \frac{1}{\theta} \rceil} \#\{B_i\}$$

where $\#\{B_i\}$ is the number of vertices in B_i.

Moreover, vertices in the same ball B_i are within distance r of each other, therefore they form a clique. By definition, for every i we have $\#\{B_i\} \leq C$. Thus we have:

$$C > \frac{n}{\lceil \frac{1}{\theta} \rceil} \geq \frac{n\theta}{1+\theta}$$

When θ tends to 0, the clique number is asymptotically greater than $n\theta$. We can now write the main result of this subsection:

Theorem 19. *In the supercritical regime, the clique number C grows as $n\theta$ asymptotically almost surely.*

Proof We still have to prove that the clique number is asymptotically almost surely smaller than $n\theta$.

In the random geometric graph $G(n,r)$, a clique of size k occurs when k vertices are in the same ball of diameter r. Without loss of generality we can center the ball on one of the vertex of the graph. We have:

$$\begin{aligned} \mathbf{P}[C > n\theta] &= \mathbf{P}[N_{n\theta} > 0] \\ &= \mathbf{P}[\exists\, x, \#\{B(x, \frac{r}{2})\} \geq n\theta - 1], \end{aligned}$$

where x is a vertex of the graph $G(n,r)$, and $\#\{B(x, \frac{r}{2})\}$ the number of vertices of $G(n,r)$ in the ball centered in x and of radius $\frac{r}{2}$.

Let $x_1, \ldots, x_n$ denote the n vertices of the graph $G(n,r)$, their positions are independent, thus we can write:

$$\begin{aligned} \mathbf{P}[\exists\, i \in \{1,\ldots,n\}, \#\{B(x_i, \frac{r}{2})\} \geq n\theta - 1] &\leq \mathbf{P}[\bigcup_{i=1}^{n} \#\{B(x_i, \frac{r}{2})\} \geq n\theta - 1] \\ &\leq \sum_{i=1}^{n} \mathbf{P}[\#\{B(x_i, \frac{r}{2})\} \geq n\theta - 1] \\ &\leq n\mathbf{P}[\#\{B(x_1, \frac{r}{2})\} \geq n\theta - 1]. \end{aligned}$$

The number of vertices in the ball $B(x_1, \frac{r}{2})$ follows a binomial distribution Binom$(n-1, \theta)$. Therefore Hoeffding's inequality implies that:

$$\begin{aligned} \mathbf{P}[\#\{B(x_1, \frac{r}{2})\} > n\theta] &\leq \mathbf{P}[\#\{B(x_i, \frac{r}{2})\} > (n-1)(\theta + \frac{\theta}{n-1})] \\ &\leq \exp(-2\frac{\theta^2}{n-1}). \end{aligned}$$

Then the clique number C is asymptotically almost surely smaller than $n\theta$, concluding the proof.

4.4 Other graph characteristics

In this section, for the graph G, we will denote its clique number by $C(G)$. The results of this section for the supercritical regime were already written in [8]. We choose to present them nonetheless since we used a different approach: they first computed the behavior of the maximum vertex degree and derive results for the clique number while we did the opposite.

4.4.1 Maximum vertex degree

Let us remind the definitions of first the degree of a vertex, then the maximum vertex degree of a graph:

Definition 35. *The degree of a vertex of a graph is the number of edges incident to the vertex.*

Definition 36. *The maximum degree of a graph G, denoted $\Delta(G)$, is the maximum degree of its vertices.*

By its definition we have the following inequality between the maximum vertex degree and the clique number of any graph G:

$$C(G) - 1 \leq \Delta(G). \tag{4.3}$$

Then if we consider the random geometric graphs $G(n, r)$ and $G(n, 2r)$, by doubling the adjacency distance between two vertices, we ensure that:

$$\Delta(G(n, r)) \leq C(G(n, 2r)) - 1. \tag{4.4}$$

Finally we have, for any graph G of n vertices $v_1, \ldots, v_n$ and N_2 edges, the equality $2N_2 = \sum_{i=1}^{n} \deg v_i$. For the graph $G(n, r)$, taking the mean, we have then thanks to Equation 4.1:

$$2^d(n-1)\theta \leq \mathbf{E}\left[\Delta(G(n, r))\right]. \tag{4.5}$$

Given these relations, we can write our main result regarding the maximum vertex degree:

Theorem 20. *In the subcritical regime, for $k \geq 1$ small compared to n and θ such that $\theta'_k < \theta < \theta_{k+1}$, the maximum vertex degree $\Delta(G(n, r))$ is asymptotically almost surely greater than $k - 1$.*

In the critical regime, the maximum vertex degree $\Delta(G(n, r))$ grows asymptotically almost surely faster than $(\ln n)^{1-\eta}$ pour tout $\eta > 0$, and slower than $2^d n\theta$.

In the supercritical regime, the maximum vertex degree $\Delta(G(n, r))$ grows as $2^d n\theta$ asymptotically in mean. It is asymptotically almost sure that $\Delta(G(n, r))$ grows slower than $2^d n\theta$.

Proof For the first part of the theorem concerning the subcritical regime, this is a direct consequence of Theorem 17 and Inequality 4.3.

In the critical regime, we use Inequalities 4.3 and 4.4, and the result from Theorem 18 for $C(G(n, r))$. Then one has to observe that the graph $G(n, 2r)$ is in the supercritical regime, so $C(G(n, 2r))$ asymptotically almost surely grows as $2^d n\theta$ according to Theorem 19.

In the supercritical regime, using Theorem 19 in Inequalities 4.4 and 4.5 concludes the proof.

4.4.2 Chromatic number

Let us first define the well-known chromatic number of a graph:

Definition 37. *The chromatic number of a graph G, denoted $\chi(G)$, is the smallest number of colors needed to color the vertices of G such that no two adjacent vertices share the same color.*

Since two vertices in the same clique can not have the same color, we have for any graph G:

$$C(G) \leq \chi(G).$$

If ones consider a greedy coloring: for each vertex the first available color in the graph is assigned. Then the number of colors used in this greedy algorithm is $\Delta(G) + 1$. Thus for any graph G, $\chi(G) \leq \Delta(G) + 1$. Then we can write, using Inequality 4.4:

$$C(G(n,r)) \leq \chi(G(n,r)) \leq C(G(n,2r)). \tag{4.6}$$

And our main result for the chromatic number asymptotical behavior is:

Theorem 21. *In the subcritical regime, for $k \geq 1$ small compared to n and θ such that $\theta'_k < \theta < \theta_{k+1}$,the chromatic number $\chi(G(n,r)$ is asymptotically almost surely greater than k.*

In the critical regime, the chromatic number $\chi(G(n,r)$ grows asymptotically almost surely faster than $(\ln n)^{1-\eta}$ pour tout $\eta > 0$, and slower than $2^d n\theta$.

In the supercritical regime, the chromatic number $\chi(G(n,r)$ grows asymptotically almost surely faster than $n\theta$, and slower than $2^d n\theta$.

Proof This is a direct consequence of Inequality 4.6 and our three main Theorems 17, 18, and 19.

4.4.3 Independence number

Let us remind the notion of independent set in a graph, and derive the concept of independent number of a graph:

Definition 38. *An independent set of a graph is a set of its vertices of which no pair is adjacent.*

Definition 39. *The independence number of a graph G, denoted $\alpha(G)$ is the size of the largest independent set of G.*

The independence number and the chromatic number of a graph G are related by the following inequality proved in [70]:

$$\alpha(G)\chi(G) \geq n, \tag{4.7}$$

where n is the number of vertices of G.

Then, in the graph $G(n,r)$, to have an independent set of size k, k balls centered on the independent vertices and of radius r, must be disjoint on the torus $\mathbb{T}_a^d$:

$$kr^d \leq a^d.$$

This is true for the largest independent set:

$$\alpha(G(n,r)) \leq \frac{1}{\theta}. \tag{4.8}$$

Thanks to these relations, we obtain our main result for the asymptotical behavior of the independence number of a random geometric graph:

Theorem 22. *In the critical and the supercritical regimes, the independence number* $\alpha(G(n,r))$ *decreases asymptotically almost surely slower than* $\frac{1}{2^d\theta}$*, and faster than* $\frac{1}{\theta}$*.*

Proof This is a direct consequence of Inequalities 4.7 and 4.8 and Theorem 21.

Part II

Applications to future cellular networks

Chapter 5

Self-configuration frequency auto-planning algorithm

From now on, we intend to expand the applications of the simplicial complex representation to other types of wireless network, especially future cellular networks. In this chapter, we present a frequency auto-planning algorithm based on the simplicial complex representation and using our reduction algorithm presented in Chapter 3. Our algorithm can be applied to random wireless networks not following a regular pattern. It aims at minimizing the number of used frequencies while maximizing the coverage for each frequency. So that a minimum number of resources is used while they are operated in a optimal way. This frequency auto-planning algorithm can in particular be used for the self-configuration of future cellular networks. We also present its performance and compare it to the greedy coloring algorithm.

5.1 Introduction

Long Term Evolution (LTE) is the 3GPP standard specified in Releases 8 and 9. Its main goal is to increase both capacity and speed in cellular networks. Indeed, cellular network usage has changed over the years and bandwidth hungry applications, as video calls, are now common. Achieving this goal costs a lot of money to the operator. A solution to limit operation expenditures is the introduction of Self-Organizing Networks (SON). 3GPP standards have identified self-organization as a necessity for future cellular networks [1]. A full description of SON in LTE can be found in [44]. SON features include self-configuration, self-optimization, and self-healing functions. In particular, self-configuration functions aims at the plug-and-play paradigm: new transmitting nodes should then be automatically configured and integrated to the existing network. Upon arrival of a new node, the neighboring nodes update their dynamic neighbor tables thanks to the Automatic Neighbor Relation (ANR) feature. This information is equivalent to the connectivity information in wireless sensor networks, needed to build the simplicial complex representation.

Among self-configuration functions, we can find the dynamic frequency auto-planning problem. It is a known problem from spectrum-sensing cognitive radio where equipments are designed to use the best wireless channels in order to limit interference [42]. The different nodes of the secondary cognitive network have to choose the best frequency to use in order to maximize the coverage and minimize the interference with the base stations of

the primary network. The hierarchy in networks makes this solution not directly practicable to future cellular networks. While in earlier releases, static frequency planning was preferred, it has became a critical point since the network has a dynamic behavior with arrivals and departures of base stations, and does not always follow a regular pattern with the introduction of Femtocells and Heterogeneous Networks (HetNet).

In this chapter, we propose a frequency auto-planning algorithm which, for any given wireless network, provides a frequency planning minimizing the number of frequencies needed for a given accepted threshold of interference. The algorithm uses the simplicial complex representation of the network, and calls several instances of the reduction algorithm introduced in Chapter 3 for the allocation of each frequency. Using simplicial complex representation combined to our reduction algorithm allow us to obtain a homogeneous coverage between frequencies.

In cellular network the frequency planning problem was first introduced for GSM networks. However the constraints were not the same: the frequency planning was static with periodic manual optimizations, and in simulations base stations were regularly deployed along an hexagonal pattern. With the commercialization of Femtocells and the deployments of outdoor relays and Picocells, cells do not follow a regular pattern anymore and can appear and disappear at any time. Therefore the frequency planning problem has to be rethought in an automatic way. A naive idea for frequency auto-planning would simply be applying the greedy coloring algorithm to the sparse interference graph [47]. However, even if it reaches an optimal solution for the number of needed frequencies, their utilization can be disparate: one frequency can be planned for a large number of nodes compared to another planned for only few of them. Then if the level of interferences increase (more users, or more powered antennas), this could lead to communication problems for the overused frequency, and a whole new planning is needed. While a more homogeneous resource utilization can be more robust if interferences increase, since there are less nodes using the same frequency on average. We provide here a frequency auto-planning algorithm which aims at a more homogeneous utilization of the resources.

The remainder of this chapter is organized as follows. After a section on related work on self-configuration in future cellular networks, we introduce our frequency auto-planning algorithm in Section 5.3, principle and full description. Then we provide simulation results with performance discussion in the Section 5.4.

5.2 Self-configuration in future cellular networks

A complete survey on SON for future cellular networks is given in [4].

Configuration of the different nodes (eNBs, relays, Femtocells) of a cellular network has to be done during the deployment of the network, but also upon the arrival and departure of every node. The classic manual configuration done for previous generations of cellular networks can not be operated in future cellular networks: changes in the network, such as arrivals and departures, occur too often. Moreover, the commercialization of private Femtocells leads to the presence in the network of nodes with no access for manual support. So the future cellular networks are heterogeneous networks with no regular pattern for its nodes. They need to be able to self-configure themselves. The initial parameter that a node needs to configure are its IP adress, its neighbor list and its radio access parameters. IP adresses are out of the scope of this work, but we will discuss the two other parameters.

The neighbor list is the connectivity information we use for the construction of the simplicial complex which represents the cellular network. The selection of the nodes to put

on one's neighbor list can be based on the geographical coordinates of the nodes and take into account the antenna pattern and transmission power [57]. However, this approach does not consider changing radio environment, and requires exact location information which can be easy to obtain for eNBs, but not for Femtocells. The authors of [53] propose a better criterion for the configuration of the neighbor list: each node scans in real time the Signal to Interference plus Noise Ratio (SINR) from other nodes. This latter approach is closer to the connectivity approach used for wireless sensor networks.

Among radio access parameters, we can find frequency but also propagation parameters since the apparition of beamforming techniques via MIMO. Let us focus on the former which is the subject of this chapter. The planning of frequency channels for new nodes that do not interfere with existing nodes while still provide enough bandwidth is still an open problem. It has been addressed in the cognitive radio field, but these algorithms usually enable opportunistic spectrum access [69]. However, it is not possible to extend this type of algorithm to the frequency allocation of new nodes in cellular networks. Indeed, the new nodes would be part of the primary network, with a quality of service to achieve, so their frequency allocation needs to be guaranteed and not opportunistic. The algorithm we propose aims at allocating frequency channels to future cellular networks with non-regular deployment, so upon the arrival of a new node, the whole network is re-configured.

5.3 Frequency auto-planning algorithm

5.3.1 Main idea

We consider a cellular network that we represent by a geometric simplicial complex: the transmitting nodes (eNBs, Femtocells, relays...) are represented by vertices, then we build the Čech complex corresponding to the different coverage radii. The Čech complex characterizes the topology of the cellular network. In this application we are especially interested in the characterization of groups of nodes close to each other by simplices. Indeed, in the frequency planning problem, the goal is to assign to each node a frequency so that the interference between them is the smallest possible using a minimum number of frequencies. Here we will only consider co-canal interference: interference between two nodes using the same frequency. We introduce what we call a rejection radius, this radius defines around every node a rejection disk. If one node is within the rejection disk of another node, then we consider that they shall not share the same frequency or the level of interference will be too high for reliable communication within each one of the two cells. This rejection radius defines the interference threshold that is acceptable in the cellular network. It is the interference criterion we will use for our auto-planning algorithm.

The algorithm begins by computing the degrees of triangles and the indices of vertices, of the simplicial complex representing the cellular network, defined in Chapter 3. Then we apply a modified version of our coverage reduction algorithm: the order in which the vertices are removed is still decided by the indices but the stopping condition is not the same anymore. Indeed we are not interested into achieving optimal coverage anymore. Therefore, instead of stopping when the maximum index among every remaining vertices is below a given number, the algorithm stops when there is no more vertices in any rejection disk of any other vertex. The vertices of the resulting simplicial complex are assigned the first frequency. Then all the removed vertices are collected, and the corresponding simplicial complex recovered. This simplicial complex is a subset of the initial simplicial complex so there is no need to build another simplicial complex from scratch. The next step is then

to reapply the modified coverage reduction algorithm to this recovered simplicial complex to obtain a second set of vertices to which we assign the second frequency. The algorithm goes on until every vertex has an assigned frequency. At the end, we have a frequency assigned to every node. We ensured that no two nodes sharing the same frequency will be too close to each other: interference will be under a given threshold. Moreover, the use of our coverage reduction algorithm with the optimized order for vertices removal allows us to obtain a homogeneous coverage for every frequency.

5.3.2 Algorithm description

Algorithm 7 Frequency auto-planning algorithm

Require: Set ω of N vertices, for each vertex v its coverage radius r_v, and its rejection radius R_v.
Computation of the Čech complex $X = \mathcal{C}_r(\omega)$
Computation of $D_1(X), \ldots, D_{s_2}(X)$
Computation of $I[v_1(X)], \ldots, I[v_{s_0}(X)]$
$I_{\max} = \max\{I[v_1(X)], \ldots, I[v_{s_0}(X)]\}$
$N_{\text{planned}} = 0$
Interference $= 1$
$X' = X$
$i = 0$
while $N_{\text{planned}} < N$ **do**
  while Interference $== 1$ **do**
    Draw w a vertex of index $I_{\max}$
    $X' = X' \backslash \{w\}$
    Computation of $D_1(X'), \ldots, D_{s'_2}(X')$
    for $i = 1 \to s'_0$ **do**
      if $I[v_i(X')] == I_{\max}$ **then**
        Recomputation of $I[v_i(X')]$
      end if
    end for
    $I_{\max} = \max\{I[v_1(X')], \ldots, I[v_{s'_0}(X')]\}$
    Interference $= 0$
    for all $u, v \in X'$ **do**
      if $\|v - u\| < \max(R_v, R_u)$ **then**
        Interference $= 1$
      end if
    end for
  end while
  for all $v \in X'$ **do**
    Frequency$(v) = i$
    $N_{\text{planned}} = N_{\text{planned}} + 1$
  end for
  $X' = X \backslash X'$
  $i = i + 1$
end while
return List of assigned frequencies Frequency$(v), \forall v \in \omega$.

We give in Algorithm 7 the full frequency auto-planning algorithm: it requires the set of vertices ω, and their coverage and rejection radii, then returns the list of assigned frequencies for every vertex of ω.

It is interesting to note that we do not need to compute the Betti numbers of the Čech complex, since we are not effectively removing vertices so the topology of the network does not change. However we use the simplicial complex representation information to locate the vertices too close to each other.

We introduce three parameters in the algorithm description. First of them is the number of planned vertices: it represents the number of vertices to which we assigned a frequency already. Then, the 'Interference' parameter is a binary number to represent if there are at least two vertices with potentially the same frequency within the rejection disk of each other. Finally we also introduce the 'Frequency' notation, for every vertex, this is where we store the frequency assigned to it.

5.4 Simulation results

5.4.1 Performance

For simulation reasons, we only consider the Vietoris-Rips complex in the 2-dimensional case.

To measure the performance of our algorithm we compare the number of frequencies our algorithm plans versus the number of frequencies the greedy coloring algorithm plans. Indeed, the frequency planning can be viewed as a graph coloring problem. We consider the geometric graph whose edges are added if at least one of its extremity vertex is within the rejection disk of the other one. Then the optimal number of frequencies to assign is the chromatic number of the graph. The greedy coloring algorithm provides a coloring for a given graph assigning the first new color available for each vertex. Therefore, the greedy coloring algorithm provides a frequency planning with a number of frequency equal to the maximum vertex degree plus one. The greedy coloring gives especially good results for the number of used colors for sparse graphs as the interference graph is. However, the greedy coloring algorithm leads to a disparate utilization of frequencies. Indeed, if there is only one clique of maximum size, one frequency will be only used for one vertex of this clique, and for no other vertex in the whole configuration. Therefore, this algorithm could give good results in a homogeneous network, but not for a cluster network for example. Our algorithm aims at a more homogeneous utilization of each ressource.

We simulate the set of vertices with a Poisson point process of intensity $\lambda = 12$ on a square of side $a = 2$. The coverage radii are sampled uniformly between $a/10$ and a^2/λ, each rejection radius was equal to half its corresponding coverage radius. The results are obtained in mean over 1000 configurations. For each realization of the Poisson process, we compute the number of frequencies planned by the greedy coloring algorithm that we denote N_g. Then on all realizations with a given N_g, we compute the mean number of frequencies, denoted N_f, planned by our algorithm.

In Table 5.1, we can see the mean number of planned frequencies given the number of frequencies planned by the greedy coloring algorithm. We also indicate which percentage of the 1000 simulations these situations represent. We can see that there is a difference between the two solutions: it is not negligible in the beginning, but it decreases with the number of frequencies. Thus, our algorithm reaches its optimal performance when the number of frequencies grows for the same mean number of nodes, that is to say when there

are clusters of nodes or when the rejection radii are quite large compared to the coverage radii.

N_g	2	3	4	5	6	7
$\mathbf{E}[N_f \mid N_g]$	3.73	4.66	5.53	6.21	6.95	6.50
Occurrence	1.1%	29.9%	45.6%	19.0%	4.2%	0.2%

Table 5.1: Mean number of planned frequencies $\mathbf{E}[N_f \mid N_g]$ for each given N_g.

We are also interested in highlighting the strong point of our frequency auto-planning algorithm: the homogeneous utilization of frequencies. So we measure, for the same configuration parameters, the percentage of area covered by each frequency compared to the total covered area for our frequency auto-planning algorithm and for the greedy coloring algorithm. The percentages are given in mean over 1000 simulations. For an optimal utilization of frequencies, each frequency should cover the whole area, but it is not always achievable if there are not enough nodes to cover several times the whole area.

Number of planned frequencies	3	4	5	6	7	8
Frequency 1	77.8%	73.2%	67.6%	63.4%	59.0%	48.9%
Frequency 2	65.8%	58.6%	58.4%	54.4%	49.2%	45.9%
Frequency 3	37.7%	46.4%	46.7%	46.1%	47.2%	46.8%
Frequency 4		26.4%	34.7%	36.4%	37.2%	43.5%
Frequency 5			18.6%	28.0%	30.6%	36.6%
Frequency 6				16.2%	24.9%	28.6%
Frequency 7					13.2%	17.9%
Frequency 8						11.4%
Occurrence	6.3%	18.4%	30.0%	24.9%	13.6%	4.9%

Table 5.2: Mean percentage of covered area by each frequency with our algorithm

Number of planned frequencies	3	4	5	6	7
Frequency 1	98.1%	97.9%	97.7%	97.9%	98.2%
Frequency 2	57.7%	54.3%	55.7%	55.0%	52.3%
Frequency 3	17.7%	17.3%	19.1%	16.5%	18.0%
Frequency 4		4.1%	4.8%	4.2%	3.8%
Frequency 5			1.7%	1.2%	1.0%
Frequency 6				0.5%	1.0%
Frequency 7					1.0%
Occurrence	12.3%	44.7%	30.0%	11.3%	1.7%

Table 5.3: Mean percentage of covered area by each frequency with the greedy coloring algorithm

We can see in Table 5.2 and Table 5.3 the percentage of area covered by each frequency planned by our algorithm and the greedy coloring algorithm. The results are presented depending on the number of planned frequencies, we also indicate the number of simulations

these results concern for statistical relevance. For our algorithm, even if the percentage decreases with the order in which the frequencies are planned, which is logical, we can see that a rather homogeneous coverage is provided. Doing that, our algorithm maximizes the usage of each resource. We can see that for the greedy coloring algorithm, the frequencies re not all used equally, the first two frequencies are always a lot more planned than the following, the other frequencies are under-used.

5.4.2 Figures

We propose in this subsection figures illustrating the execution of our frequency auto-planning algorithm. In the first figure, Figure 5.1, we can see the initial cellular network and its coverage simplicial complex representation. In the cellular network figure, the black circles are the coverage radii, while the pink ones are the rejection radii.

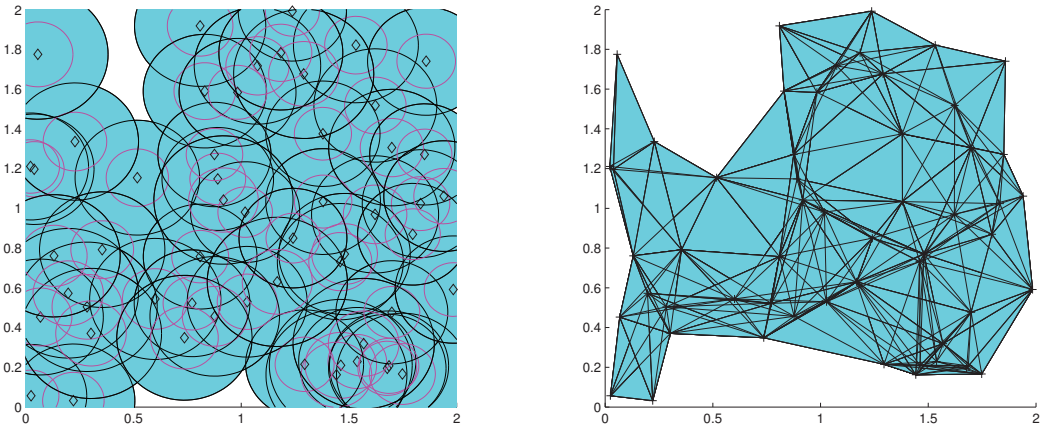


Figure 5.1: The cellular network and its coverage representation.

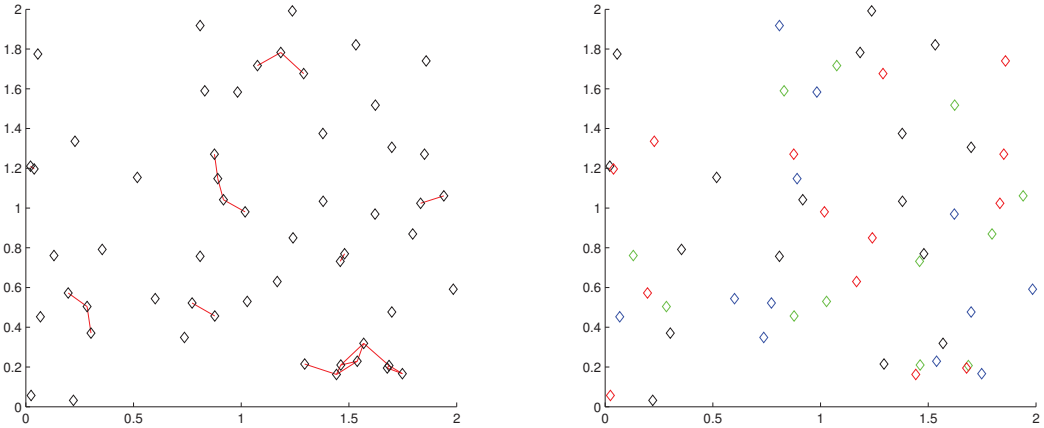


Figure 5.2: Interference relations and frequency planning scheme.

Then the interference graph is represented in Figure 5.2 next to the frequency planning scheme obtained by our algorithm for the configuration of Figure 5.1. In the left figure, vertices that can induce interference to each other are linked by an edge. In the figure on the right, a different color represent a different frequency. We can see that our algorithm has planned four frequencies (black, red, green and blue). Finally in Figure 5.3, we represent the covered area for each frequency of the previously obtained planning.

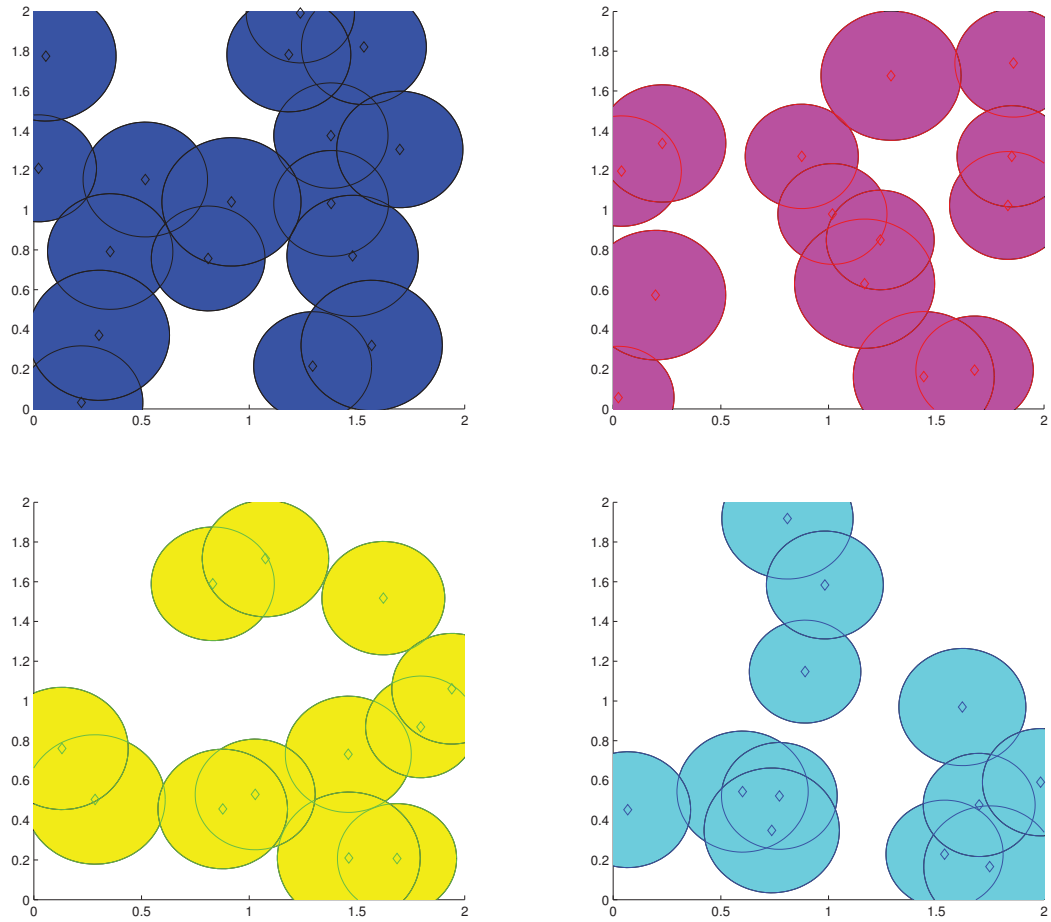


Figure 5.3: Coverage for each frequency.

Chapter 6

Self-optimization energy conservation algorithm

In this chapter, we propose an algorithm for the self-optimization during off-peak hours in future cellular networks. This algorithm is issued from significant modifications of the reduction algorithm for simplicial complexes of Chapter 3. Indeed, we propose an energy conservation algorithm of which the goal is not to reduce the number of vertices optimally without modifying the topology anymore. The energy conservation algorithm takes into account not only coverage, but also traffic, and is able to satisfy any required user demand. The performance of the algorithm is also discussed and compared to the optimal not always achievable solution.

6.1 Introduction

Among SON features, we can find the self-optimization functions, which defines the ability of the network to adapt its behavior to different traffic scenarii. Indeed, in LTE cellular networks, eNode-Bs (eNB) have multiple configurable parameters. An example is output power, so cells size can be configured when capacity is the limitation rather than coverage. Moreover, fast and reliable X2 communication interfaces connect eNBs. So the whole network has the capability to adapt to different traffic situations. Then, users traffic can be observed via eNBs and User Equipments (UEs) measurements. Therefore, the self-optimization functions aim at using this traffic observations to adapt the whole network, and not only each cell independently, to the traffic situation. As previously, ANR information is used to build the simplicial complex representation of the network.

One case where self-optimization is often needed is the adaptation to off-peak hours. Typically a cellular network is deployed to match daily peak hours traffic requirements. Therefore during off-peak hours, the network is daily under-used. This leads to a huge unneeded amount of energy consumption. An idea is thus to switch-off some of the eNBs during off-peak hours, while other eNBs adjust their configuration parameters to keep the entire area covered. In case of a growth in traffic, the switched-off eNBs could be woken up to satisfy the users demand.

In this chapter, we enhance our reduction algorithm to satisfy any user traffic. Our reduction algorithm, as it is presented in Chapter 3,only satisfies perfect connectivity and coverage. However, in cellular networks, especially in urban areas, coverage is not the limiting factor, capacity is. So the optimal solution is not optimal coverage anymore, but

depends on the required traffic. We use simplicial complex representation to represent a cellular network and its topology. Then we use a modified reduction algorithm to reach an optimally used network. The performance of our energy conservation algorithm is discussed and compared to the minimun configuration required by traffic.

The first approach to energy efficiency in under-used wireless network is to switch-off the non-needed nodes. It is done based on cell traffic for Femtocells in [9]: after a given period of time in idle mode, the node puts itself on stand-by. However, if one wants to take into account the whole network, it has to consider the coverage of the network before disconnecting, which is not the case of Femtocells, which are by definition redundant to the base stations. Without considerations of traffic, we proposed in Chapter 3 an algorithm that reduce power consumption in wireless networks by putting on standby some of the nodes without impacting the coverage. We can also cite [17] that proposes a game-theoretic approach in which nodes are put on standby according to a coverage function, but unmodified coverage is not guaranteed. In both these works, only coverage is taken into account. This approach could eventually fit the requirements of cellular network in non-urban cells, if their deployment has coverage redundancy. But it is not valid for urban cells, where it is not coverage but capacity that delimits cells.

The remainder of this chapter is organized as follows. First, we provide related work on self-optimization for future cellular networks in Section 6.2. Then, our self-optimization algorithm is presented, including a full algorithm description, in Section 6.3. Finally, performance results and examples of simulation are given in Section 6.4.

6.2 Self-optimization in future cellular networks

In order to ensure that future cellular networks are still efficient in terms of both Quality of Service (QoS) and costs, the self-configuration is not sufficient. Indeed, future cellular networks have the ability to adjust their parameters to match different traffic situation. Periodic optimization based on log reports, and operated centrally is not a effective solution in terms of speed and costs. That is why we need self-optimization. Self-optimization can be classified in three types depending on its goal.

First we can consider load balancing optimization. There is multiple ways to adapt a cellular network to different loads: it is for example possible to adapt the resources available in different nodes. These schemes were mainly introduced for GSM [21], and then CDMA [3], but the universal frequency reuse of LTE and LTE-Advanced diminishes their applicability. Then one can adapt the traffic strategy with admission controls on given cells and forced handovers [35]. However, as the previous solution, it is not very suitable for LTE and LTE-A which require hard handover. Finally it is possible to modify the coverage of a node by changing either its antennas radiation pattern [31] or the output power [22]. We use this latter approach to reach an optimal result for our algorithm: we adapt the coverage radius of each node to be the minimum required to cover a given area.

The second type self-optimization is the capacity and coverage adaptation via the use of relay nodes [49], while the third is interference optimization. Our algorithm could lie in this third category as the simplest approach towards interference control is switching off idle nodes [9]. Our algorithm goes a little bit further by adapting the switching-off of the nodes to the whole network situation, and not only the traffic in a given cell. Indeed, in case of low traffic, some nodes can cover the area of other ones.

6.3 Energy conservation algorithm

6.3.1 Main idea

We consider a cellular network and represent it by its associated coverage simplicial complex. Since we want to use a minimum number of nodes, we consider for each one its maximum coverage radius. Then we construct the Čech complex corresponding to the set of nodes with their maximum covered cells. In this application, we are not only interested in the topology of the network but also by the characterization of the clusters of nodes by simplices. Indeed, we intend to optimize the number of switched-on nodes for the user traffic requested in a given area. This area will be defined by the nodes that are serving it.

Before beginning the algorithm, we have to define how we will represent user demand. To do this we will create groups of vertices. These groups have to make sense geometrically: they need to represent clusters of vertices. So, every group will be defined by a simplex. Then these groups have to be defined such that each vertex is part of one group exactly. To consider user traffic, we will assign a traffic to each group: for every group of k vertices/nodes, we will draw uniformly an integer number between zero and k that is the number of required nodes to keep switched-on. This is what will represent the required QoS for the cellular network. This QoS metric is quite artificial, but the algorithm can take into account any QoS in terms of number of ressources required from a given pool of ressources. This one is the easier to implement if we want to consider that a same pool of ressources is provided by nodes geographically close to each other with no location information.

As in Chapter 3, the algorithm begins by the computation of the Betti numbers, since we do not want to modify the network's topology by turning off nodes. Then we compute the triangle degrees and the vertices indices of the coverage reduction algorithm. As in Chapter 5, the order for the removal of vertices follows the principle of the coverage reduction algorithm. But the breaking point is different. Instead of stopping when the area is covered by a minimum number of vertices, the algorithm stops when each group has been reduced to its required QoS. Then given this configuration of switched-on nodes, i.e. kept vertices, the algorithm tries to reduce as much as possible the coverage radius of each node without creating a coverage hole. The order in which the coverage radii are examined is random. Finally, we obtain a configuration of vertices that defines the nodes to keep switched-on that is optimal. From the first part of the algorithm, we ensure that enough nodes are kept on to satisfy the user demand. Then, from the second part, we ensure that no energy is spend uselessly by optimizing the size of the serving cells.

6.3.2 Algorithm description

We give in Algorithm 8 the full energy conservation algorithm. It requires the set of vertices ω and their coverage radii, then returns the list of kept vertices with their new coverage radii. We can see that the breaking point of the 'while' loop of the algorithm takes the 'QoS' parameter into account. Then a vertex can be removed if and only if it does not modify the number of connected component and the number of coverage holes, and it is not needed for the QoS requirements. Coverage radii reduction for each vertex is done in the last loop.

Algorithm 8 Energy conservation algorithm

Require: Set ω of N vertices, for each vertex v its coverage radius r_v.

Computation of the Čech complex $X = \mathcal{C}_r(\omega)$
Creation of the list of boundary vertices L_C
Computation of $\beta_0(X)$ and $\beta_1(X)$
Computation of $D_1(X), \ldots, D_{s_2}(X)$
Computation of $I[v_1(X)], \ldots, I[v_{s_0}(X)]$
for all $v \in L_C$ **do**
  $I[v] = -1$
end for
$I_{\max} = \max\{I[v_1(X)], \ldots, I[v_{s_0}(X)]\}$
$N_{\text{group}} = 0$
for all Simplex $S_k \in X$ from largest to smallest **do**
  if $\forall v \in S_k, \text{Group}(v) == 0$ **then**
    $N_{\text{group}} = N_{\text{group}} + 1$
    $\forall v \in S_k \text{Group}(v) = N_{\text{group}}$
    $\text{Size}(N_{\text{group}}) = k + 1$
    Draw $\text{QoS}(N_{\text{group}})$ among $\{0, \ldots, k+1\}$
  end if
end for
while $I_{\max} > 2$ **and** Size $\geq$ QoS **do**
  Draw w a vertex of index $I_{\max}$
  $X' = X \backslash \{w\}$
  Computation of $\beta_0(X'), \beta_1(X')$
  if $\beta_0(X') \neq \beta_0(X)$ **and** $\beta_1(X') \neq \beta_1(X)$**and** $\text{Size}(\text{Group}(w)) \leq \text{QoS}(\text{Group}(w))$ **then**
    $I[w] = -1$
  else
    $\text{Size}(\text{Group}(w)) = \text{Size}(\text{Group}(w)) - 1$
    Computation of $D_1(X'), \ldots, D_{s'_{k_0}}(X')$
    for $i = 1 \rightarrow s'_0$ **do**
      if $I[v_i(X')] == I_{\max}$ **then**
        Recomputation of $I[v_i(X')]$
      end if
    end for
    $I_{\max} = \max\{I[v_1(X')], \ldots, I[v_{s'_0}(X')]\}$
    $X = X'$
  end if
end while
for all $v \in X$ taken in random order **do**
  $X' = X$
  while $\beta_0(X') == \beta_0(X)$**and** $\beta_1(X') == \beta_1(X)$ **do**
    Reduce r_v
  end while
  $X = X'$
end for
return List of kept vertices v and their coverage radii r_v.

For our simulations, we choose to give advantage to the larger simplices for the constitution of the QoS groups. Thus, the first group will consist of the largest simplex, or one randomly chosen among the largest ones, then simplices of smaller size will become groups if none of their 0-face is already part of another group, until every vertex is part of a group. It is possible to consider other rules for the constitution of the groups, but it has to follow two conditions: every vertex must pertain to at least a group, and must not pertain to more than a group. The groups are represented by the variable 'Group', that for each vertex gives its group number. Then 'QoS' represents the minimum number of vertices required for a given group, while 'Size' represents its number of vertices.

6.4 Simulation results

For simulation reasons, we only consider the Vietoris-Rips complex in the 2-dimensional case. We choose the groups to be simplices from the largest to the smallest as in the algorithm description.

6.4.1 Performance

We compare the performance of our algorithm to an optimal, not always achievable solution. Indeed we do not know of a energy conservation algorithm that switch-off vertices during off-peak hours while maintaining coverage. We compare the number of switched-on vertices after the execution of our energy conservation algorithm, to the number of vertices needed for the QoS, given by the 'QoS' parameter. It is important to note that this optimal solution is not always achievable since it does not take into account that the area is to stay covered. Some vertices have to be kept for traffic reasons, while other are kept to maintain connectivity and/or coverage.

Our simulation results are computed on 1000 configurations of Poisson point processes of intensity $\lambda = 6$ on a square of size $a = 2$ with a fixed boundary of vertices. The coverage radii are sampled uniformly between $a/10$ and a^2/λ, except for the boundary vertices for which it was set to $1/\sqrt{\lambda}$. For each group the 'QoS' number is a sampled integer between zero and the size of the group. We denote by N_o the optimal number of vertices, and by N_k the number of kept vertices with our energy conservation algorithm. First we compute the percentage of simulations for which we have a given difference between the obtained number and the optimal number of kept vertices.

$N_k - N_o$	0	1	2	3	4	5	6	7	$8-10$
Occurrence	8.3%	18.7%	24.5%	21.7%	13.6%	8.6%	3.3%	1.0%	0.3%

Table 6.1: Occurrences of given differences between N_k and N_o.

We can see in Table 6.1 the percentage of simulations in which the number of kept vertices is different from the optimal number of vertices. For 8.6% of the simulations the optimal number is reached. In 87.1% of our simulations the difference between the optimal and the effective number of kept vertices is smaller than 4, and it never exceeds 10. The number of boundary vertices is 12, these vertices are not removable (in order to never shrink the covered area). So less than 10 vertices needed to cover the whole area on top of the needed vertices for the traffic seems plausible.

To have more advanced comparison, for the 1000 configurations, we compute the optimal number of vertices. Then for each optimal number of vertices N_o that occurred the most, we compute the mean number of kept vertices $\mathbf{E}[N_k|N_o]$ over the simulations which have N_o for optimal number. The results are given in Table 6.2. For comparison, we also compute the difference between N_o and N_k in percent. We finally indicate which percent of our 1000 simulations these cases occur to show the relevance of these statistical results.

N_o	20	21	22	23	24	25	26	27	28
$\mathbf{E}[N_k\|N_o]$	23.40	24.14	24.91	25.06	26.43	27.37	27.91	28.84	29.56
Difference	17.0%	14.9%	13.2%	8.9%	10.1%	9.5%	7.3%	6.8%	5.6%
Occurrence	7.0%	7.3%	7.9%	6.9%	6.1%	6.8%	7.9%	6.8%	5.4%

Table 6.2: Mean number of kept vertices $\mathbf{E}[N_k|N_o]$ for a given optimal number N_o.

We can see in Table 6.2 that the more vertices are needed, the less difference there is between our result and the optimal one. Indeed, if more vertices are needed, there is a great chance that these vertices can cover the whole area.

6.4.2 Figures

We propose in this subsection figures illustrating the execution of our energy conservation algorithm. In the first figure, Figure 6.1, we can see the initial cellular network and its coverage simplicial complex representation. The vertices have different coverage radii, and there is no fixed boundary. The boundary vertices are in red.

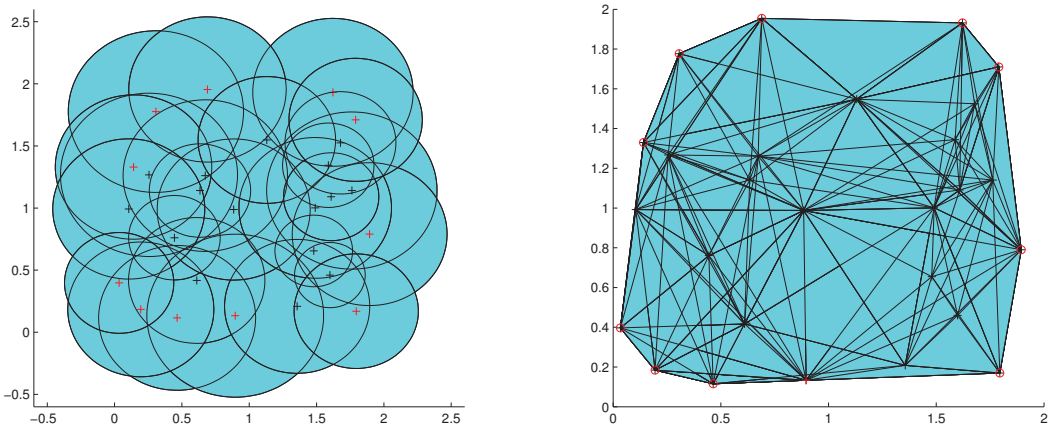
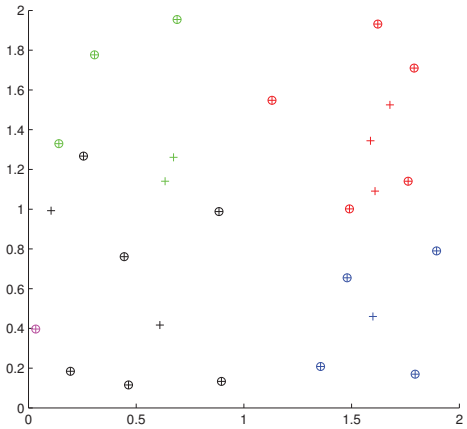


Figure 6.1: The cellular network and its coverage representation.

Then for the configuration of Figure 6.1, we represent the groups of nodes in different colors, and give a table with their corresponding QoS and size in Figure 6.2. The kept vertices are circled. Finally in Figure 6.3, we can see the final configuration of the cellular network with the optimized coverage radii and its corresponding simplicial complex for the configuration of Figure 6.1.



Color	Black	Red	Blue	Green	Magenta
QoS	6	5	4	2	0
Size	8	8	5	5	1

Figure 6.2: Groups of QoS.

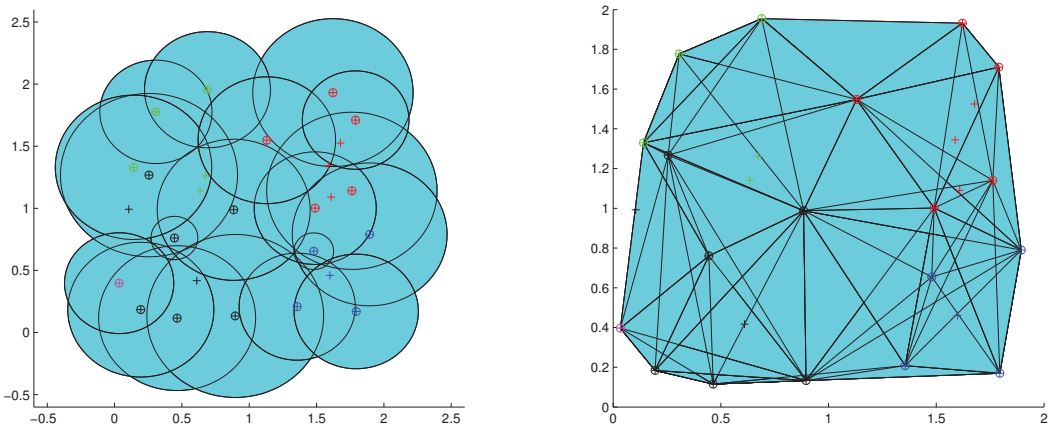


Figure 6.3: Final configuration.

Chapter 7

Disaster recovery algorithm

In this chapter, we present an algorithm for the recovery of wireless networks after a disaster. Considering a damaged wireless network, presenting coverage holes or/and many disconnected components, we propose a disaster recovery algorithm which repairs the network. It provides the list of locations where to put new nodes in order to patch the coverage holes and mend the disconnected components. In order to do this we first consider the simplicial complex representation of the network, then the algorithm adds supplementary vertices in excessive number, and afterwards runs the reduction algorithm from Chapter 3 to reach an optimal result. One of the novelty of this work resides in the proposed method for the addition of vertices. We use a determinantal point process: the Ginibre point process which has inherent repulsion between vertices, and has never been simulated before for wireless networks representation. We compare both the determinantal point process addition method with other vertices addition methods, and the whole disaster recovery algorithm to the greedy algorithm for the set cover problem. This chapter is issued from [87].

7.1 Introduction

The third and last of the SON functions is self-healing. In future cellular networks, nodes would be able to appear and disappear at any time. Since the cellular network is not only constituted of operators base stations anymore, the operator does not control the arrivals or departures of nodes. But the disappearances of nodes can be more generalized: for example in case of a natural disaster (floods, earthquakes or tsunamis...). The self-healing functions aim at reducing the impacts from the failures of nodes must it be in isolated cases, like the turning off of a Femtocells, or more serious cases where the whole network is damaged. We are interested in this latter case.

In case of a disaster, a wireless network can be seriously damaged: some of its nodes can be completely destroyed. However such networks are not necessarily built with redundancy and then can be sensitive to such damages. Coverage holes can appear resulting in no signal for communication or no monitoring at all of a whole area. Paradoxically, reliable and efficient communication and/or monitoring is especially needed in such situations. Therefore, solutions for damage recovery for the coverage of wireless networks are much needed. As in Chapter 3, we use simplicial complex representation for wireless network as a tool to compute mathematically the coverage of the network.

In this chapter, we present a homology based algorithm for disaster recovery of wireless

networks. We represent wireless networks with Čech simplicial complexes characterizing their coverage. Given a set of vertices and their coverage radius, our algorithm first adds supernumerary vertices in order to patch every existing coverage hole and connect every components, then runs an improved version of the reduction algorithm presented in Chapter 3 in order to reach an optimal result with a minimum number of added vertices. At the end, we obtain the locations in which to put new nodes. For the addition of new vertices, we first compared three usual methods presenting low complexity: grid positioning, uniform positioning and the use of the Sobol sequence, a statistical tool built to provide uniform coverage of the unit square. Then, we propose the use of a determinantal point process: the Ginibre point process. This process has the ability to create repulsion between vertices, and therefore has the inherent ability to locate areas with low density of vertices: namely coverage holes. Therefore using this process, we will optimally patch the damaged wireless network. The use and simulation of determinantal point processes in wireless networks is new, and it provides tremendous results compared to classic methods.We finally compared our whole distaster recovery algorithm performance to the classic recovery algorithm performance: the greedy algorithm for the set cover problem.

This is the first algorithm that we know of that adds too many vertices then remove them to reach an optimal result instead of adding the exact needed number of vertices. This, first, allows flexibility in the choice of the new vertices positions, which can be useful when running the algorithm in a real life scenario. Indeed, in case of a disaster, every locations are not always available for installing new nodes and preferring some areas or locations can be done with our algorithm. The originality of our work lies also in the choice of the vertices addition method we suggest. On top of flexibility, our algorithm provides a more reliable repaired wireless network than other algorithms. Indeed, adding the exact needed number of vertices can be optimal mathematically speaking but it is very sensitive to the adherence of the nodes positions chosen by the algorithm. To compare our work to literature, we can see that the disaster recovery problem can be viewed as a set cover problem. It suffices to define the universe as the area to be covered and the subsets as the balls of radii the coverage radii. Then the question is to find the optimal set of subsets that cover the universe, considering there are already balls centered on the existing vertices. A greedy algorithm can solve this problem as explained in [19]. We can see in [41] that ϵ-nets also provide an algorithm for the set cover problem via a sampling of the universe. We can also cite landmark-based routing, seen in [34] and [7], which, using furthest point sampling, provides a set of nodes for optimal routing that we can interpret as a minimal set of vertices to cover an area.

The remainder of this chapter is structured as follows: after a section on related work we present the main idea of our disaster recovery algorithm in Section 7.3. Then in Section 7.4, we compare usual vertices addition methods. In Section 7.5, we expose the determinantal method for new vertices addition. Finally in Section 7.6 we compare the performance of the whole disaster recovery algorithm with the greedy algorithm for the set cover problem.

7.2 Recovery in future cellular networks

The first step of recovery in cellular networks is the detection of failures. The detection of the failure of a cell occurs when its performance is considerably and abnormally reduced. In [65], the authors distinguish three stages of cell outage: degraded, crippled and catatonic. This last stage matches with the event of a disaster when there is complete outage of the damaged cells. After detection, compensation from other nodes can

occur through relay assisted handover for ongoing calls, adjustments of neighboring cell sizes via power compensation or antenna tilt. In [5], the authors not only propose a cell outage management description but also describe compensation schemes. These steps of monitoring and detection, then compensation of nodes failures are comprised under the self-healing functions of future cellular networks described in [2].

In this work, we are interested in what happens when self-healing is not sufficient. In case of serious disasters, the compensation from remaining nodes and traffic rerouting might not be sufficient to provide service everywhere. In this case, the cellular network needs a manual intervention: the adding of new nodes to compensate the failures of former nodes. However a traditional restoration with brick-and-mortar base stations could take a long time, when efficient communication is particularly needed. In these cases, a recovery trailer fleet of base stations can be deployed by operators [64], it has been for example used by AT&T after 9/11 events. But a question remains: where to place the trailers carrying the recovery base stations. An ideal location would be adjacent to the failed node. However, these locations are not always available because of the disaster, and the recovery base stations may not have the same coverage radii than the former ones. Therefore a new deployment for the recovery base stations has to be decided, in which one of the main goal is complete coverage of damaged area. This becomes a mathematical set cover problem. It can be solved by a greedy algorithm [19], ϵ-nets [41], or furthest point sampling [7,34]. But these mathematical solutions provide an optimal mathematical result that do not consider any flexibility at all in the choosing of the new nodes positions, and that can be really sensitive to imprecisions in the nodes positions.

7.3 Main idea

We consider a damaged wireless network presenting coverage holes with a fixed boundary, in order to know the domain to cover, of which we can see an example in Figure 7.1.

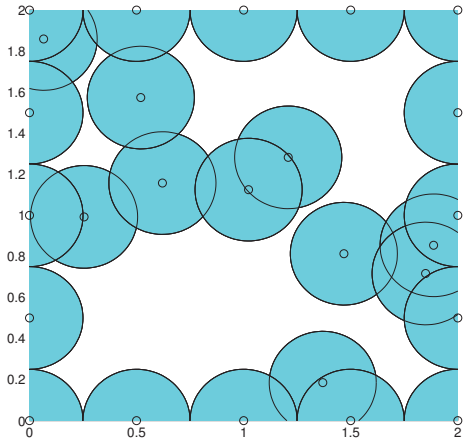


Figure 7.1: A damaged wireless network with a fixed boundary.

We consider as inputs the set of existing vertices: the nodes of a damaged wireless

network, and their coverage radii. We also need a list of boundary nodes, these nodes can be fictional, but we need to know the whole area that is to be covered. Then we build the Čech complex characterizing the coverage of the wireless network, the Betti number β_1 of the Čech complex counting the number of coverage holes of the wireless network. We restrict ourselves to wireless networks with a fixed coverage radius r, but it is possible to build the Čech complex of a wireless networks with different coverage radii using the intersection of different size coverage balls.

The algorithm begins by adding new vertices in addition to the set of existing vertices presenting coverage holes. We suggest here the use of three usual methods, and the new determinantal addition method. As we can see in Section 7.4, it is possible to consider any vertices addition methods must they be deterministic or random based: flexibility is one of the greatest advantage of our algorithm. In particular, it is possible to consider a method with pre-defined positions for some of the vertices in real-life scenarii.

For any non-deterministic method, we choose that the number of added vertices, that we denote by N_a, is determined as follows. First, it is set to be the minimum number of vertices needed to cover the whole area minus the number of existing vertices. This way, we take into account the number of existing vertices, that we denote by N_i. Then the Betti numbers β_0 and β_1 are computed via linear algebra thanks to the simplicial complex representation. If there is still more than one connected component, and coverage holes, then the number of added vertices N_a is incremented with a random variable u following an exponential growth:

- $N_a := \lceil \frac{a^2}{\pi r^2} \rceil - N_i$.
- After adding the N_a vertices, if $\beta_0 \neq 1$ or $\beta_1 \neq 0$,
  Then, $N_a = N_a + u$, and $u = 2 * u$.

The next step of our approach is to run the coverage reduction algorithm from [86] which maintains the topology of the wireless network: the algorithm removes vertices from the simplicial complex without modifying its Betti numbers. At this step, we remove some of the supernumerary vertices we just added in order to achieve an optimal result with a minimum number of added vertices.

We give in Algorithm 9 the outline of the algorithm. The algorithm requires the set of initial vertices ω_i, the fixed coverage radius r, as well as the list of boundary vertices L_b. It is important to note that only connectivity information is needed to build the Čech complex.

7.4 Vertices addition methods

In this section, we propose three vertices addition methods. The aim of this part of the algorithm is to add enough vertices to patch the coverage of the simplicial complex, but the less vertices the better since the results will be closer to the optimal solution. We consider grid and uniform positioning which require minimum simulation capacities and are well known in wireless networks management. Then we propose the use of the Sobol sequence, which is a statistical tool built to provide uniform coverage of the unit square. The grid method is deterministic, so the number of added vertices as well as their position are set. The uniform method is random, the number of added vertices is then computed as presented in Section 7.3.

Algorithm 9 Disaster recovery algorithm

Require: Set of vertices ω_i, radius r, boundary vertices L_b
  Computation of the Čech complex $X = \mathcal{C}_r(\omega_i)$
  $N_a = \lceil \frac{a^2}{\pi r^2} \rceil - N_i$
  Addition of N_a vertices to X following chosen method
  Computation of $\beta_0(X)$ and $\beta_1(X)$
  $u = 1$
  while $\beta_0 \neq 1$ or $\beta_1 \neq 0$ **do**
    $N_a = N_a + u$
    $u = 2 * u$
    Addition of N_a vertices to X following chosen method
    Computation of $\beta_0(X)$ and $\beta_1(X)$
  end while
  Coverage reduction algorithm on X
  return List L_a of kept added vertices.

7.4.1 Grid

The first method we suggest ensures perfect coverage: the new vertices are positioned along a square grid in a lattice graph where the distance between two neighboring vertices is $\sqrt{2}r$. The number of vertices is set. Therefore this method is completely independent from the initial configuration. We can see an example of the grid vertices addition method on the damaged network of Figure 7.1 in Figure 7.2. Existing vertices are black circles while added vertices are red plusses.

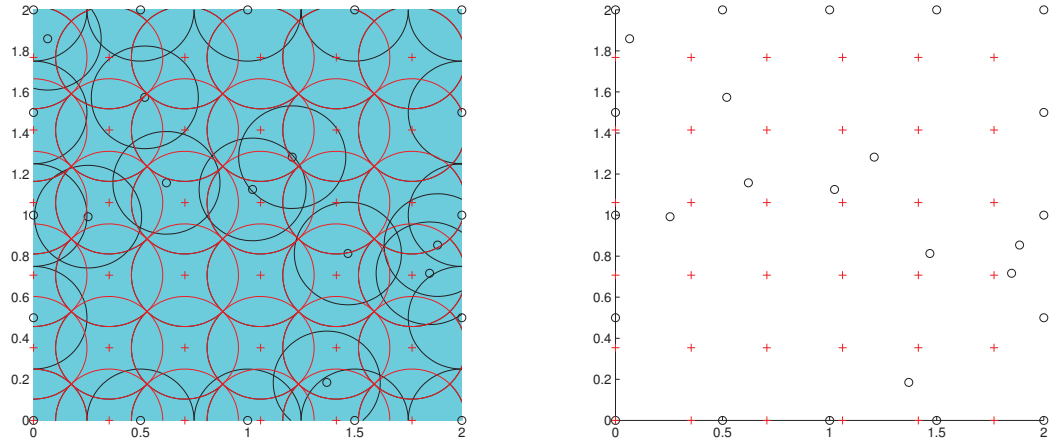


Figure 7.2: With the grid addition method.

7.4.2 Uniform

Here, the number of added vertices N_a is computed accordingly to the method presented in Section 7.3, taking into account the number of existing vertices N_i. Then the N_a vertices are sampled following a uniform law on the entire domain. An obtained configuration with this method on the network of Figure 7.1 is shown in Figure 7.3.

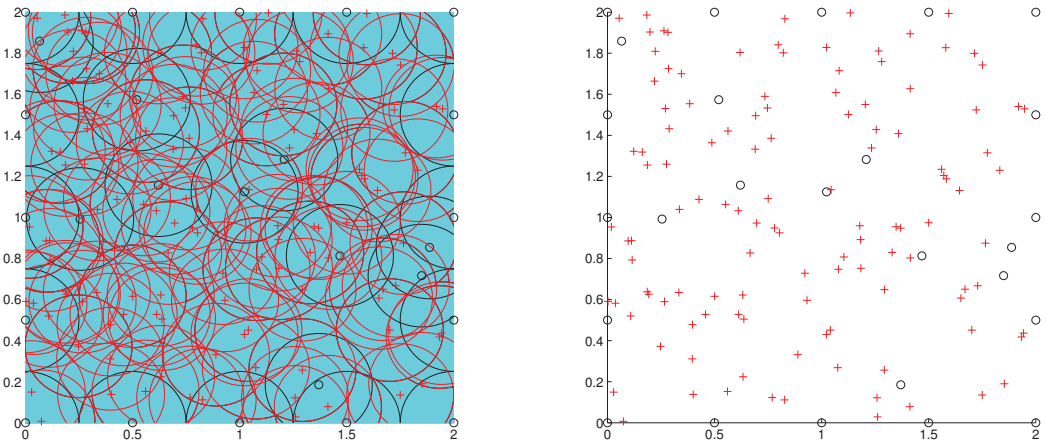


Figure 7.3: With the uniform addition method.

7.4.3 Sobol sequence

Thanks to this method, we are able to take into account the positions of the new added vertices. The Sobol sequence is a statistical tool used to provide uniform coverage of the unit square. Thus, vertices positioned with the Sobol sequences reach complete coverage faster than uniform positioning because the aggregation phenomenon is statistically avoided. The Sobol initialization set is known, for instance on a square the first position is the middle of the square, then come the middles of the four squares included in the big square, etc. To randomize the positions drawn, the points are scrambled. Therefore the complexity of the simulation of this method is really low.

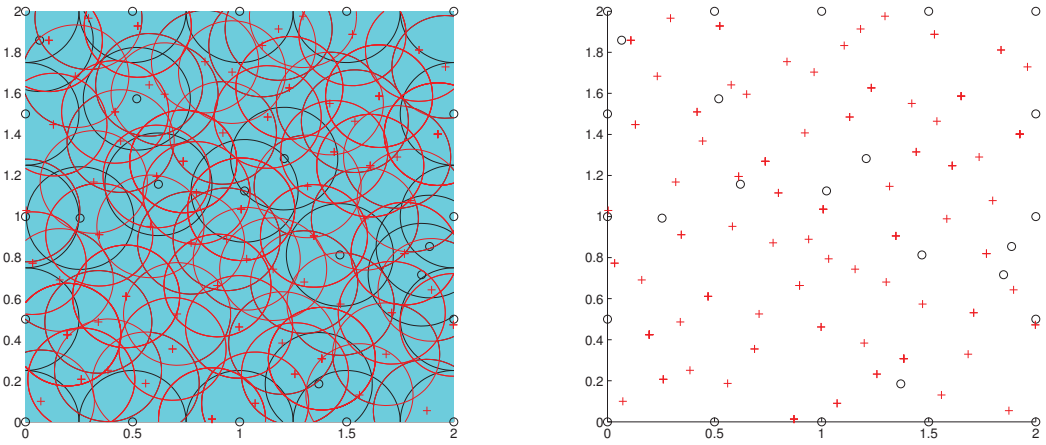


Figure 7.4: With the Sobol sequence addition method.

For our simulation, we used the set of initialization numbers provided by Bratley and Fox in [16]. Then we scrambled the points produced with the random method described in [60]. An example of this method is given in Figure 7.4.

7.4.4 Comparison

We can compare the vertices addition method presented here along two variables: their complexity and their efficiency. First, we compare the complexities of the two methods. They all are of complexity $O(N_a)$: computations of N_a positions, and the Sobol method scramble N_a positions already known by most simulation tools. For the random methods we have to add the complexity of computing the coverage via the Betti numbers, which is of the order of the number of triangles times the number of edges that is $O((N_a+N_i)^5(\frac{r}{a})^6)$ for a square of side a according to [26].

To compare the methods efficiency we count the number of vertices each have to add on average to reach complete coverage. The grid method being determinist, the number of added vertices is constant: $N_a = (\lfloor \frac{a}{\sqrt{2}r} \rfloor + 1)^2$ for a Čech complex or $N_a = (\lfloor \frac{a}{2r} \rfloor + 1)^2$ for a Vietoris-Rips complex which is an approximation of the Čech complex easier to simulate. We can see in Table 7.1 the mean number of added vertices on 1000 simulations for each method in different scenarii on a square of side $a = 1$ with coverage radius $r = 0.25$, and a Vietoris-Rips complex. Scenarii are defined by the mean percentage of area covered before running the recovery algorithm: if there are many or few existing vertices, and thus few or many vertices to add. We need to note that number of added vertices is computed following our incrementation method presented in Section 7.3 and these results only concern the vertices addition methods before the reduction algorithm runs.

Percentage of area initially covered	20%	40%	60%	80%
Grid method	9.00	9.00	9.00	9.00
Uniform method	32.76	29.18	23.71	16.46
Sobol sequence method	29.80	29.04	24.16	15.98

Table 7.1: Mean number of added vertices $\mathbf{E}[N_a]$

We can observe that the Sobol sequence method gives better results than the uniform method except for a 60% of covered area. Since the Sobol sequence partitions uniformly the area, it takes advantages that there are not too many existing vertices in the scenarii with small percentage of covered area. The grid method is mathematically optimal for the number of added vertices to cover the whole area, however it is not optimal in a real life scenario where positions can not be defined with such precision, and any imprecision leads to a coverage hole. This method fares even or better both in complexity and in number of added vertices.

7.5 Determinantal addition method

In this section, we present the determinantal method and compare it to the three methods presented in Section 7.4.

7.5.1 Definitions

The most common point process in wireless network representation is the Poisson point process. However in this process, conditionally to the number of vertices, their positions are independent from each other (as in the uniform positioning method presented in Section 7.4). This independence creates some aggregations of vertices, that is not convenient for

our application. That is why we introduce the use of determinantal point processes, in which the vertices positions are not independent anymore. We can see in Figure 7.5 the differences between points sampled uniformly and sampled with repulsion on the unit disk. We can see that the independence of vertices positions of the Poisson point process creates some clusters, while determinantal processes provide a more uniform coverage.

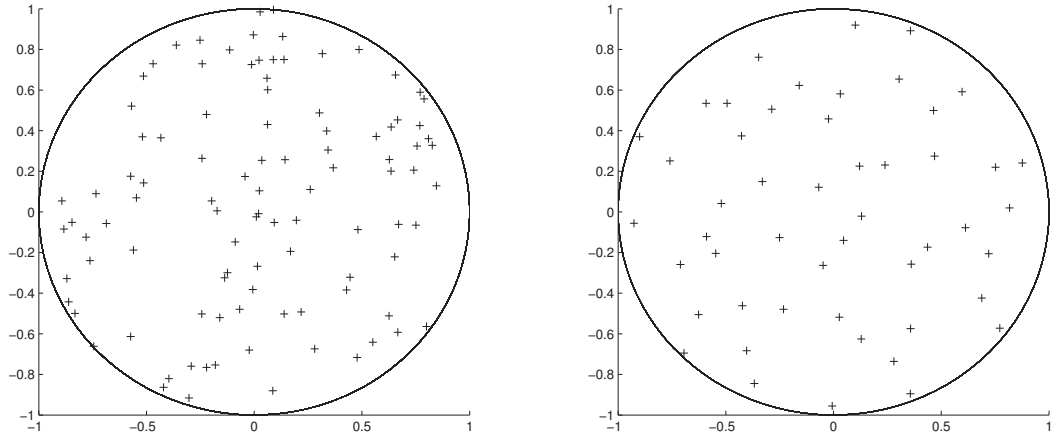


Figure 7.5: Uniform vs determinantal sampling.

General point processes can be characterized by their so-called Papangelou intensity. Informally speaking, for x a location, and ω a realization of a given point process, that is a set of vertices, $c(x,\omega)$ is the probability to have a vertex in an infinitesimal region around x knowing the set of vertices ω. For Poisson process, $c(x,\omega) = 1$ for any x and any ω. A point process is said to be repulsive (resp. attractive) whenever $c(x,\omega) \geq c(x,\zeta)$ (resp. $c(x,\omega) \leq c(x,\zeta)$) as soon as $\omega \subset \zeta$. For repulsive point process, that means that the greater the set of vertices, the smaller the probability to have an other vertex.

Among repulsive point processes, we are in particular interested in determinantal processes:

Definition 40 (Determinantal point process)**.** *Given X a Polish space equipped with the Radon measure μ, and K a measurable complex function on X^2, we say that N is a determinantal point process on X with kernel K if it is a point process on X with correlation functions $\rho_n(x_1,\ldots,x_n) = \det(K(x_i,x_j)_{1\leq i,j\leq n})$ for every $n \geq 1$ and $x_1,\ldots,x_n \in X$.*

We can see that when two vertices x_i and x_j tends to be close to each other for $i \neq j$, the determinant tends to zero, and so does the correlation function. That means that the vertices of N repel each other. There exist as many determinantal point processes as functions K. We are interested in the following:

Definition 41 (Ginibre point process)**.** *The Ginibre point process is the determinantal point process with kernel $K(x,y) = \sum_{k=1}^{\infty} B_k \phi_k(x)\overline{\phi_k(y)}$, where $B_k, k = 1,2,\ldots,$ are k independent Bernoulli variables and $\phi_k(x) = \frac{1}{\sqrt{\pi k!}} e^{\frac{-|x|^2}{2}} x^k$ for $x \in \mathbb{C}$ and $k \in \mathbb{N}$.*

The Ginibre point process is invariant with respect to translations and rotations, making it relatively easy to simulate on a compact set. Moreover, the repulsion induced by a

Ginibre point process is of electrostatic type. The principle behind the repulsion lies in the probability density used to draw vertices positions. The probability to draw a vertex at the exact same position of an already drawn vertex is zero. Then, the probability increases with increasing distance from every existing vertices. Therefore the probability to draw a vertex is greater in areas the furthest away from every existing vertices, that is to say in coverage holes. Therefore, added vertices are almost automatically located in coverage holes thus reducing the number of superfluous vertices.

7.5.2 Simulation

Using determinantal point processes allows us to not only take into account the number of existing vertices, via the computation of N_a, but we also take into account the existing vertices positions, then every new vertex position as it is added. It suffices to consider the N_i existing vertices as the N_i first vertices sampled in the process, then each vertex is taken into account as it is drawn. The Ginibre process is usually defined on the whole plane thus we needed to construct a process with the same repulsive characteristics but which could be restricted to a compact set. Moreover, we needed to be able to set the number of vertices to draw. Due to space limitations, we will not delve into these technicalities but they are developed in [27]. We can see a realisation of our simulation for the recovery of the wireless network of Figure 7.1 in Figure 7.6.

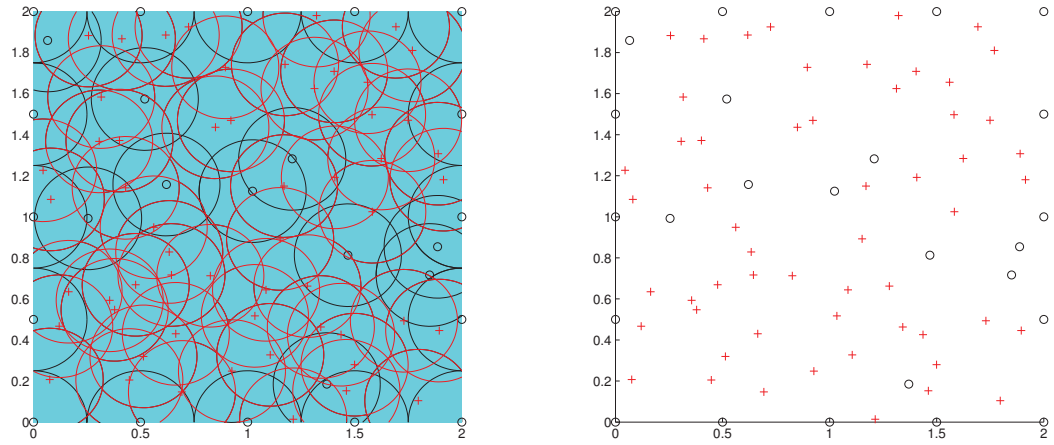


Figure 7.6: With the determinantal addition method with a Ginibre basis.

7.5.3 Comparison

We now compare the determinantal vertices addition method to the methods presented in Section 7.4.

As for the complexity, since the determinantal method takes into account the position of both existing vertices and randomly added vertices, it is the more complex. First taking into account the existing vertices positions is of complexity $O(N_i^2)$, then the position drawing with the rejection sampling is of complexity $O(N_a(N_a + N_i))$ at most. Thus we have a final complexity of $O(N_i^2 + N_a^2 + N_a N_i))$. To which we add the Betti numbers computation complexity: $O((N_a + N_i)^5(\frac{r}{a})^6)$.

We give in Table 7.2 the comparison between the mean number of added vertices for the three methods. The simulation parameters being the same as Section 7.4. The determinantal method provides the best results in all scenarii among the random methods by far. And it is the best method among all for the most covered scenario.

Percentage of area initially covered	20%	40%	60%	80%
Grid method	9.00	9.00	9.00	9.00
Uniform method	32.76	29.18	23.71	16.46
Sobol sequence method	29.80	29.04	24.16	15.98
Determinantal method	14.07	12.52	9.67	5.73

Table 7.2: Mean number of added vertices $\mathbf{E}[N_a]$

7.6 Performance comparisons

After adding the new vertices, according to Algorithm 9, we run the coverage reduction algorithm described in Chapter 3. Therefore, from the N_a added vertices we keep only what we call the final number of added vertices $N_f < N_a$. We can see in Figure 7.7 an execution of the reduction algorithm on the intermediate configuration of Figure 7.6. Removed vertices are represented by blue diamonds. We now compare the performance results of

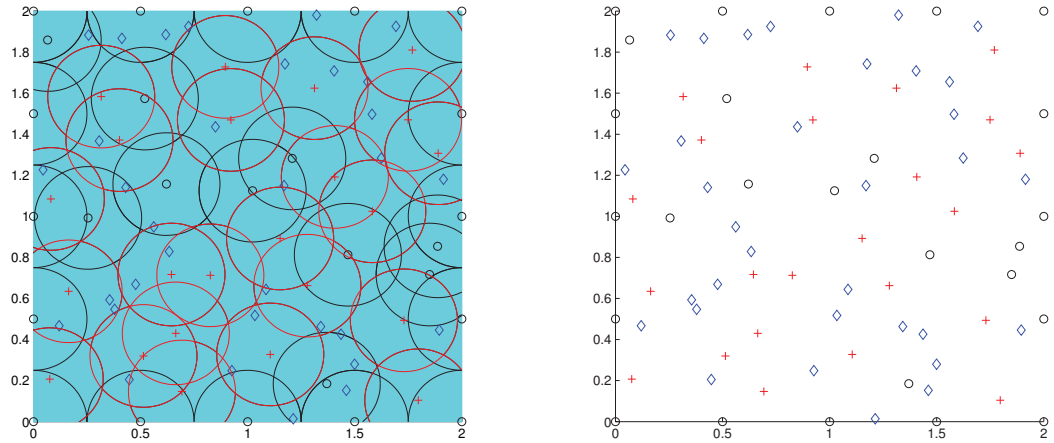


Figure 7.7: The coverage reduction algorithm run on the determinantal method example.

the whole disaster recovery algorithm to the most known coverage recovery algorithm: the greedy algorithm for the set cover problem.

7.6.1 Complexity

The greedy algorithm method lays a square grid of parameter $\sqrt{2}r$ for the Čech complex of potential new vertices. Then the first added vertex is the furthest from all existing vertices. The algorithm goes on adding the furthest potential vertex of the grid from all

vertices (existing+added). It stops when the furthest vertex is in the coverage ball of an existing or added vertex. One can note that our algorithm with the grid method gives the exact same result as the greedy algorithm, number of added vertices and their positions being exactly the same. Both algorithm lays the same square grid of potential new vertices. Then the greedy algorithm picks vertices one by one until perfect coverage is reached. While our algorithm consider all the vertices and removes the non-needed vertices. Then for the $(i+1)$-th vertex addition, the greedy algorithm computes the distances from all $N_i + i$ existing vertices to all $(\lfloor \frac{a}{\sqrt{2}r} \rfloor + 1)^2 - i$ potential vertices. Therefore the complexity of the greedy algorithm is in $O((N_i + N_a)(\lfloor \frac{a}{\sqrt{2}r} \rfloor + 1)^2))$.

For the complexity of our algorithm, we consider first the complexity of building the simplicial complex associated with the network which is in $O((N_i + N_a)^C)$, where C is the clique number. This complexity seems really high since C can only be upper bounded by $N_i + N_a$ in the general case but it is the only way to compute the coverage when vertices position are not defined along a grid. Then the complexity of the coverage reduction algorithm is of the order of $O((1 + (\frac{r}{a})^2)^{N_i+N_a})$ (see Chapter 3). So the greedy algorithm appears less complex than ours in the general case. However when r is small before a or when the dimension is greater than 2, then the power factor becomes $d > 2$ and C is a finite integer, so the trend is reversed.

7.6.2 Mean final number of added vertices

We compare here the mean number of added vertices between our homology algorithm with the determinantal addition method, and the greedy algorithm.

Results presented in Table 7.3 are simulated in the same conditions as in Section 7.4 and given in mean over 1000 simulations. They concern the final number of added vertices: the number of added vertices kept after the reduction algorithm, or added with the greedy algorithm.

Percentage of area initially covered	20%	40%	60%	80%
Greedy algorithm	3.69	3.30	2.84	1.83
Homology algorithm	4.42	3.87	2.97	1.78

Table 7.3: Mean final number of added vertices $\mathbf{E}[N_f]$

The numbers of vertices added in the final state both with our recovery algorithm and the greedy algorithm are roughly the same. They both tend to the minimum number of vertices required to cover the uncovered area depending on the initial configuration. Nonetheless, we can see that our algorithm performs a little bit worse than the greedy algorithm in the less covered area scenarii because the vertices are not optimally positioned and it can be seen when just a small percentage of area is covered, and whole parts of the grid from the greedy algorithm are used, instead of isolated vertices. In compensation, our homology algorithm performs better in more covered scenarii.

7.6.3 Smoothed robustness

To show the advantages of our disaster recovery algorithm we choose to evaluate the robustness of the algorithm when the added vertices positions are slightly moved, i.e. when the nodes positioning does not strictly follow the theoretical positioning. In order

to do this, we apply a Gaussian perturbation to each the added vertices position. The covariance matrix of the perturbation is given by $\Sigma = \sigma^2 \mathrm{Id}$ with $\sigma^2 = 0.01$, which means that the standard deviation for each vertex is of $\sigma = 0.1$. Other simulations parameters are unchanged, results in Table 7.4 and 7.5 are given in mean over 1000 simulations. First, we compute the average number of holes β_1 created by the Gaussian perturbation in Table 7.4. Then in Table 7.5, we counted the percentage of simulations in which the number of holes is still zero after the Gaussian perturbation on the new vertices positions.

Percentage of area initially covered	20%	40%	60%	80%
Greedy algorithm	0.68	0.65	0.45	0.35
Homology algorithm	0.62	0.53	0.37	0.26

Table 7.4: Mean number of holes $\mathbf{E}[\beta_1]$ after the Gaussian perturbation

Percentage of area initially covered	20%	40%	60%	80%
Greedy algorithm	40.8%	47.7%	61.0%	69.3%
Homology algorithm	50.9%	58.1%	67.9%	75.3%

Table 7.5: Probability that there is no hole $\mathbf{P}(\beta_1 = 0)$ after the Gaussian perturbation

We can see that the perturbation on the number of holes decreases with the percentage of area initially covered, since the initial vertices are not perturbed. Our homology algorithm clearly performs better, even in the least covered scenarii, there are less than 50% of simulations that create coverage holes, which is not the case for the greedy algorithm. The greedy algorithm also always create more coverage holes in mean than our disaster recovery algorithm for the same vertices positions perturbation. Therefore our algorithm seems more fitted to the disaster recovery case when a recovery network is deployed in emergency both indoor, via Femtocells, and outdoor, via a trailer fleet, where exact GPS locations are not always available, and exact theoretical positioning is not always followed.

Chapter 8

Conclusion

This chapter aims at summarizing the major contributions of this thesis in these different fields: wireless networks, simplicial homology and probability theory. Then, in the second section we discuss the possible further directions of this work.

8.1 Contributions

This thesis focuses on the application of simplicial homology to wireless sensor networks through the description of a reduction algorithm for simplicial complexes. Therefore this work has contributed to three different domains: wireless networks, be it cellular or sensor networks, via the provided applications; simplicial homology via our main algorithm; and probability via the clique number behavior computations. The main contributions are detailed below:

- **Simplicial complex representation, reduction algorithm applications and determinantal point process simulation for wireless networks.**

  The main contribution of this thesis is the development of a reduction algorithm for simplicial complexes. This algorithm is relevant in regard to the simplicial complex representation of wireless sensor networks. Indeed, considering a wireless sensor network with superficial nodes, our reduction algorithm provides which sensors can be switched off without loss of connectivity or coverage inducing potentially considerable power savings. With our algorithm, the sensors can be switched off in an optimized order; moreover, we proved that the reached solution is an optimum, found boundaries for the number of switched-off sensors, and properties for the resulting optimal sensor network.

  Furthermore, we applied the simplicial complex representation to other wireless networks, namely cellular networks. We introduced a frequency auto-planning algorithm for the self-configuration of SON in future cellular networks. We incorporated traffic considerations to propose an energy conservation algorithm for self-optimization in LTE. Finally, we presented a disaster recovery algorithm more fitted to real-life situations than any known solutions, for the healing of a wireless network.

  Finally, We introduced the simulation of a determinantal point process, the Ginibre point process, and its applications to wireless networks. Indeed the Ginibre point process exhibits repulsion between vertices, and thus has the inherent ability to locate areas with less vertices density, like coverage hole.

- **Reduction algorithm for simplicial complexes.**
  Our reduction algorithm can be applied to any type of simplicial complex. In simplicial homology, reduction algorithm often aims at reducing the number of simplices of a simplicial complex, in order to compute its homology most of the time. Our algorithm uses the opposite approach: we first compute the homology and then use this information as a tool to reduce the number of simplices without modifying the homology in an optimized order. This is the first reduction algorithm for simplicial complexes using this approach we know of.

- **Clique number in random geometric graph.**
  While computing the complexity of our algorithm, we investigated the asymptotical behavior of the clique number of a random geometric graph on the torus in any dimension. This is a well known graph characteristic and a problem often treated. We applied results from simplicial homology concerning the number of simplices in a random geometric complex to graph theory and the random geometric graph. We were able to find the almost sure asymptotical behavior of the clique number of a random geometric graph in every three percolation regimes. Moreover we derived results for related graph characteristics such as the maximum vertex degree, the chromatic number and the independence number.

8.2 Future research directions

As of future research directions, since the simplicial complex representation for wireless networks is pretty new, we think there is still much to do. We can in particular develop two new directions:

- **Simplicial complex representation for collaboration networks.**
  Some pioneer works have begun to propose the use of simplicial complex representation for collaboration networks [63] or social structures [52]. It is possible to generalize the notion of vertex degree in graphs (number of incident edges to a vertex), to any simplex (number of incident larger simplices to a simplex). We can also consider our definition of a k-simplex degree in our reduction algorithm: the size of the largest simplex it is a face of. These generalized degrees would have interpretation in collaboration networks: number of groups of a given size an individual or a group is part of, size of the largest group an individual or a group is part of. Moreover, we think that it is possible to estimate these degrees via random walks on simplicial complexes. Random walks on simplicial complexes are a generalization of random walks on graph, where it is possible to walk from k-simplex to k-simplex through $k+n$-simplices connections [71].

- **Persistent homology applications.**
  Persistent homology has been applied to wireless sensor networks [25], but only for homology computation reasons. We think that there is inherent interest in the persistent homology representation. Indeed, let us consider the following problem: A grid of sensors is deployed in a cell of a cellular network in order to measure reception. Then we can build a coverage simplicial complex as follows: if k sensors within communication range receive the signal with a SINR above a given threshold, then a $(k-1)$-simplex is drawn. Increasing the threshold would create a family of

persistent complexes. Therefore, using persistent homology, one would be able to obtain a coverage map for any acceptable threshold of SINR.

Appendix A

Simplicial homology of random configurations [26]

L. Decreusefond, E. Ferraz, H. Randriambololona, A. Vergne

Given a Poisson process on a d-dimensional torus, its random geometric simplicial complex is the complex whose vertices are the points of the Poisson process and simplices are given by the Čech complex associated to the coverage of each point. By means of Malliavin calculus, we compute explicitly the three first order moments of the number of k-simplices, and provide a way to compute higher order moments. Then, we derive the mean and the variance of the Euler characteristic. We use a concentration inequality for Poisson processes to find bounds for the tail distribution of the Betti number of first order and the Euler characteristic in such simplicial complexes.

A.1 Poisson point process and Malliavin calculus

The space of configurations on $X = [0, a)^d$, is the set of locally finite simple point measures (see [20, 79] for details):

$$\Omega^X = \left\{ \omega = \sum_{k=1}^{n} \delta(x_k) \ : \ (x_k)_{k=1}^{k=n} \subset X, \ n \in \mathbb{N} \cup \{\infty\} \right\},$$

where $\delta(x)$ denotes the Dirac measure at $x \in X$. It is often convenient to identify an element ω of Ω^X with the set corresponding to its support, i.e. $\sum_{k=1}^{n} \delta(x_k)$ is identified with the unordered set $\{x_1, \ldots, x_n\}$. For $A \in \mathcal{B}(X)$, we have $\delta(x)(A) = \mathbf{1}_A(x)$, so that

$$\omega(A) = \sum_{x \in \omega} \mathbf{1}_A(x),$$

counts the number of atoms in A. Simple measure means that $\omega(\{x\}) \leq 1$ for any $x \in X$. Locally finite means that $\omega(K) < \infty$ for any compact K of X. The configuration space Ω^X is endowed with the vague topology and its associated σ-algebra denoted by $\mathcal{F}^X$. To characterize the randomness of the system, we consider that the set of points is represented by a Poisson point process ω with intensity measure $\mathrm{d}\Lambda(x) = \lambda \, \mathrm{d}x$ in X. The parameter λ is called the intensity of the Poisson process. Since ω is a Poisson point process of intensity measure Λ:

i) For any compact A, $\omega(A)$ is a random variable of parameter $\Lambda(A)$:

$$\mathbf{P}(\omega(A) = k) = e^{-\Lambda(A)} \frac{\Lambda(A)^k}{k!}.$$

ii) For any disjoint sets $A, A' \in \mathcal{B}(X)$, the random variables $\omega(A)$ and $\omega(A')$ are independent.

Along this paper, we refer $\mathbf{E}_\Lambda[F]$ as the mean of some function F depending on ω given that the intensity measure of this process is Λ. The notations $\mathrm{Var}_\Lambda[F]$ and $\mathrm{Cov}_\Lambda[F, G]$ are defined accordingly. As said above, a configuration ω can be viewed as a measure on X. It also induces a measure on any X^n, called the factorial measure associated to ω of order n, defined by

$$\omega^{(n)}(C) = \sum_{\substack{(x_1,\cdots,x_n)\in\omega \\ x_i\neq x_j}} \mathbf{1}_C(x_1, \cdots, x_n),$$

for any $C \in X^n$, with the convention that $\omega^{(n)}$ is the null measure if ω has less than n points. Let $f \in L^1(\Lambda^{\otimes n})$ and let F be a random variable given by

$$F(\omega) = \sum_{\substack{x_i\in\omega \\ x_i\neq x_j}} f(x_1, \ldots, x_n) = \int f(x_1, \ldots, x_n)\, \mathrm{d}\omega^{(n)}(x_1, \cdots, x_n).$$

The Campbell-Mecke formula for Poisson point processes states that

$$\mathbf{E}_\Lambda[F] = \int_{X^n} f(x_1, \ldots, x_n)\, \mathrm{d}\Lambda(x_1) \ldots\, \mathrm{d}\Lambda(x_n).$$

In view of this result, it is natural to introduce the compensated factorial measures defined by :

$$d\omega_\Lambda^{(1)}(x) = d\omega(x) - \mathrm{d}\Lambda(x)$$

and for $n \geq 2$, for any $f \in L^1(\Lambda^{\otimes n})$,

$$\int f(x_1, \cdots, x_n) d\omega_\Lambda^{(n)}(x_1, \cdots, x_n)$$
$$= \int \left(\int f(x_1, \cdots, x_n)(d(\omega - \sum_{j=1}^{n-1} \delta(x_j))(x_n) - \mathrm{d}\Lambda(x_n)) \right) d\omega_\Lambda^{(n-1)}(x_1, \cdots, x_{n-1}).$$

A real-valued function $f : X^n \to \mathbb{R}$ is called symmetric if

$$f(x_{\sigma(1)}, \ldots, x_{\sigma(n)}) = f(x_1, \ldots, x_n)$$

for all permutations σ of $\mathfrak{S}_n$. Then the space of square integrable symmetric functions of n variables is denoted by $L^2(X, \Lambda)^{\circ n}$. For $f \in L^2(X, \Lambda)^{\circ n}$, the multiple Poisson stochastic integral $I_n(f_n)$ is then defined as

$$I_n(f_n)(\omega) = \int f_n(x_1, \ldots, x_n)\, \mathrm{d}\omega_\Lambda^{(n)}(x_1, \cdots, x_n).$$

It is known that $I_n(f_n) \in L^2(\Omega^X, \mathbf{P})$. Moreover, if $f_n \in L^2(X,\, \Lambda)^{\circ n}$ and $g_m \in L^2(X,\, \Lambda)^{\circ m}$, the isometry formula

$$\mathbf{E}_\Lambda\left[I_n(f_n)I_m(g_m)\right] = n!\,\mathbf{1}_m(n)\ \langle f_n, g_m\rangle_{L^2(X,\Lambda)^{\circ n}} \tag{A.1}$$

holds true. Furthermore, we have:

Theorem 23. *Every random variable $F \in L^2(\Omega^X,\, \mathbf{P})$ admits a unique Wiener-Poisson decomposition of the type*

$$F = \mathbf{E}_\Lambda\left[F\right] + \sum_{n=1}^{\infty} I_n(f_n),$$

where the series converges in $L^2(\Omega^X, \mathbf{P})$ and, for each $n \geq 1$, the kernel f_n is an element of $L^2(X,\Lambda)^{\circ n}$. Moreover, by definition $\mathrm{Var}_\Lambda\left[F\right] = \|F - \mathbf{E}_\Lambda\left[F\right]\|^2_{L^2(\Omega^X,\mathbf{P})}$ then we have the isometry

$$\mathrm{Var}_\Lambda\left[F\right] = \sum_{n=1}^{\infty} n!\ \|f_n\|^2_{L^2(X,\Lambda)^{\circ n}}. \tag{A.2}$$

For $f_n \in L^2(X,\Lambda)^{\circ n}$ and $g_m \in L^2(X,\Lambda)^{\circ m}$, we define $f_n \otimes^l_k g_m$, $0 \leq l \leq k$, to be the function:

$$(y_{l+1}, \ldots, y_n, x_{k+1}, \ldots, x_m) \longmapsto \int_{X^l} f_n(y_1, \ldots, y_n) g_m(y_1, \ldots, y_k, x_{k+1}, \ldots, x_m)\ \mathrm{d}\Lambda(y_1)\ldots\ \mathrm{d}\Lambda(y_l). \tag{A.3}$$

We denote by $f_n \circ^l_k g_m$ the symmetrization in $n+m-k-l$ variables of $f_n \otimes^l_k g_m$, $0 \leq l \leq k$. This leads us to the next proposition (see [79] for a proof):

Proposition 1. *For $f_n \in L^2(X,\Lambda)^{\circ n}$ and $g_m \in L^2(X,\Lambda)^{\circ m}$, we have*

$$I_n(f_n)I_m(g_m) = \sum_{s=0}^{2(n\wedge m)} I_{n+m-s}(h_{n,m,s}),$$

where

$$h_{n,m,s} = \sum_{s\leq 2i\leq 2(s\wedge n\wedge m)} i!\binom{n}{i}\binom{m}{i}\binom{i}{s-i} f_n \circ^{s-i}_i g_m$$

belongs to $L^2(X,\Lambda)^{\circ n+m-s}$, $0 \leq s \leq 2(m\wedge n)$.

In what follows, given $f \in L^2(X,\Lambda)^{\circ q}$ $(q \geq 2)$ and $t \in X$, we denote by $f(*,x)$ the function on X^{q-1} given by $(x_1, \ldots, x_{q-1}) \longmapsto f(x_1, \ldots, x_{q-1}, x)$.

Definition 42. *Let $\mathrm{Dom}\, D$ be the the set of random variables $F \in L^2(\Omega^X,\, \mathbf{P})$ admitting a chaotic decomposition such that*

$$\sum_{n=1}^{\infty} qq!\|f_n\|^2_{L^2(X,\Lambda)^{\circ n}} < \infty.$$

Let D be defined by

$$D\ :\ \mathrm{Dom}\, D \to L^2(\Omega^X \times X, \mathbf{P}\otimes\Lambda)$$
$$F = \mathbf{E}_\Lambda\left[F\right] + \sum_{n\geq 1} I_n(f_n) \longmapsto D_x F = \sum_{n\geq 1} nI_{n-1}(f_n(*,x)).$$

It is known, cf. [48], that we also have

$$D_x F(\omega) = F(\omega \cup \{x\}) - F(\omega), \ \mathbf{P} \otimes \Lambda - a.e.$$

Definition 43. *The Ornstein-Uhlenbeck operator L is given by*

$$LF = -\sum_{n=1}^{\infty} n I_n(f_n),$$

whenever $F \in \operatorname{Dom} L$, given by those $F \in L^2(\Omega^X, \mathbf{P})$ such that their chaos expansion verifies

$$\sum_{n=1}^{\infty} q^2 q! \|f_n\|^2_{L^2(X,\Lambda)^{\circ n}} < \infty.$$

Note that $\mathbf{E}_\Lambda[LF] = 0$, by definition and (A.1).

Definition 44. *For $F \in L^2(\Omega^X, \mathbf{P})$ such that $\mathbf{E}_\Lambda[F] = 0$, we may define L^{-1} by*

$$L^{-1} F = -\sum_{n=1}^{\infty} \frac{1}{n} I_n(f_n).$$

Combining Stein's method and Malliavin calculus yields the following theorem, see [72]:

Theorem 24. *Let $F \in \operatorname{Dom} D$ be such that $\mathbf{E}_\Lambda[F] = 0$ and $Var(F) = 1$. Then,*

$$d_W(F, \mathcal{N}(0,1)) \leq \mathbf{E}_\Lambda \left[\left| 1 + \int_X [D_x F \times D_x L^{-1} F] \, d\Lambda(x) \right| \right] + \int_X \mathbf{E}_\Lambda \left[|D_x F|^2 \left| D_x L^{-1} F \right| \right] \, d\Lambda(x).$$

Another result from the Malliavin calculus used in this work is the following one, quoted from [79]:

Theorem 25. *Let $F \in \operatorname{Dom} D$ be such that $DF \leq K$, a.s., for some $K \geq 0$ and we denote*

$$\|DF\|_{L^\infty(L^2(X,\Lambda),\mathbf{P})} := \sup_\omega \int_X |D_x F(\omega)|^2 \, d\Lambda(x) < \infty.$$

Then

$$\mathbf{P}(F - \mathbf{E}_\Lambda[F] \geq x) \leq \exp\left(-\frac{x}{2K} \log\left(1 + \frac{xK}{\|DF\|_{L^\infty(L^2(X,\Lambda),\mathbf{P})}} \right) \right). \tag{A.4}$$

A.2 First order moments

Let ω denote a generic realization of a Poisson point process on the torus $\mathbb{T}^d_a$ and $\mathcal{C}_\epsilon(\omega)$ the associated Čech complex with $\epsilon < a/4$. A Poisson process in $\mathbb{R}^d$ of intensity λ dilated by a factor α is a Poisson process of intensity $\lambda \alpha^{-d}$. Hence, statistically, the homological properties of a Poisson process of intensity λ, inside a torus of length a with ball sizes ϵ are the same as that of a Poisson process of intensity $\lambda \alpha^{-d}$, inside a torus of length αa with ball sizes $\alpha\epsilon$. Thus there are only two degrees of freedom among λ, a, and ϵ. For instance, we can set $a = 1$ and the general results are obtained by a multiplication of magnitude

a^d. Strictly speaking, Betti numbers, Euler characteristic and number of k-simplices are functions of $\mathcal{C}_\epsilon(\omega)$ but we will skip this dependence for the sake of notations. We also define N_k as the number of $(k-1)$-simplices.

In this section, we evaluate the mean of the number of $(k-1)$-simplices $\mathbf{E}_\Lambda[N_k]$ and the mean of the Euler characteristic $\mathbf{E}_\Lambda[\chi]$. We introduce some notations. Let

$$\Delta_k^{(d)} = \{(v_1, \ldots, v_k) \in ([0,\, a)^d)^k,\ v_i \neq v_j, \forall i \neq j\}.$$

For any integer k, we define $\varphi_k^{(d)}$ as:

$$\begin{aligned}\varphi_k^{(d)} &: ([0,\, a)^d)^k \longrightarrow \{0,1\}\\ (v_1,\, \cdots,\, v_k) &\longmapsto \begin{cases}\prod_{1\leq i<j\leq k} \mathbf{1}_{[0,\,2\epsilon)}(\rho_d(v_i,\, v_j)) & \text{if } (v_1, \ldots, v_k) \in \Delta_k^{(d)},\\ 0 & \text{otherwise.}\end{cases}\end{aligned}$$

In words, this means that $\varphi_k^{(d)}(v_1, \cdots, v_k) = 1$ if $[v_1, \cdots, v_k]$ is a $(k-1)$-simplex and 0 otherwise.

Theorem 26. *The mean number of $(k-1)$-simplices N_k is given by*

$$\mathbf{E}_\Lambda[N_k] = \frac{\lambda a^d (\lambda (2\epsilon)^d)^{k-1} k^d}{k!}.$$

Proof The number of $(k-1)$-simplices can be counted by the expression:

$$N_k(\omega) = \frac{1}{k!} \int \varphi_k^{(d)}(v_1, \cdots, v_k)\ \mathrm{d}\omega^{(k)}(v_1, \cdots, v_k).$$

According to the Campbell-Mecke formula and since the max-distance can be tensorized, we have:

$$\begin{aligned}\mathbf{E}_\Lambda[N_k] &= \frac{\lambda^k}{k!} \int_{X^k} \varphi_k^{(d)}(v_1, \cdots, v_k)\ \mathrm{d}v_1 \ldots\ \mathrm{d}v_k\\ &= \frac{\lambda^k}{k!} \left(\int_{[0,a)^k} \varphi_k^{(1)}(x_i,\, x_j)\ \mathrm{d}x_1 \ldots\ \mathrm{d}x_k \right)^d.\end{aligned}$$

A moment of thought reveals that for any $(x_1, \cdots, x_k) \in \Delta_k^{(1)}$, since $\epsilon < a/4 < a/2$, there exists a unique index i such that for all $j \in \{1, \cdots, k\} \backslash \{i\}$, one and only one of the two following conditions holds:

$$x_i < x_j < x_i + 2\epsilon \text{ or } x_i < x_j + a < x_i + 2\epsilon.$$

Let $\zeta(x_1, \cdots, x_k)$ denote this index i. Hence, by invariance by translation of the Lebesgue measure,

$$\begin{aligned}&\int_{\zeta^{-1}(1)} \varphi_k^{(1)}(x_i,\, x_j)\ \mathrm{d}x_1 \ldots\ \mathrm{d}x_k\\ &\qquad = (k-1)! \int_0^a \mathrm{d}x_1 \int_{[x_1,\, x_1+2\epsilon)^{k-1}} \prod_{j=2}^{k-1} \mathbf{1}_{x_j < x_{j+1}}\ \mathrm{d}x_2 \ldots\ \mathrm{d}x_k = a(2\epsilon)^{k-1}.\end{aligned}$$

The very same identity holds for any integral on the set $\zeta^{-1}(i)$ for any $i \in \{1, \cdots, k\}$ hence

$$\int_{[0,a)^k} \varphi_k^{(1)}(x_i, x_j)\, \mathrm{d}x_1 \dots\, \mathrm{d}x_k = ka(2\epsilon)^{k-1}.$$

The proof is thus complete.

By depoissonization, we can estimate the mean number of k-simplices for a Binomial process: a process with n points uniformly distributed over the torus.

Corollary 27. *The mean number $(k-1)$-simplices N_k given $N_1 = n$ is*

$$\mathbf{E}_\Lambda\left[N_k \,|\, N_1 = n\right] = \binom{n}{k} k^d \left(\frac{2\epsilon}{a}\right)^{d(k-1)}.$$

Proof According to Theorem 26, we have:

$$\frac{\lambda a^d (\lambda(2\epsilon)^d)^{k-1} k^d}{k!} = \sum_{n=0}^{\infty} \mathbf{E}_\Lambda\left[N_k \,|\, N_1 = n\right] e^{-\lambda a^d} \frac{(\lambda a^d)^n}{n!}.$$

The principle of depoissonization is then to invert the transform Θ defined by:

$$\begin{aligned} \Theta : \mathbb{R}^{\mathbb{N}} &\longrightarrow \mathbb{R}[\lambda] \\ (\alpha_n,\, n \geq 0) &\longmapsto \sum_{n\geq 0} \alpha_n e^{-\lambda} \frac{\lambda^n}{n!}. \end{aligned}$$

We have that $(\lambda a^d)^k = \sum_{n\geq k} \frac{n!}{(n-k)!} \frac{(\lambda a^d)^n}{n!} e^{-\lambda a^d}$. The result follows.

Remark 11. *Considering the maximum norm simplifies the calculations. However, even for the Euclidean norm, it is still possible to find a closed-form expression for $\mathbf{E}_\Lambda[N_2]$ and $\mathbf{E}_\Lambda[N_3]$ when we consider the Vietoris-Rips complex in $\mathbb{T}_a^2$. We are limited to small orders because no formula seems to be known for the area of the intersection of k balls in general position. For $k = 2$ and 3, the expectations are given by the following formulas:*

$$\begin{aligned} \mathbf{E}_\Lambda[N_2] &= \frac{\pi(a\lambda\epsilon)^2}{2}, \\ \mathbf{E}_\Lambda[N_3] &= \pi\left(\pi - \frac{3\sqrt{3}}{4}\right)\frac{\lambda^3 a^2 \epsilon^4}{6}. \end{aligned}$$

Consider now the Bell's polynomial $B_d(x)$, defined as (see [13]):

$$B_n(x) = \sum_{k=0}^{n} \left\{ {n \atop k} \right\} x^k,$$

where n is a positive integer and $\left\{ {n \atop k} \right\}$ is the Stirling number of the second kind. An equivalent definition of B_n can be:

$$B_n(x) = e^{-x} \sum_{k=0}^{\infty} \frac{x^k k^d}{k!}.$$

These polynomials appear rather surprisingly in the computations of the mean value of the Euler characteristic.

Theorem 28. *The mean of the Euler characteristic of the simplicial complex $\mathcal{C}_\epsilon(\omega)$ is given by*

$$\mathbf{E}_\Lambda[\chi] = -\left(\frac{a}{2\epsilon}\right)^d e^{-\lambda(2\epsilon)^d} B_d(-\lambda(2\epsilon)^d).$$

Proof Since

$$N_k \leq \frac{1}{k!}\prod_{j=0}^{k-1}(N_1 - j) \leq \frac{N_1^k}{k!}, \text{ then } \sum_{k=1}^{\infty} N_k \leq \sum_{k=1}^{\infty}\frac{N_1^k}{k!} \leq e^{N_1}.$$

As $\mathbf{E}_\Lambda\left[e^{N_1}\right] < \infty$, we have $\mathbf{E}_\Lambda\left[-\sum_{k=1}^{\infty}(-1)^k N_k\right] = -\sum_{k=1}^{\infty}(-1)^k \mathbf{E}_\Lambda[N_k]$ and

$$\begin{aligned}
\mathbf{E}_\Lambda[\chi] &= -\sum_{k=1}^{\infty}(-1)^k \frac{\lambda^k (ak(2\epsilon)^{k-1})^d}{k!} \\
&= \frac{a^d e^{-\lambda(2\epsilon)^d}}{-(2\epsilon)^d} e^{\lambda(2\epsilon)^d} \sum_{k=0}^{\infty}\frac{(-\lambda(2\epsilon)^d)^k k^d}{k!} \\
&= -\left(\frac{a}{2\epsilon}\right)^d e^{-\lambda(2\epsilon)^d} B_d(-\lambda(2\epsilon)^d).
\end{aligned}$$

The proof is thus complete.

If we take $d = 1$, 2 and 3, we obtain:

$$\begin{aligned}
&\mathbf{E}_\Lambda[\chi] = a\lambda e^{-\lambda 2\epsilon}, \text{ for } d = 1; \\
&\mathbf{E}_\Lambda[\chi] = a^2\lambda e^{-\lambda(2\epsilon)^2}\left(1 - \lambda(2\epsilon)^2\right), \text{ for } d = 2; \\
&\mathbf{E}_\Lambda[\chi] = a^3\lambda e^{-\lambda(2\epsilon)^3}\left(1 - 3\lambda(2\epsilon)^3 + (\lambda(2\epsilon)^3)^2\right), \text{ for } d = 3.
\end{aligned}$$

The next corollary is an immediate consequence of Corollary 27, obtained again by depoissonization.

Corollary 29. *The expectation of χ for a binomial point process with n points is given by:*

$$\mathbf{E}_\Lambda[\chi \,|\, N_1 = n] = \sum_{k=0}^{n}(-1)^k \binom{n}{k} k^d \left(\frac{2\epsilon}{a}\right)^{d(k-1)}.$$

So far, we have not say a word about Betti numbers. It turns out that the preceding computations lead to a bound of the tail of β_0, the number of connected components.

Theorem 30. *For $y > \lambda a^d$, we have*

$$\mathbf{P}_\Lambda(\beta_0 \geq y) \leq \exp\left(-\frac{y - \lambda a^d}{2}\log\left(1 + \frac{y - \lambda a^d}{(2^d - 1)^2\lambda}\right)\right).$$

Proof β_0 is the number of connected components. Since there are more points than connected components, $\mathbf{E}_\Lambda[\beta_0] \leq \mathbf{E}_\Lambda[N_1] = \lambda a^d$. According to the definition of D, $\sup_{x\in X} D_t\beta_0$ is the maximum variation of β_0 induced by the addition of an arbitrary point. If the point x is at a distance smaller than ϵ from ω, then $D_x\beta_0 \leq 0$, otherwise,

$D_x\beta_0 = 1$, so $D_x\beta_0 \leq 1$ for any $x \in X$. Besides, this added point can join at most two connected components in each dimension, so in d dimensions it can join at most 2^d connected component, which means that $D\beta_0$ ranges from $-(2^d-1)$ to 1, and then

$$\|D\beta_0\|_{L^\infty(L^2(X,\Lambda),\mathbf{P})} \leq \sup_\omega \int_X |D_x\beta_0|^2 \, \mathrm{d}\Lambda(x) \leq \lambda a^d (2^d-1)^2.$$

Since the function f defined by

$$f(x,y) = \exp\left(-\frac{k_1-x}{2k_2}\log\left(1+\frac{(k_1-x)k_2}{k_3 y}\right)\right).$$

is strictly increasing with respect to x and y for $k_1 > x$, it follows from Theorem 25 that:

$$\mathbf{P}_\Lambda(\beta_0 \geq y) \leq \exp\left(-\frac{y-\lambda a^d}{2}\log\left(1+\frac{y-\lambda a^d}{(2^d-1)^2\lambda a^d}\right)\right),$$

for $y > \lambda a^d \geq \mathbf{E}_\Lambda[\beta_0]$.

A.3 Second order moments

We now deal with the computations of the second order moments. The proofs rely on the chaos decomposition of the number of simplices (see Lemma 11) and the multiplication formula for iterated integrals (see Proposition 1). The computations are rather technical and not detailed here. We make the following convention: For any integer k,

$$\int_{X^0} \varphi_k^{(d)}(v_1, \cdots, v_k) \, \mathrm{d}v_1 \ldots \, \mathrm{d}v_0 = \varphi_k^{(d)}(v_1, \cdots, v_k).$$

Lemma 11. *We can rewrite N_k as*

$$N_k = \frac{1}{k!}\sum_{i=0}^{k}\binom{k}{i}\lambda^{k-i} \, I_i\left(\int_{X^{k-i}} \varphi_k^{(d)}(v_1, \cdots, v_k) \, dv_1 \ldots \, dv_{k-i}\right).$$

Proof For $k = 1$, the result is immediate with the convention made above. Once we have seen that

$$\begin{aligned}
&\int \varphi_k^{(d)}(v_1, \ldots, v_k) \, \mathrm{d}\omega^{(k)}(v_1, \cdots, v_k) \\
&= \int \left(\int \varphi_k^{(d)}(v_1, \ldots, v_k)(d(\omega - \sum_{j=1}^{k-1}\delta(v_j))(v_k) - \, \mathrm{d}\Lambda(v_k)) \right) \mathrm{d}\omega^{(k-1)}(v_1, \cdots, v_{k-1}) \\
&\qquad + \int \left(\int_X \varphi_k^{(d)}(v_1, \ldots, v_k) \, \mathrm{d}\Lambda(v_k) \right) \mathrm{d}\omega^{(k-1)}(v_1, \cdots, v_{k-1}),
\end{aligned}$$

the result follows by induction.

Theorem 31. *The covariance between the number of $(k-1)$-simplices N_k, and the number of $(l-1)$-simplices, N_l for $l \leq k$ is given by*

$$\mathrm{Cov}_\Lambda[N_k, N_l] = \sum_{i=1}^{l} \frac{\lambda a^d (\lambda(2\epsilon)^d)^{k+l-i-1}}{i!(k-i)!(l-i)!}\left(k+l-i+2\frac{(k-i)(l-i)}{i+1}\right)^d.$$

Remark 12. *As for the first moment it is still possible to find, considering the Euclidean norm, a closed-form expression for* $\mathrm{Var}_\Lambda[N_k]$. *We did not find a general expression for any dimension. However, when we consider the Rips-Vietoris complex in* $\mathbb{T}^2_a$, *the variance of the number of 1-simplices and 2-simplices are given by:*

$$\mathrm{Var}_\Lambda[N_2] = \left(\frac{a}{2\epsilon}\right)^2 \left(\frac{\pi}{2}(4\lambda\epsilon^2)^2 + \pi^2(4\lambda\epsilon^2)^3\right),$$

and

$$\mathrm{Var}_\Lambda[N_3] = \left(\frac{a}{2\epsilon}\right)^2 \left((4\lambda\epsilon)^3\frac{\pi}{6}\left(\pi - \frac{3\sqrt{3}}{4}\right) + (4\lambda\epsilon^2)^4\pi\left(\frac{\pi^2}{2} - \frac{5}{12} - \frac{\pi\sqrt{3}}{2}\right) + (4\lambda\epsilon^2)^5\frac{\pi^2}{4}\left(\pi - \frac{3\sqrt{3}}{4}\right)^2\right).$$

Since we have an expression for the variance of the number of k-simplices, it is possible to calculate the variance of the Euler characteristic.

Theorem 32. *The variance of the Euler characteristic is:*

$$\mathrm{Var}_\Lambda[\chi] = \lambda a^d \sum_{n=1}^{\infty} c_n^d (\lambda(2\epsilon)^d)^{n-1},$$

where

$$c_n^d = \sum_{j=\lceil (n+1)/2\rceil}^{n} \left[2\sum_{i=n-j+1}^{j} \frac{(-1)^{i+j}}{(n-j)!(n-i)!(i+j-n)!}\left(n + \frac{2(n-i)(n-j)}{1+i+j-n}\right)^d - \frac{1}{(n-j)!^2(2j-n)!}\left(n + \frac{2(n-j)^2}{1+2j-n}\right)^d\right].$$

Theorem 33. *In one dimension, the expression of the variance of the Euler characteristic is:*

$$\mathrm{Var}_\Lambda[\chi] = a\left(\lambda e^{-2\lambda\epsilon} - 4\lambda^2\epsilon e^{-4\lambda\epsilon}\right).$$

Theorem 34. *If* $d = 2$, *we have* $D\chi \leq 2$ *and thus*

$$\mathbf{P}(\chi - \mathbf{E}_\Lambda[\chi] \geq x) \leq \exp\left(-\frac{x}{4}\log\left(1 + \frac{x}{2\lambda a^2}\right)\right).$$

Proof In two dimensions, the Euler characteristic is given by:

$$\chi = \beta_0 - \beta_1 + \beta_2.$$

If we add a vertex on the torus, either the vertex is isolated or not. In the first case, it forms a new connected component increasing β_0 by 1, and the number of holes, i.e. β_1, remains the same. Otherwise, as there is no new connected component, β_0 is the same, but the new vertex can at most fill one hole, increasing β_1 by 1. Therefore, the variation of $\beta_0 - \beta_1$ is at most 1.

Furthermore, when we add a vertex to a simplicial complex, we know from Proposition **??** that $D\beta_2 \leq 1$ hence $D\chi \leq 2$. Then, we use Eq. (A.4) to complete the proof.

A.4 Third order moments

Higher order moments can be computed in a similar way but the computations become trickier as the order increases. We here restrict our computations to the third order moments to illustrate the general procedure. The proof is not given here.

We are interested in the central moment, so we introduce the following notation for the centralized number of $(k-1)$-simplices: $\widetilde{N_k} = N_k - \mathbf{E}_\Lambda[N_k]$.

Theorem 35. *The third central moment of the number of $(k-1)$-simplices is given by:*

$$\mathbf{E}_\Lambda\left[\widetilde{N_k}^3\right] = \sum_{i,j,s,t} \frac{\lambda^{3k-i-j} t!}{(k!)^3} \binom{k}{i}\binom{k}{j}\binom{k}{s}\binom{i}{t}\binom{j}{t}\binom{t}{i+j-s-t} \mathcal{J}_3(k,i,j,s,t),$$

with $s \geq |i-j|$, and $\mathcal{J}_3(k,i,j,s,t)$ is an integral depending on k, i, j, s and t, defined below in the proof.

Appendix B

Efficient simulation of the Ginibre process [27]

L. Decreusefond, I. Flint, A. Vergne

The Ginibre point process is one of the main examples of determinantal point processes on the complex plane. It forms a recurring model in stochastic matrix theory as well as in practical applications. However, this model has mostly been studied from a probabilistic point of view in the fields of stochastic matrices and determinantal point processes, and thus using the Ginibre process to model random phenomena is a topic which is for the most part unexplored. In order to obtain a determinantal point process more suited for simulation, we introduce a modified version of the classical kernel. Then, we compare three different methods to simulate the Ginibre point process and discuss the most efficient one depending on the application at hand.

B.1 Introduction

Determinantal point processes form a class of point processes which exhibit repulsion, and model a wide variety of phenomena. After their introduction by Macchi in [59], they have been studied in depth from a probabilistic point of view in [81,82] wherein we find and overview of their mathematical properties. Other than modeling fermion particles (see the account of the determinantal structure of fermions in [83], and also [82] for other examples), they are known to appear in many branches of stochastic matrix theory (see [82] or the thorough overview of [6] for example) and in the study of the zeros of Gaussian analytic functions (see [46]). The Ginibre point process in particular was first introduced in [38] and arises in many problems regarding determinantal point processes. To be more specific, the eigenvalues of a hermitian matrix with (renormalized) complex Gaussian entries (which is a subclass of the so-called Gaussian Unitary Ensemble) are known to form a Ginibre point process. Moreover, the Ginibre point process is the natural extension of the Dyson point process to the complex plane. As such, and as explained in [38], it models the positions of charges of a two-dimensional Coulomb gas in a harmonic oscillator potential, at a temperature corresponding to $\beta = 2$. It should be noted that the Dyson model is a determinantal point process on $\mathbb{R}$ which is of central importance, as it appears as the bulk-scaling limit of a large class of determinantal point processes, c.f. [15].

Simulation of determinantal point processes is mostly unexplored, and was in fact initiated in [45] wherein the authors give a practical algorithm for the simulation of de-

terminantal point processes. Theoretical discussion of the aforementioned algorithm as well as statistical aspects have also been explored in [56]. In the recent papers [62, 85, 87], different authors have used the Ginibre point process to model phenomena arising in networking. The reason being that this particular model has many advantages with regards to applications. It is indeed invariant with respect to rotations and translations, which gives us a natural compact subset on which to simulate it: the ball centered at the origin. Moreover, the electrostatic repulsion between particles seems to be fitting for many applications. Our aim in this paper is to study the simulation of the Ginibre point process form a practical point of view, and give different methods which will be more or less suited to the application at hand. The main problem that arises in practice is that although the eigenvalues of matrices in the GUE ensemble form a Ginibre point process, these eigenvalues are *not* compactly supported, although after renormalization, they tend to a compactly supported as N tends to infinity (this is known as the circular law in stochastic matrix theory). Moreover, as will be seen here, truncating to a natural compact and letting N tend to infinity is not the most efficient way to proceed, even though this operation preserves the determinantal property of the point process. Therefore, our methods will rely on the modification of the kernel associated with the Ginibre point process. We study in depth the projection of the kernel onto a compact, its truncation to a finite rank, and in the last part a combination of both operations. Each of these operations on the kernel will have different results on the resulting point process, as well as the simulation techniques involved.

We proceed as follows. We start in Section B.2 by a general definition of a point process, as well as determinantal point process. In Section B.3, we present more specifically the Ginibre point process, and prove some probabilistic properties. We discuss the truncation, and the projection of the Ginibre kernel and gives the basic ideas that will yield different simulation techniques.

B.2 Notations and general results

B.2.1 Point processes

Let E be a Polish space, $\mathcal{O}(E)$ the family of all non-empty open subsets of E and $\mathcal{B}$ denotes the corresponding Borel σ-algebra. We also consider λ a Radon measure on $(E, \mathcal{B})$. Let $\mathcal{X}$ be the space of locally finite subsets in E, sometimes called the configuration space:

$$\mathcal{X} = \{\xi \subset E \,:\, |\Lambda \cap \xi| < \infty \text{ for any compact set } \Lambda \subset E\}.$$

In fact, $\mathcal{X}$ consists of all simple positive integer-valued Radon measures (by simple we mean that for all $x \in E$, $\xi(x) \leq 1$). Hence, it is naturally topologized by the vague topology, which is the weakest topology such that for all continuous and compactly supported functions f on E, the mapping

$$\xi \mapsto \langle f, \xi \rangle := \sum_{y \in \xi} f(y)$$

is continuous. We denote by $\mathcal{F}$ the corresponding σ-algebra. We call elements of $\mathcal{X}$ configurations and identify a locally finite configuration ξ with the atomic Radon measure $\sum_{y \in \xi} \varepsilon_y$, where we have written ε_y for the Dirac measure at $y \in E$.

Next, let $\mathcal{X}_0 = \{\xi \in \mathcal{X} : |\xi| < \infty\}$ be the space of all finite configurations on E. $\mathcal{X}_0$ is naturally equipped with the trace σ-algebra $\mathcal{F}_0 = \mathcal{F}|_{\mathcal{X}_0}$. A random point process is defined as a probability measure μ on $(\mathcal{X}, \mathcal{F})$. A random point process μ is characterized by its Laplace transform L_μ, which is defined for any measurable non-negative function f on E as

$$\mathrm{L}_\mu(f) = \int_{\mathcal{X}} e^{-\sum_{x\in\xi} f(x)}\, \mu(\mathrm{d}\xi).$$

For the precise study of point processes, we also introduce the λ-sample measure, as well as subsequent tools. Most of our notations are inspired from the ones in [36].

Definition 1. *The λ-sample measure L on $(\mathcal{X}_0, \mathcal{F}_0)$ is defined by the identity*

$$\int f(\alpha)\, L(d\alpha) = \sum_{n\geq 0} \frac{1}{n!} \int_{E^n} f(\{x_1, \dots, x_n\})\, \lambda(dx_1) \dots \lambda(dx_n),$$

for any measurable nonnegative function f on $\mathcal{X}_0$.

Point processes are often characterized via their correlation function, defined as below.

Definition 2 (Correlation function)**.** *A point process μ is said to have a correlation function $\rho : \mathcal{X}_0 \to \mathbb{R}$ if ρ is measurable and*

$$\int_{\mathcal{X}} \sum_{\alpha\subset\xi,\ \alpha\in\mathcal{X}_0} f(\alpha)\, \mu(d\xi) = \int_{\mathcal{X}_0} f(\alpha)\, \rho(\alpha)\, L(d\alpha),$$

for all measurable nonnegative functions f on $\mathcal{X}_0$. For $\xi = \{x_1, \dots, x_n\}$, we will sometimes write $\rho(\xi) = \rho_n(x_1, \dots, x_n)$ and call ρ_n the n-th correlation function, where here ρ_n is a symmetrical function on E^n.

It can be noted that correlation functions can also be defined by the following property, both characterizations being equivalent in the case of simple point processes.

Proposition B.2.1. *A point process μ is said to have correlation functions $(\rho_n)_{n\in\mathbb{N}}$ if for any $A_1, \dots, A_n$ disjoint bounded Borel subsets of E,*

$$\mathbb{E}[\prod_{i=1}^{n} \xi(A_i)] = \int_{A_1\times\dots\times A_n} \rho_n(x_1, \dots, x_n)\, \lambda(dx_1) \dots \lambda(dx_n).$$

Recall that ρ_1 is the mean density of particles with respect to λ, and

$$\rho_n(x_1, \dots, x_n)\, \lambda(\mathrm{d}x_1) \dots \lambda(\mathrm{d}x_n)$$

is the probability of finding a particle in the vicinity of each x_i, $i = 1, \dots, n$. We also need to define the Janossy density of μ, which is defined as follows:

Definition 3. *For any compact subset $\Lambda \subseteq E$, the Janossy density j_Λ is defined (when it exists) as the density function of μ_Λ with respect to L_Λ.*

In the following, we will write $j^n_\Lambda(x_1, \dots, x_n) = j_\Lambda(\{x_1, \dots, x_n\})$ for the n-th Janossy density, i.e. the associated symmetric function of n variables, for a configuration of size $n \in \mathbb{N}$. The Janossy density $j_\Lambda(x_1, \dots, x_n)$ is in fact the joint density (multiplied by a constant) of the n points given that the point process has exactly n points. Indeed, by definition of the Janossy intensities, the following relation is satisfied, for any measurable $f : \mathcal{X}_0 \to \mathbb{R}$,

$$E[f(\xi)] = \sum_{n\geq 0} \frac{1}{n!} \int_{\Lambda^n} f(\{x_1, \dots, x_n\}) j_\Lambda(\{x_1, \dots, x_n\})\, \lambda(\mathrm{d}x_1) \dots \lambda(\mathrm{d}x_n).$$

B.2.2 Determinantal processes

For details on this part, we refer to [81,82]. For any compact subset $\Lambda \subset E$, we denote by $L^2(\Lambda, \lambda)$ the set of functions square integrable with respect to the restriction of the measure λ to the set Λ. This becomes a Hilbert space when equipped with the usual norm:

$$\|f\|^2_{L^2(\Lambda,\lambda)} = \int_\Lambda |f(x)|^2 \mathrm{d}\lambda(x).$$

For Λ a compact subset of E, P_Λ is the projection from $L^2(E, \lambda)$ onto $L^2(\Lambda, \lambda)$, i.e., $P_\Lambda f = 1_\Lambda$. The operators we deal with are special cases of the general set of continuous maps from $L^2(E, \lambda)$ into itself.

Definition 4. *A map T from $L^2(E, \lambda)$ into itself is said to be an integral operator whenever there exists a measurable function, which we still denote by T, such that*

$$Tf(x) = \int_E T(x, y) f(y) d\lambda(y).$$

The function $T : E \times E \to \mathbb{R}$ is called the kernel of T.

Definition 5. *Let T be a bounded map from $L^2(E, \lambda)$ into itself. The map T is said to be trace-class whenever for a complete orthonormal basis $(h_n, n \geq 1)$ of $L^2(E, \lambda)$,*

$$\|T\|_1 := \sum_{n\geq 1} (|T|h_n, h_n)_{L^2} < \infty,$$

where $|T| := \sqrt{TT^*}$. Then, the trace of T is defined by

$$\mathrm{trace}(T) = \sum_{n\geq 1} (Th_n, h_n)_{L^2}.$$

It is easily shown that the notion of trace does not depend on the choice of the complete orthonormal basis. Note that if T is trace-class then T^n also is trace-class for any $n \geq 2$, since we have that $\|T^n\|_1 \leq \|T\|^{n-1}\|T\|_1$ (see e.g. [32]).

Definition 6. *Let T be a trace class operator. The Fredholm determinant of $(\mathrm{I}+T)$ is defined by:*

$$\mathrm{Det}(\mathrm{I}+T) = \exp\left(\sum_{n=1}^{+\infty} \frac{(-1)^{n-1}}{n} \mathrm{trace}(T^n)\right),$$

where I *stands for the identity operator on $L^2(E, \lambda)$.*

The Fredholm determinant can also be expanded as a function of the usual determinant, as can be observed in the following proposition, which can be obtained easily by expanding the exponential in the previous definition (see [81]):

Proposition B.2.2. *For a trace class integral operator T, we have:*

$$\mathrm{Det}(\mathrm{I}-T) = \sum_{n=0}^{+\infty} \frac{1}{n!} \int_{\Lambda^n} \det\left(T(x_i, x_j)\right)_{1\leq i,j\leq n} d\lambda(x_1) \dots d\lambda(x_n).$$

With the previous definitions in mind, we move onto the precise definition of determinantal point processes. To that effect, we will henceforth use the following set of hypotheses:

Hypothesis 1. *The map T is an Hilbert-Schmidt operator from $L^2(E,\,\lambda)$ into $L^2(E,\,\lambda)$ which satisfies the following conditions:*

i) T is a bounded symmetric integral operator on $L^2(E,\,\lambda)$, with kernel $T(.,.)$.

ii) The spectrum of T is included in $[0,\,1]$.

iii) The map T is locally of trace class, i.e., for all compact subsets $\Lambda \subset E$, the restriction $T_\Lambda := P_\Lambda T P_\Lambda$ of T to $L^2(\Lambda, \lambda)$ is of trace class.

For a compact subset $\Lambda \subset E$, the map $J[\Lambda]$ is defined by:

$$J[\Lambda] = (\mathrm{I} - T_\Lambda)^{-1} T_\Lambda, \tag{B.1}$$

so that T and $J[\Lambda]$ are quasi-inverses in the sense that

$$(\mathrm{I} - T_\Lambda)\,(\mathrm{I} + J[\Lambda]) = \mathrm{I}\,.$$

For any compact Λ, the operator $J[\Lambda]$ is also a trace class operator in $L^2(\Lambda,\,\lambda)$. In the following theorem, we define a general determinantal process with three equivalent characterizations: in terms of their Laplace transforms, Janossy densities or correlation functions. The theorem is also a theorem of existence, a problem which is far from being trivial.

Theorem B.2.1 (See [81])**.** *Assume Hypothesis 1 is satisfied. There exists a unique probability measure $\mu_{T,\lambda}$ on the configuration space $\mathcal{X}$ such that, for any nonnegative bounded measurable function f on E with compact support, we have:*

$$L_{\mu_{T,\lambda}}(f) = \mathrm{Det}\left(\mathrm{I} - T[1 - e^{-f}]\right),$$

where $T[1 - e^{-f}]$ is the bounded operator on $L^2(E, \lambda)$ with kernel :

$$(T[1 - e^{-f}])(x, y) = \sqrt{1 - \exp(-f(x))}\,T(x, y)\,\sqrt{1 - \exp(-f(y))}.$$

This means that for any integer n and any $(x_1, \cdots, x_n) \in E^n$, the correlation functions of $\mu_{T,\lambda}$ are given by:

$$\rho_{n,T}(x_1, \cdots,\, x_n) = \det\left(T\,(x_i,\, x_j)\right)_{1 \le i,j \le n},$$

and for $n = 0$, $\rho_{0,T}(\emptyset) = 1$. For any compact subset $\Lambda \subset E$, the operator $J[\Lambda]$ is an Hilbert-Schmidt, trace class operator, whose spectrum is included in $[0, +\infty[$. For any $n \in \mathbb{N}$, any compact $\Lambda \subset E$, and any $(x_1, \cdots, x_n) \in \Lambda^n$ the n-th Janossy density is given by:

$$j^n_{\Lambda,T}\,(x_1,\, \cdots,\, x_n) = \mathrm{Det}\,(\mathrm{I} - T_\Lambda)\,\det\left(J[\Lambda](x_i,\, x_j)\right)_{1 \le i,j \le n}. \tag{B.2}$$

For $n = 0$, we have $j^0_{\Lambda,T}\,(\emptyset) = \mathrm{Det}\,(I - T_\Lambda)$.

We also need a simple condition on the kernels to ensure proper convergence of the associated determinantal measure. This is provided by Proposition 3.10 in [81]:

Proposition B.2.3. *Let $(T^{(n)})_{n \ge 1}$ be integral operators with nonnegative continuous kernels $T^{(n)}(x, y)$, $x, y \in E$. Assume that $T^{(n)}$ satisfy Hypothesis 1, $n \ge 1$, and that $T^{(n)}$ converges to a kernel T uniformly on each compact as n tends to infinity. Then, the kernel T defines an integral operator T satisfying Hypothesis 1. Moreover, the determinantal measure $\mu_{T^{(n)},\lambda}$ converges weakly to the measure $\mu_{T,\lambda}$ as n tends to infinity.*

In the remainder of this section, we shall consider a general determinantal process of kernel T with respect to a reference measure λ on E. We will assume that T satisfies Hypothesis 1. Consider a compact subset $\Lambda \subset E$. Then, by Mercer's theorem, the projection operator T_Λ can be written as

$$T_\Lambda(x,y) = \sum_{n\geq 0} \lambda_n^\Lambda \varphi_n^\Lambda(x)\overline{\varphi_n^\Lambda(y)}, \tag{B.3}$$

for $x, y \in \mathbb{C}$. Here, $(\varphi_n^\Lambda)_{n\in\mathbb{N}}$ are the eigenvectors of T_Λ and $(\lambda_n)_{n\in\mathbb{N}}$ the associated eigenvalues. Note that since T_Λ is trace-class, we have

$$\sum_{n\geq 0} |\lambda_n^\Lambda| < \infty.$$

In this case, the operator $J[\Lambda]$ defined in (B.1) can be decomposed in the same basis as T_Λ.

$$J[\Lambda](x,y) = \sum_{n\geq 0} \frac{\lambda_n^\Lambda}{1-\lambda_n^\Lambda} \varphi_n^\Lambda(x)\overline{\varphi_n^\Lambda(y)}, \tag{B.4}$$

for $x, y \in \Lambda$.

Let us conclude this section by mentioning the particular case of the determinantal projection process. We define a projection kernel (onto $\{\phi_n,\ 0 \leq n \leq N\} \subset L^2(E,\lambda)$) to be

$$T_p(x,y) = \sum_{n=0}^{N} \varphi_n(x)\overline{\varphi_n(y)}, \quad \forall x, y \in \mathbb{C}$$

where $N \in \mathbb{N}$, and $(\varphi_n)_{n\in\mathbb{N}}$ is an orthonormal family of $L^2(E,\lambda)$. We call the associated determinantal process a determinantal projection process (onto $\{\phi_n,\ 0 \leq n \leq N\} \subset L^2(E,\lambda)$). In this case, it is known that the associated determinantal process has N points almost surely, as was first proved in [82]. These determinantal processes are particularly interesting since they benefit from a specific simulation technique which will be explained in the next section.

B.3 Simulation of the Ginibre point process

Proofs of this section are not detailed here but are available in the full version of this article.

B.3.1 Definition and properties

The Ginibre process, denoted by μ in the remainder of this paper, is defined as the determinantal process on $\mathbb{C}$ with integral kernel

$$K(z_1,z_2) = \frac{1}{\pi} e^{z_1\bar{z_2}} e^{-\frac{1}{2}(|z_1|^2+|z_2|^2)}, \quad z_1, z_2 \in \mathbb{C}, \tag{B.5}$$

with respect to $\lambda := \mathrm{d}\ell(z)$, the Lebesgue measure on $\mathbb{C}$ (i.e. $\mathrm{d}\ell(z) = \mathrm{d}x\,\mathrm{d}y$, when $z = x + iy$). It can be naturally decomposed as:

$$K(z_1,z_2) = \sum_{n\geq 0} \phi_n(z_1)\overline{\phi_n(z_2)}, \quad z_1, z_2 \in \mathbb{C},$$

where $\phi_n(z) := \frac{1}{\sqrt{\pi n!}} e^{-\frac{1}{2}|z|^2} z^n$, for $n \in \mathbb{N}$ and $z \in \mathbb{C}$. It can be easily verified that $(\phi_n)_{n\in\mathbb{N}}$ is an orthonormal family of $L^2(\mathbb{C}, \mathrm{d}\ell)$. In fact, $(\phi_n)_{n\in\mathbb{N}}$ is a dense subset of $L^2(\mathbb{C}, \mathrm{d}\ell)$. The Ginibre process μ verifies the following basic properties:

Proposition B.3.1. *The Ginibre process μ, i.e. the determinantal process with kernel K satisfies the following:*

- *μ is ergodic with respect to the translations on the plane.*
- *μ is isotropic.*
- *$\mu(\mathbb{C}) = +\infty$ almost surely, i.e. the Ginibre point process has an infinite number of points almost surely.*

Since μ has an infinite number of points almost surely, it is possible to simulate it directly. Therefore, in the rest of this paper, we are interested in modifying the kernel K in order to obtain versions of the Ginibre point process which will be usable in applications.

B.3.2 Truncated Ginibre point process

The first idea is to consider the truncated Ginibre kernel, defined for $N \in \mathbb{N}_*$ by

$$K^N(z_1, z_2) = \sum_{n=0}^{N-1} \phi_n(z_1)\overline{\phi_n(z_2)}, \quad z_1, z_2 \in \mathbb{C}. \tag{B.6}$$

This is in fact a truncation of the sum in (B.5). We also call μ^N the associated determinantal point process with intensity measure $\mathrm{d}\ell$. We remark that $\mu^N \to \mu$ weakly, when $N \to \infty$. As it is a projection kernel, we have seen in Section B.2 that μ^N has N points almost surely. μ^N is clearly not translation invariant anymore; however, it remains isotropic for the same reason that μ is. Physically, μ^N is the distribution of N polarized electrons in a perpendicular magnetic field, filling the N lowest Landau levels, as is remarked in [80]. As μ^N has N points almost surely, it is entirely characterized by its joint distribution p which is calculated in the following proposition.

Proposition B.3.2. *Let μ^N be the point process with kernel given by (B.6). Then, μ^N has N points almost surely and its joint density p is given by*

$$p(z_1, \dots, z_N) = \frac{1}{\pi^N} \prod_{p=0}^{N} \frac{1}{p!}\, e^{-\sum_{p=1}^{N} |z_p|^2} \prod_{1\le p<q\le N} |z_p - z_q|^2, \tag{B.7}$$

for $z_1, \dots, z_N \in \mathbb{C}$.

It is also known that the radii of the points of μ^N have the same distribution as independent gamma random variables. More precisely, we can find in [54] the following result:

Proposition B.3.3. *Let $\{X_1, \dots, X_N\}$ be the $N \in \mathbb{N}_*$ unordered points, distributed according to μ^N. Then, $\{|X_1|, \dots, |X_N|\}$ has the same distribution as $\{Y_1, \dots, Y_N\}$, where for $1 \le i \le N$, $Y_i^2 \sim \mathrm{gamma}(i, 1)$, and the Y_i are independent.*

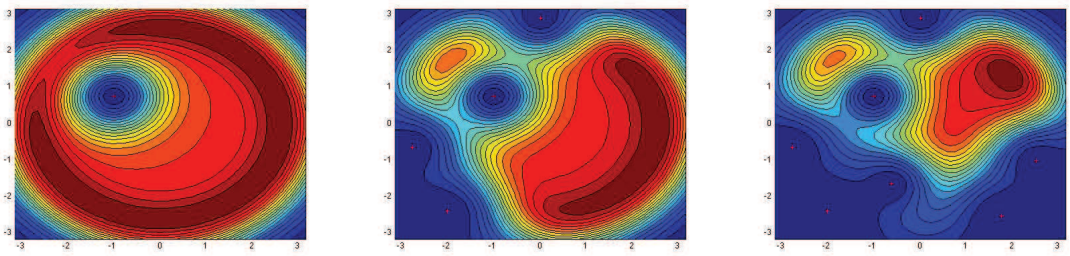
However, it should be noted that this does not yield a practical simulation technique, as the angle of $X_1, \dots, X_N$ are strongly correlated, and do not follow a known distribution.

We now move on to the problem of simulating a truncated Ginibre point process with kernel given by (B.6). Since μ^N has N points almost surely, there is no need to simulate the number of points. One only needs to simulate the position of the N points. For this specific case, there is in fact a more natural way of simulating the Ginibre process. Indeed, it was proven in [38] that the eigenvalues of an $N \times N$ hermitian matrix with complex gaussian entries are distributed according to μ^N. More precisely, consider a matrix $N := (N_{nm})_{1 \leq n,m \leq N}$, such that for $1 \leq n, m \leq N$,

$$N_{nm} = \frac{1}{\sqrt{2}} \left(N_{nm}^1 + i N_{nm}^2 \right),$$

where $N_{nm}^1, N_{nm}^2 \sim \mathcal{N}(0,1)$, $1 \leq n, m \leq N$ are independent centered gaussian random variables. Then, the eigenvalues of N are distributed according to μ^N. This is by far the most efficient way of simulating the truncated Ginibre process.

We also remark that we could have applied the general simulation technique in order to simulate the truncated Ginibre point process. However, the simulation procedure is much slower than calculating the eigenvalues of an $N \times N$ matrix. We still show the results of the algorithm of a realization of the resulting point process in the following. This allows proper visualization of the associated densities. We chose $a = 3$ and $N = 8$ in this example. We plot the densities p_i as color gradients before the simulation of the i-th point. We also mark by red points the previously simulated points. Therefore, the resulting point process consists of the red points in Figure B.3.4. The steps plotted in the following figure correspond to $i = 7, i = 4$, and $i = 1$ respectively.



However, one runs into a practical problem when simulating the truncated Ginibre process: the support of its law is the whole of $\mathbb{C}^N$. Recall that the joint law of μ^N is known to be given by (B.7) which has support on $\mathbb{C}^N$. Moreover, projecting onto a compact subset randomizes the number of points in the point process. Therefore, this first method is only useful in applications where point process need not be in a fixed compact.

B.3.3 Ginibre point process on a compact subset

We now consider more specifically the projection of the Ginibre process onto $\mathcal{B}_R$, and thus we consider the projection kernel $K_R := P_{\mathcal{B}_R} K P_{\mathcal{B}_R}$ of the integral operator K onto

$\mathrm{L}^2(\mathcal{B}_R, \mathrm{d}\ell)$, where $\mathcal{B}_R := \overline{\mathcal{B}(0,R)}$ is the closed ball of $\mathbb{C}$ of radius $R \geq 0$ with center 0. In this specific case, the kernel of the operator K_R takes the form:

$$K_R(z_1, z_2) = \sum_{n \geq 0} \lambda_n^R \phi_n^R(z_1) \overline{\phi_n^R(z_2)}, \tag{B.8}$$

where $\phi_n^R(z) := Z_{R,n}^{-1} \phi_n(z) 1_{z \in \mathcal{B}_R}$, $n \in \mathbb{N}$, $z \in \mathbb{C}$ and $Z_{R,n}^{-1} \in \mathbb{R}$ is a constant depending only on n. This result does not hold in general, but is due to the fact that $(\phi_n^R(\cdot))_{n \geq 0}$ is still an orthonormal family of $L^2(\mathcal{B}_R, \mathrm{d}z)$. Indeed, for $m, n \in \mathbb{N}$,

$$\begin{aligned}
\int_{\mathcal{B}_R} \phi_n^R(z) \overline{\phi_m^R(z)} \, \mathrm{d}\ell(z) &= Z_{R,n}^{-2} \left(\frac{1}{\sqrt{n!m!}} \int_0^R r^{n+m+1} e^{-r^2} \, \mathrm{d}r \right) \left(\frac{1}{\pi} \int_{-\pi}^{\pi} e^{i(n-m)\theta} \, \mathrm{d}\theta \right) \\
&= Z_{R,n}^{-2} 1_{n=m} \left(\frac{2}{n!} \int_0^R r^{2n+1} e^{-r^2} \, \mathrm{d}r \right) \\
&= Z_{R,n}^{-2} 1_{n=m} \frac{\gamma(n+1, R^2)}{n!},
\end{aligned}$$

where γ is the lower incomplete Gamma function defined as

$$\gamma(z, a) := \int_0^a e^{-t} t^{z-1} \, \mathrm{d}t,$$

for $z \in \mathbb{C}$ and $a \geq 0$. Hence, in the following, we shall take $Z_{R,n} := \sqrt{\frac{\gamma(n+1,R^2)}{n!}}$. Therefore, the associated eigenvalues are

$$\lambda_n^R := \int_{\mathcal{B}_R} |\phi_n(z)|^2 \, \mathrm{d}\ell(z) = Z_{R,n}^2 = \frac{\gamma(n+1, R^2)}{n!}.$$

As is expected, $0 \leq \lambda_n^R \leq 1$ for any $n \in \mathbb{N}, R \geq 0$, and $\lambda_n^R \xrightarrow[R \to \infty]{} 1$ for any $n \in \mathbb{N}$.

Now that we have specified the eigenfunctions and associated eigenvalues, the simulation of the Ginibre process on a compact is that of the determinantal point process with kernel given by (B.8). Therefore, the general case algorithm fully applies. The time-consuming step of the algorithm will be the simulation of the Bernoulli random variables. Recall that the cumulative distribution function of $T = \sup\{n \in \mathbb{N}_* \ / \ B_n = 1\}$ is given by:

$$F(m) = \sum_{n \leq m} \frac{\gamma(n+1, R^2)}{n!} \prod_{i=n+1}^{\infty} \frac{\Gamma(i+1, R^2)}{i!},$$

for $m \in \mathbb{N}_*$.

We remark that we can not simulate the Ginibre point process restricted to a compact in the same way as in the previous section. Indeed, taking a $N \times N$ matrix with complex gaussian entries, and conditioning on the points being in $\mathcal{B}_{\sqrt{N}}$ yields a determinantal point process with kernel (B.10), which is not our target point process, as the sum is truncated at N. Therefore, the method developed in the previous section does not apply here. Hence, the algorithm is twofold, and the first step goes as follows:

Algorithm 10 Simulation of the Ginibre process on a compact subset (Step 1)

evaluate numerically $R \leftarrow \prod_{i\geq 1} \frac{\Gamma(i+1,R^2)}{i!}$, for example by calculating $e^{\sum_{i=1}^{N} \ln(\frac{\Gamma(i+1,R^2)}{i!})}$, where N is chosen such that $\ln(\frac{\Gamma(N+1,R^2)}{N!}) < \epsilon$, ϵ given by the user.

sample $U \leftarrow \mathcal{U}([0,1])$ according to a uniform distribution on $[0,1]$.
$m \leftarrow 0$
while $U < R$ **do**
  $m \leftarrow m+1$
  $R \leftarrow \frac{m!\gamma(m+1,R^2)}{\gamma(m,R^2)\Gamma(m+1,R^2)} R$
end while
for $i = 0 \to m-1$ **do**
  $B_i \leftarrow \mathrm{Be}(\frac{\gamma(i+1,R^2)}{i!})$, where here $\mathrm{Be}(\lambda)$ is an independent drawing of a Bernoulli random variable of parameter λ
end for
if $m > 0$ **then**
  return $\{B_0, \dots, B_{m-1}, 1\}$
end if
if $m = 0$ **then**
  return $\{1\}$
end if

Remark. *The series* $\prod_{i\geq n} \frac{\Gamma(i+1,R^2)}{i!}$, *for* $n \in \mathbb{N}_*$ *is convergent since it is equal to* $\prod_{i\geq n}(1-\lambda_i^R)$ *which is convergent because* $\sum_{i\geq 0} \lambda_i^R < \infty$ *since the considered operator is locally trace-class.*

We write $\{B_0, \dots, B_{m-1}, 1\}$ for the value returned by the previous algorithm, with the convention that $\{B_0, \dots, B_{m-1}, 1\} = \{1\}$ if $m = 0$. Then by Theorem 7 of [45], the law of the Ginibre point process on a compact is the same as that of the determinantal point process of kernel

$$K(z_1, z_2) = \sum_{k=0}^{m} B_k \phi_n(z_1)\overline{\phi_n(z_2)}, \quad z_1, z_2 \in \mathbb{C}.$$

Now, we move onto the second part of the algorithm, which is this time straightforward as it suffices to follow the general simulation algorithm closely. It only remains to notice that the method described in this section yields a determinantal point process on $\mathcal{B}_{\sqrt{N}}$. In order to simulate the point process on $\mathcal{B}_a$, for $a \geq 0$, it suffices to apply a homothetic transformation to the N points, which translates to a homothety on the eigenvectors. To sum up, the simulation algorithm of the truncated Ginibre process on a centered ball of radius $a \geq 0$ is as follows:

We end this subsection by mentioning the difficulties arising in the simulation under the density p_i, $1 \leq i \leq N-1$. As is remarked in [56], in the general case, we have no choice but to simulate by rejection sampling and the Ginibre point process is no different (except the case $i = N-1$ which is a gaussian random variable). Therefore in practice, we draw a uniform random variable u on $\mathcal{B}_a$ and choose $p_i(u)/\sup_{y\in\mathcal{B}_a} p_i(y)$. Note that the authors

Algorithm 11 Simulation of the Ginibre process on a compact subset (Step 2)

define $\phi_k(z) = \frac{N}{\pi a^2 \gamma(k+1,N)} e^{-\frac{N}{2a^2}|z|^2} (\frac{Nz}{a^2})^k$, for $z \in \mathcal{B}_a$ and $0 \leq k \leq m$.
define $\mathbf{v}(z) := (\phi_{i_0}(z), \dots, \phi_{i_k}(z), \phi_m(z))$, for $z \in \mathcal{B}_a$, and where $\{i_0, \dots, i_k\} = \{0 \leq i \leq m-1 \ : \ B_i = 1\}$
set $N := k+2$
sample X_N from the distribution with density $p_N(x) = \|\mathbf{v}(x)\|^2/N, \ \ x \in \Lambda$
set $\mathbf{e}_1 = \mathbf{v}(X_N)/\|\mathbf{v}(X_N)\|$
for $i = N-1 \to 1$ **do**
  sample X_i from the distribution with density

$$p_i(x) = \frac{1}{i}\Big[\|\mathbf{v}(x)\|^2 - \sum_{j=1}^{N-i} |\mathbf{e}_j^* \mathbf{v}(x)|^2\Big]$$

  set $\mathbf{w}_i = \mathbf{v}(X_i) - \sum_{j=1}^{N-i} \Big(\mathbf{e}_j^* \mathbf{v}(X_i)\Big) \mathbf{e}_j, \quad \mathbf{e}_{N-i+1} = \mathbf{w}_i/\|\mathbf{w}_i\|$
end for
return $(X_1, \dots, X_N)$

in [56] give a closed form bound on p_i which is given by

$$p_i(x) \leq \frac{1}{i} \min_{i+1 \leq k \leq N} \left(K^N(x,x) - \frac{|K^N(x, X_k)|^2}{K^N(X_k, X_k)} \right), \tag{B.9}$$

where $X_{i+1}, \dots, X_N$ is the result of the simulation procedure up to step i. In practice however, the error made in the previous inequality is not worth the gain made by not evaluating $\sup_{y \in \mathcal{B}_a} p_i(y)$. Therefore, in our simulations, we have chosen not to use (B.9).

B.3.4 Truncated Ginibre process on a compact subset

In this subsection, we begin by studying the truncated Ginibre point process on a compact subset, and specifically discuss the optimal choice of the compact subset onto which we project. We begin by studying the general projection of the truncated Ginibre process onto a centered ball of radius $R \geq 0$ which is again a determinantal point process whose law can be explicited. To that end, we wish to study $K_R^N := P_{\mathcal{B}_R} K^N P_{\mathcal{B}_R}$ of the integral operator K onto $\mathrm{L}^2(\mathcal{B}_R, \mathrm{d}\ell)$. The associated kernel is given by

$$K_R^N(z_1, z_2) = \sum_{n=0}^{N-1} \lambda_n^R \phi_n^R(z_1) \overline{\phi_n^R(z_2)}, \tag{B.10}$$

for $z_1, z_2 \in \mathcal{B}_R$. The question of the Janossy densities of the associated determinantal process is not as trivial as the non-projected one. Indeed, μ_R^N does not have N points almost surely. However, it is known that it has less than N points almost surely (see e.g. [82]). Therefore, it suffices to calculate the Janossy densities $j_R^0, \dots, j_R^N$ to characterize the law of μ_R^N. These are given by the following proposition:

Proposition B.3.4. *The point process μ_R^N with kernel given by (B.10) has less than N*

points almost surely, and its Janossy densities are given by

$$j_R^k(z_1,\dots,z_k) = \frac{1}{\pi^k}\prod_{p=0}^{k-1}\frac{1}{p!}e^{-\sum_{p=1}^k |z_p|^2}\prod_{1\le i<j\le k}|z_i-z_j|^2 \sum_{\{i_1,\dots,i_k\}\subset\{1,\dots,N\}} |s_{\lambda(i_1,\dots,i_k)}(z_1,\dots,z_k)|^2,$$

for $0 \le k \le N$ *and* $z_1,\dots,z_k \in \mathcal{B}_R$.

Next, we wish to determine the optimal $R \ge 0$ onto which we project the truncated Ginibre process. In regards to this question, we recall that the particle density of the general Ginibre process is constant, and

$$\rho_1(z) = K(z,z) = \frac{1}{\pi},$$

for $z \in \mathbb{C}$. However, the particle density of the truncated Ginibre process is not constant. If we denote by ρ_n^N the n-th correlation function of μ^N, then we have

$$\rho_1^N(z) = \frac{1}{\pi}e^{-\frac{1}{2}|z|^2}\sum_{k=0}^{N-1}\frac{|z|^{2k}}{k!},$$

for $z \in \mathbb{C}$. As can be checked easily, we have $\int_{\mathbb{C}} \rho_1^N(z)\,\mathrm{d}z = N$ as well as

$$\rho_1^N(z) \le \frac{1}{\pi}, \quad \forall z \in \mathbb{C}, \tag{B.11}$$

and in fact it is known that $\rho_1^N(\sqrt{N}z) \xrightarrow[N\to\infty]{} \frac{1}{\pi}1_{|z|\le 1}$, which is known as the circular law in stochastic matrix theory. It therefore appears that it is optimal to project onto $\mathcal{B}_{\sqrt{N}}$. Moreover, the following bounds were obtained in [38].

Proposition B.3.5. *For* $|z|^2 < N+1$, *we have*

$$\frac{1}{\pi} - \rho_1^N(z) \le \frac{1}{\pi}e^{-|z|^2}\frac{|z|^{2N}}{N!}\frac{N+1}{N+1-|z|^2}.$$

For $|z|^2 \ge N$, *we have*

$$\rho_1^N(z) \le \frac{1}{\pi}e^{-|z|^2}\frac{|z|^{2N}}{N!}\frac{N}{|z|^2-N}.$$

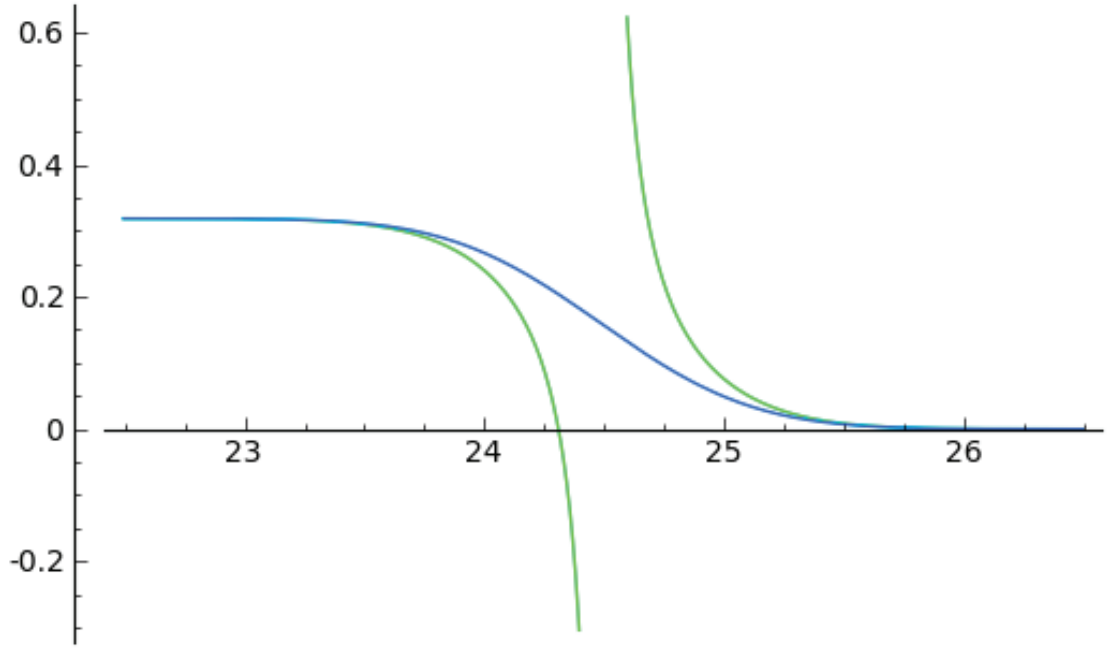
As was noticed in [38], if we set $|z| = \sqrt{N}+u$, for $-1 \le u \le 1$, both of the right hand sides of the inequalities in Proposition B.3.5 tend to

$$\frac{1}{2\sqrt{2}u\pi^{3/2}}e^{-2u^2},$$

as N tends to infinity. This is obtained by standard calculations involving in particular the Stirling formula. That is to say, for $|z| \le \sqrt{N}$, and $z = \sqrt{N}-u$,

$$\rho_1^N(\sqrt{N}-u) \ge \frac{1}{\pi} - \frac{1}{2\sqrt{2}u\pi^{3/2}}e^{-2u^2}, \tag{B.12}$$

Figure B.1: $\rho_1^N(|z|)$ for $N = 600$ and $|z|$ around $\sqrt{N}$ (blue). Upper and lower bounds obtained in (B.12) and (B.13) (green).



as well as for $|z| \geq \sqrt{N}$, and $z = \sqrt{N} + u$

$$\rho_1^N(\sqrt{N} + u) \leq \frac{1}{2\sqrt{2}u\pi^{3/2}} e^{-2u^2}, \tag{B.13}$$

as N tends to infinity. These bounds exhibit the sharp fall of the particle density around $|z| = \sqrt{N}$.

The previous results yield in particular the next proposition (see [38]).

Proposition B.3.6. *Let us write* $\delta(N) := \int_{|z|>\sqrt{N}} \rho_1^N(z)\, d\ell(z)$. *Then, we have*

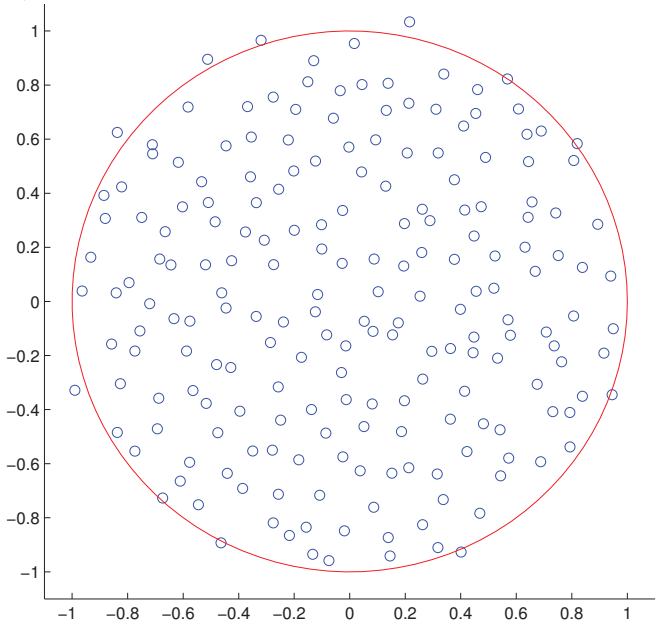
$$\delta(N) \sim \sqrt{\frac{N}{2\pi}}, \tag{B.14}$$

as $N \to \infty$.

Proposition B.3.6 means that as $N \to \infty$, the average number of particles falling outside of the $\mathcal{B}_{\sqrt{N}}$ is of the order of $\frac{1}{\sqrt{N}}$, as N tends to infinity. Therefore, from now on, we will consider the truncated Ginibre process of rank N projected onto $\mathcal{B}_{\sqrt{N}}$. Assume that we need to simulate μ^N on a compact subset. Then, we no longer control the number of points, i.e. there is again a random number of points in the compact subset, as seen in Figure B.2.

Therefore, our additional idea is to condition the number of points on being equal to N. As we have calculated previously in Proposition B.3.6, there is a number of particles falling

Figure B.2: A realization of μ^N for $N = 200$ (blue circles) renormalized to fit in the circle of radius 1 (in red)



outside of the ball of radius $\mathcal{B}_{\sqrt{N}}$ which grows as $\frac{1}{\sqrt{N}}$ as $N \to \infty$. Since the projection onto $\mathcal{B}_{\sqrt{N}}$ of the truncated Ginibre process takes the determinantal form (B.10), one can easily calculate the probability of all the points falling in $\mathcal{B}_{\sqrt{N}}$. Indeed, we have that

$$\mathbb{P}_{\mu^N}(\xi_{\mathcal{B}^c_{\sqrt{N}}} = \emptyset) = \prod_{n=0}^{N-1} \lambda_n^N = \prod_{n=0}^{N-1} \frac{\gamma(n+1, N)}{n!}. \tag{B.15}$$

It can be shown that this probability tends to 0 as N tends to infinity. That is, if we are required to simulate the Ginibre process on a compact conditionally on it having N points, the conditioning requires more and more computation time as N tends to infinity.

However, we are not forced to simulate the conditioning on there being N points. Instead, we introduce a new kernel, as well as the associated point process. We set

$$K_t^N(z_1, z_2) = \sum_{n=0}^{N-1} \phi_n^N(z_1)\overline{\phi_n^N(z_2)}, \quad z_1, z_2 \in \mathcal{B}_R, \tag{B.16}$$

and where ϕ_n^N corresponds to the function ϕ_n restricted to the compact $\mathcal{B}_{\sqrt{N}}$ (after renormalization). We emphasize that this is in fact $\mu^N|_{\mathcal{B}_{\sqrt{N}}}$ conditioned on there being N points in the compact $\mathcal{B}_{\sqrt{N}}$, this result being due to Theorem 7 in [45]. Moreover, the determinantal point process associated with this kernel benefits from the efficient simulations techniques developed in the previous subsection. Here, the fact that we can explicit the projection kernel associated with the conditioning is what ensures the efficiency of the simulation.

Let us start by proving that μ_t^N, the associated determinantal process with kernel K_t^N, converges to μ weakly as N tends to infinity. This is a consequence of Proposition B.2.3, as is proved in the following:

Theorem B.3.1. *We have that K_t^N converges uniformly on compact subsets to K as N tends to infinity. As a consequence, the associated determinantal measures converge weakly to the determinantal point process of kernel K.*

We now return to the problem of simulating the determinantal point process with kernel given by (B.16). As it is a projection process, it is efficiently simulated according to the basic algorithm described in the full version. On the other hand, the time-consuming step of generating the Bernoulli random variables is not necessary anymore, as we are working conditionally on there being N points. Lastly, the method described in this section yields a determinantal point process on $\mathcal{B}_N$. As before, in order to simulate on $\mathcal{B}_a$, we to apply a homothetic transformation to the N points, which translates to a homothety on the eigenvectors. To sum up, the simulation algorithm of the truncated Ginibre process on a centered ball of radius $a \geq 0$ is as follows:

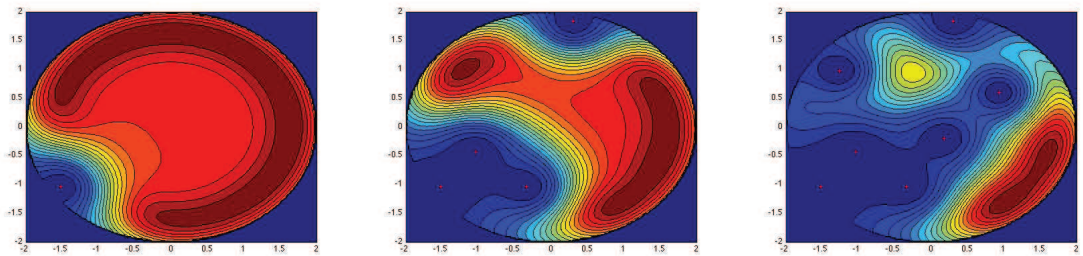
Algorithm 12 Simulation of the truncated Ginibre process on a compact

define $\phi_k(z) = \frac{N}{\pi a^2 \gamma(k+1,N)} e^{-\frac{N}{2a^2}|z|^2} (\frac{Nz}{a^2})^k$, for $z \in \mathcal{B}_N$ and $0 \leq k \leq N-1$.
define $\mathbf{v}(z) := (\phi_0(z), \ldots, \phi_{N-1}(z))$, for $z \in \mathcal{B}_N$.
sample X_N from the distribution with density $p_N(x) = \|\mathbf{v}(x)\|^2/N, \ \ x \in \Lambda$
set $\mathbf{e}_1 = \mathbf{v}(X_N)/\|\mathbf{v}(X_N)\|$
for $i = N-1 \to 1$ **do**
  sample X_i from the distribution with density

$$p_i(x) = \frac{1}{i}\Big[\|\mathbf{v}(x)\|^2 - \sum_{j=1}^{N-i} |\mathbf{e}_j^* \mathbf{v}(x)|^2\Big]$$

  set $\mathbf{w}_i = \mathbf{v}(X_i) - \sum_{j=1}^{N-i} \Big(\mathbf{e}_j^* \mathbf{v}(X_i)\Big)\, \mathbf{e}_j, \quad \mathbf{e}_{N-i+1} = \mathbf{w}_i/\|\mathbf{w}_i\|$
end for
return $(X_1, \ldots, X_N)$

The resulting process is a determinantal point process of kernel (B.16). Its support is on the compact $\mathcal{B}_a$ and has N points almost surely. We now give a brief example of the results of the algorithm applied for $a = 2$ and $N = 9$ at steps $i = 8$, $i = 5$, and $i = 2$ respectively. We have plotted the densities used for the simulation of the next point.



This determinantal point process presents the advantage of being easy to use in simulations, as well as having N points almost surely. Moreover, Theorem B.3.1 proves its convergence to the Ginibre point process as N tends to infinity.

Bibliography

[1] 3GPP. TR 36.913: Requirements for further advancements for evolved universal terrestrial radio access (E-UTRA). *Tech. Rep.*, Mar. 2009.

[2] 3GPP. Telecommunications management; self-healing OAM; concepts and requirements. *Tech. Rep.*, 3GPP TS 32.541 v1.6.1, 2010.

[3] T. Al-Meshhadany and K. Al-Agha. VCB by means of soft2hard handover in WCDMA. In *Mobile and Wireless Communications Network, 2002. 4th International Workshop on*, pages 487–491, 2002.

[4] O. Aliu, A. Imran, M. Imran, and B. Evans. A survey of self organisation in future cellular networks. *Communications Surveys Tutorials, IEEE*, 15(1):336–361, 2013.

[5] M. Amirijoo, L. Jorguseski, T. Kurner, R. Litjens, M. Neuland, L. Schmelz, and U. Turke. Cell outage management in LTE networks. In *Wireless Communication Systems, 2009. ISWCS 2009. 6th International Symposium on*, pages 600–604, 2009.

[6] G. W. Anderson, A. Guionnet, and O. Zeitouni. *An introduction to random matrices*, volume 118 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 2010.

[7] A.Nguyen, N.Milosavljević, Q.Fang, J.Gao, and L.J.Guibas. Landmark selection and greedy landmark-descent routing for sensor networks. In *Proceedings of IEEE INFOCOM 2007*, 2007.

[8] M. J. B. Appel and R. P. Russo. The maximum vertex degree of a graph on uniform points in [0,1]d. *Advances in Applied Probability*, 29(3):pp. 567–581, 1997.

[9] I. Ashraf, L. T. W. Ho, and H. Claussen. Improving energy efficiency of femtocell base stations via user activity detection. In *Wireless Communications and Networking Conference (WCNC), 2010 IEEE*, pages 1–5, 2010.

[10] X. Bai, S. Kumar, D. Xuan, Z. Yun, and T. H. Lai. Deploying wireless sensors to achieve both coverage and connectivity. In *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '06, pages 131–142, New York, NY, USA, 2006. ACM.

[11] Y. Bejerano. Simple and efficient k-coverage verification without location information. In *Proc. IEEE INFOCOM*, pages 897–905, Phoenix, Arizona, USA, Apr. 2008.

[12] Y. Bejerano. Coverage verification without location information. *IEEE Trans. Mobile Comput.*, 11(4):631–643, Apr. 2012.

[13] E. T. Bell. Exponential polynomials. *Ann, Math.*, 35:258–277, 1934.

[14] B. Bollobás and P. Erdös. Cliques in random graphs. *Mathematical Proceedings of the Cambridge Philosophical Society*, 80(3):419–427, 1976.

[15] F. Bornemann. On the scaling limits of determinantal point processes with kernels induced by sturm-liouville operators. arXiv:math-ph/1104.0153, to appear, 2011.

[16] P. Bratley and B. L. Fox. Algorithm 659: Implementing sobol's quasirandom sequence generator. *ACM Trans. Math. Softw.*, 14(1):88–100, Mar. 1988.

[17] E. Campos-Nañez, A. Garcia, and C. Li. A game-theoretic approach to efficient power management in sensor networks. *Oper. Res.*, 56(3):552–561, 2008.

[18] F. Chazal, V. de Silva, and S. Oudot. Persistence stability of geometric complexes. arXiv:math.AT/1207.3885, to appear, 2012.

[19] V. Chvatal. A Greedy Heuristic for the Set-Covering Problem. *Mathematics of Operations Research*, 4(3):233–235, 1979.

[20] D. Daley and D. V. Jones. *An introduction to the theory of point processes*. Springer, 2002.

[21] S. Das, S. Sen, and R. Jayaram. A structured channel borrowing scheme for dynamic load balancing in cellular networks. In *Distributed Computing Systems, 1997., Proceedings of the 17th International Conference on*, pages 116–123, 1997.

[22] S. Das, H. Viswanathan, and G. Rittenhouse. Dynamic load balancing through coordinated scheduling in packet data systems. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 1, pages 786–796 vol.1, 2003.

[23] V. de Silva and G. Carlsson. Topological estimation using witness complexes. *IEEE Symposium on Point-based Graphic*, pages 157–166, 2004.

[24] V. de Silva and R. Ghrist. Coordinate-free coverage in sensor networks with controlled boundaries via homology. *International Journal of Robotics Research*, 25, december 2006.

[25] V. de Silva and R. Ghrist. Coverage in sensor networks via persistent homology. *Algebraic & Geometric Topology*, 7:339–358, 2007.

[26] L. Decreusefond, E. Ferraz, H. Randriam, and A. Vergne. Simplicial homology of random configurations. *Advances in Applied Probability*, 46(2):1–20, 2014.

[27] L. Decreusefond, I. Flint, and A. Vergne. Efficient simulation of the Ginibre process. hal-00869259, Oct. 2013.

[28] L. Decreusefond, P. Martins, and A. Vergne. Clique number of random geometric graphs. hal-00864303, Mar. 2013.

[29] I. Dietrich and F. Dressler. On the lifetime of wireless sensor networks. *ACM Trans. Sen. Netw.*, 5(1):5:1–5:39, Feb. 2009.

[30] 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(1-2):29–58, 2012.

[31] L. Du, J. Bigham, and L. Cuthbert. An intelligent geographic load balance scheme for mobile cellular networks. In *Computer Communications and Networks, 2002. Proceedings. Eleventh International Conference on*, pages 348–353, 2002.

[32] N. Dunford and J. T. Schwartz. *Linear operators. Part I.* Wiley Classics Library. John Wiley & Sons Inc., New York, 1988. General theory, With the assistance of William G. Bade and Robert G. Bartle, Reprint of the 1958 original, A Wiley-Interscience Publication.

[33] Q. Fang, J. Gao, and L. Guibas. Locating and bypassing routing holes in sensor networks. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 4, pages 2458 – 2468 vol.4, march 2004.

[34] Q. Fang, J. Gao, L. Guibas, V. de Silva, and L. Zhang. GLIDER: Gradient landmark-based distributed routing for sensor networks. In *Proc. IEEE Conference on Computer Communications (INFOCOM)*, 2005.

[35] T. Fujii and S. Nishioka. Selective handover for traffic balance in mobile radio communications. In *Communications, 1992. ICC '92, Conference record, SUPERCOMM/ICC '92, Discovering a New World of Communications., IEEE International Conference on*, pages 1840–1846 vol.4, 1992.

[36] H. Georgii and H. J. Yoo. Conditional intensity and Gibbsianness of determinantal point processes. *J. Stat. Phys.*, 118(1-2):55–84, 2005.

[37] R. Ghrist and A. 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, Piscataway, NJ, USA, 2005. IEEE Press.

[38] J. Ginibre. Statistical ensembles of complex, quaternion, and real matrices. *J. Mathematical Phys.*, 6:440–449, 1965.

[39] A. Goel, S. Rai, and B. Krishnamachari. Monotone properties of random geometric graphs have sharp thresholds. *Ann. Appl. Probab.*, 15(4):2535–2552, 2005.

[40] A. Hatcher. *Algebraic Topology.* Cambridge University Press, 2002.

[41] D. Haussler and E. Welzl. Epsilon-nets and simplex range queries. In *Proceedings of the second annual symposium on Computational geometry*, SCG '86, pages 61–71, New York, NY, USA, 1986. ACM.

[42] S. Haykin. Cognitive radio: brain-empowered wireless communications. *Selected Areas in Communications, IEEE Journal on*, 23(2):201–220, 2005.

[43] T. W. Haynes, S. T. Hedetniemi, and P. J. Slater. *Fundamentals of domination in graphs*, volume 208 of *Monographs and Textbooks in Pure and Applied Mathematics.* Marcel Dekker Inc., New York, 1998.

[44] S. Hmlinen, H. Sanneck, and C. Sartori. *LTE Self-Organising Networks (SON): Network Management Automation for Operational Efficiency*. Wiley Publishing, 1st edition, 2012.

[45] J. B. Hough, M. Krishnapur, Y. Peres, and B. Virág. Determinantal processes and independence. *Probab. Surv.*, 3:206–229 (electronic), 2006.

[46] J. B. Hough, M. Krishnapur, Y. Peres, and B. Virág. *Zeros of Gaussian analytic functions and determinantal point processes*, volume 51 of *University Lecture Series*. American Mathematical Society, Providence, RI, 2009.

[47] S. Irani. Coloring inductive graphs on-line. *Algorithmica*, 11(1):53–72, 1994.

[48] Y. Ito. Generalized poisson functionals. *Probab. Theory Related Fields*, 77:1–28, 1988.

[49] P. Jiang, J. Bigham, and J. Wu. Self-organizing relay stations in relay based cellular networks. *Comput. Commun.*, 31(13):2937–2945, Aug. 2008.

[50] T. Kaczyński, M. Mrozek, and M. Ślusarek. Homology computation by reduction of chain complexes. *Comput. Math. Appl.*, 35(4):59–70, 1998.

[51] M. Kahle. Random geometric complexes. *Discrete & Computational Geometry*, 45:553–573, 2011. 10.1007/s00454-010-9319-3.

[52] K. F. Kee, L. Sparks, D. C. Struppa, and M. Mannucci. Social groups, social media, and higher dimensional social structures: A simplicial model of social aggregation for computational communication research. *Communication Quarterly*, 61(1):35–58, 2013.

[53] D. Kim, B. Shin, D. Hong, and J. Lim. Self-configuration of neighbor cell list utilizing E-UTRAN nodeB scanning in LTE systems. In *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*, pages 1–5, 2010.

[54] E. Kostlan. On the spectra of Gaussian matrices. *Linear Algebra Appl.*, 162/164:385–388, 1992. Directions in matrix theory (Auburn, AL, 1990).

[55] M. Krivelevich and B. Sudakov. The phase transition in random graphs: A simple proof. *Random Structures & Algorithms*, 2012.

[56] F. Lavancier, J. Møller, and E. Rubak. Determinantal point process models and statistical inference. arXiv:math.ST/1205.4818, to appear, 2012.

[57] J. Li and R. Jantti. On the study of self-configuration neighbour cell list for mobile WiMAX. *Next Generation Mobile Applications, Services and Technologies, International Conference on*, 0:199–204, 2007.

[58] R. Luce and A. Perry. A method of matrix analysis of group structure. *Psychometrika*, 14:95–116, 1949.

[59] O. Macchi. The coincidence approach to stochastic point processes. *Advances in Appl. Probability*, 7:83–122, 1975.

[60] J. Matoušek. On the l2-discrepancy for anchored boxes. *J. Complex.*, 14(4):527–556, Dec. 1998.

[61] D. Matula. On the complete subgraphs of a random graph. In *Proc. Of the Second Chapel Hill Conference on Combinatorial Mathematics and Its Applications*, pages 356–369, 1970.

[62] N. Miyoshi and T. Shirai. A cellular network model with ginibre configurated base stations. *Research Rep. on Math. and Comp. Sciences (Tokyo inst. of tech.)*, 2012.

[63] T. Moore, R. Drost, P. Basu, R. Ramanathan, and A. Swami. Analyzing collaboration networks using simplicial complexes: A case study. In *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*, pages 238–243, 2012.

[64] K. Morrison. Rapidly recovering from the catastrophic loss of a major telecommunications office. *Communications Magazine, IEEE*, 49(1):28–35, 2011.

[65] C. Mueller, M. Kaschub, C. Blankenhorn, and S. Wanke. A cell outage detection algorithm using neighbor cell list reports. In K. Hummel and J. Sterbenz, editors, *Self-Organizing Systems*, volume 5343 of *Lecture Notes in Computer Science*, pages 218–229. Springer Berlin Heidelberg, 2008.

[66] A. Muhammad and M. Egerstedt. Control using higher order laplacians in network topologies. In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems, Kyoto, Japan*, Kyoto,Japan, July 2006.

[67] A. Muhammad and A. Jadbabaie. Decentralized computation of homology groups in networks by gossip. In *American Control Conference, 2007. ACC '07*, pages 3438 –3443, july 2007.

[68] T. Müller. Two-point concentration in random geometric graphs. *Combinatorica*, 28:529–545, 2008.

[69] A. Nasif and B. Mark. Opportunistic spectrum sharing with multiple cochannel primary transmitters. *Wireless Communications, IEEE Transactions on*, 8(11):5702–5710, 2009.

[70] E. M. Palmer. *Graphical evolution*. Wiley, New-York, 1985.

[71] O. Parzanchevski and R. Rosenthal. Simplical complexes: spectrum, homology and random walks. arXiv:math.CO/1211.6775, to appear, january 2013.

[72] G. Peccati, J. Solé, M. Taqqu, and F. Utzet. Stein's method and normal approximation of poisson functionals. *Annals of Probability*, 38(2):443–478, 2010.

[73] M. Penrose. *Random Geometric Graphs (Oxford Studies in Probability)*. Oxford University Press, USA, July 2003.

[74] M. D. Penrose. Focusing of the scan statistic and geometric clique number. *Adv. in Appl. Probab.*, 34(4):739–753, 2002.

[75] M. D. Penrose and J. E. Yukich. Central limit theorems for some graphs in computational geometry. *Annals of Applied Probability*, 11(4):1005—1041, 2001.

[76] M. D. Penrose and J. E. Yukich. Weak laws of large numbers in geometric probability. *Annals of Applied Probability*, 13(1):277—303, 2003.

[77] M. D. Penrose and J. E. Yukich. Limit theory for point processes in manifolds. *Annals of Applied Probability*, 2013.

[78] N. Privault. *Stochastic Analysis in Discrete and Continuous Settings.* Springer, 2009.

[79] N. Privault. Stochastic analysis in discrete and continuous settings with normal martingales. *Lecture Notes in Mathematics*, 1982, 2009.

[80] A. Scardicchio, C. E. Zachary, and S. Torquato. Statistical properties of determinantal point processes in high-dimensional Euclidean spaces. *Phys. Rev. E (3)*, 79(4):041108, 19, 2009.

[81] T. Shirai and Y. Takahashi. Random point fields associated with certain Fredholm determinants. II. Fermion shifts and their ergodic and Gibbs properties. *Ann. Probab.*, 31(3):1533–1564, 2003.

[82] A. Soshnikov. Determinantal random point fields. *Uspekhi Mat. Nauk*, 55(5(335)):107–160, 2000.

[83] H. Tamura and K. R. Ito. A canonical ensemble approach to the fermion/boson random point processes and its applications. *Comm. Math. Phys.*, 263(2):353–380, 2006.

[84] R. Tarjan and A. Trojanowski. Finding a maximum independent set. *SIAM Journal on Computing*, 6(3):537–546, 1977.

[85] G. L. Torrisi and E. Leonardi. Large deviations of the interference in the Ginibre network model. arXiv:cs.IT/1304.2234, to appear, 2013.

[86] A. Vergne, L. Decreusefond, and P. Martins. Reduction algorithm for simplicial complexes. In *INFOCOM, 2013 Proceedings IEEE*, pages 475–479, 2013.

[87] A. Vergne, I. Flint, L. Decreusefond, and P. Martins. Homology based algorithm for disaster recovery in wireless networks. hal-00800520, Mar. 2013.

[88] G. Wang, G. Cao, and T. La Porta. Movement-assisted sensor deployment. In *Proc. IEEE INFOCOM*, volume 4, pages 2469–2479, Mar. 2004.

[89] F. Yan, P. Martins, and L. Decreusefond. Connectivity-based distributed coverage hole detection in wireless sensor networks. In *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, pages 1 –6, dec. 2011.

[90] F. Yan, P. Martins, and L. Decreusefond. Accuracy of homology based approaches for coverage hole detection in wireless sensor networks. In *ICC 2012*, June 2012.

[91] F. Yan, A. Vergne, P. Martins, and L. Decreusefond. Homology-based Distributed Coverage Hole Detection in Wireless Sensor Networks. hal-00783403, Oct. 2012.

[92] C. Zhang, Y. Zhang, and Y. Fang. Detecting coverage boundary nodes in wireless sensor networks. In *Networking, Sensing and Control, 2006. ICNSC '06. Proceedings of the 2006 IEEE International Conference on*, pages 868 –873, 0-0 2006.

[93] C. Zhang, Y. Zhang, and Y. Fang. Localized algorithms for coverage boundary detection in wireless sensor networks. *Wirel. Netw.*, 15(1):3–20, Jan. 2009.

[94] A. Zomorodian and G. Carlsson. Computing persistent homology. *Discrete & Computational Geometry*, 33:249–274, 2005. 10.1007/s00454-004-1146-y.

Topologie algébrique appliquée aux réseaux de capteurs

Anaïs VERGNE

RESUME : La représentation par complexes simpliciaux fournit une description mathématique de la topologie d'un réseau de capteurs, c'est-à-dire sa connectivité et sa couverture. Dans ces réseaux, les capteurs sont déployés aléatoirement en grand nombre afin d'assurer une connectivité et une couverture parfaite. Nous proposons un algorithme qui permet de déterminer quels capteurs mettre en veille, sans modification de topologie, afin de réduire la consommation d'énergie. Notre algorithme de réduction peut être appliqué à tous les types de complexes simpliciaux, et atteint un résultat optimal. Pour les complexes simpliciaux aléatoires géométriques, nous obtenons des bornes pour le nombre de sommets retirés, et trouvons des propriétés mathématiques pour le complexe simplicial obtenu. En cherchant la complexité de notre algorithme, nous sommes réduits à calculer le comportement asymptotique de la taille de la plus grande clique dans un graphe géométrique aléatoire. Nous donnons le comportement presque sûr de la taille de la plus grande clique pour les trois régimes de percolation du graphe géométrique.

Dans la deuxième partie, nous appliquons la représentation par complexes simpliciaux aux réseaux cellulaires, et améliorons notre algorithme de réduction pour répondre à de nouvelles demandes. Tout d'abord, nous donnons un algorithme pour la planification automatique de fréquences, pour la configuration automatique des réseaux cellulaires de la nouvelle génération bénéficiant de la technologie SON. Puis, nous proposons un algorithme d'économie d'énergie pour l'optimisation des réseaux sans fil. Enfin, nous présentons un algorithme pour le rétablissement des réseaux sans fil endommagés après une catastrophe. Dans ce dernier chapitre, nous introduisons la simulation des processus ponctuels déterminantaux dans les réseaux sans fil.

ABSTRACT : Simplicial complex representation gives a mathematical description of the topology of a wireless sensor network, i.e., its connectivity and coverage. In these networks, sensors are randomly deployed in bulk in order to ensure perfect connectivity and coverage. We propose an algorithm to discover which sensors are to be switched off, without modification of the topology, in order to reduce energy consumption. Our reduction algorithm can be applied to any type of simplicial complex and reaches an optimum solution. For random geometric simplicial complexes, we find boundaries for the number of removed vertices, as well as mathematical properties for the resulting simplicial complex. The complexity of our reduction algorithm boils down to the computation of the asymptotical behavior of the clique number of a random geometric graph. We provide almost sure asymptotical behavior for the clique number in all three percolation regimes of the geometric graph.

In the second part, we apply the simplicial complex representation to cellular networks and improve our reduction algorithm to fit new purposes. First, we provide a frequency auto-planning algorithm for self-configuration of SON in future cellular networks. Then, we propose an energy conservation fot the self-optimization of wireless networks. Finally, we present a disaster recovery algorithm for any type of damaged wireless network. In this last chapter, we also introduce the simulation of determinantal point processes in wireless networks.

TELECOM ParisTech

EDITE

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY