Fusion de l'information dans les réseaux de capteurs : application à la surveillance de phénomènes physiques

Nisrine Ghadban

Thèse
de doctorat
de l'UTT

Nisrine GHADBAN

Fusion de l'information dans les réseaux de capteurs : application à la surveillance de phénomènes physiques

Spécialité :
Optimisation et Sûreté des Systèmes

2015TROY0037 **Année 2015**

Thèse en cotutelle avec l'Université Libanaise - Beyrouth - Liban

Université Libanaise

utt
université de technologie
Troyes

THESE

pour l'obtention du grade de

DOCTEUR de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

Spécialité : OPTIMISATION ET SURETE DES SYSTEMES

présentée et soutenue par

Nisrine GHADBAN

le 14 décembre 2015

Fusion de l'information dans les réseaux de capteurs : application à la surveillance de phénomènes physiques

JURY

M. R. LENGELLÉ	PROFESSEUR DES UNIVERSITES	Président
M. F. ABDALLAH	PROFESSEUR	Rapporteur
Mme J. FARAH	PROFESSEUR	Examinateur
M. C. FRANCIS	PROFESSEUR	Directeur de thèse
M. P. HONEINE	PROFESSEUR DES UNIVERSITES	Directeur de thèse
Mme L. OUKHELLOU	DIRECTRICE DE RECHERCHE	Rapporteur

Personnalités invitées

M. M. BERAR	MAITRE DE CONFERENCES
Mme F. MOURAD-CHEHADE	MAITRE DE CONFERENCES

A vous mes chers parents, je dédie ce modeste travail...

Remerciements

Il me sera très difficile de remercier tout le monde car c'est grâce à l'aide de nombreuses personnes que j'ai pu mener cette thèse à son terme. A l'issue de la rédaction de cette recherche, je suis convaincue que la thèse est loin d'être un travail solitaire. En effet, je n'aurais jamais pu réaliser ce travail doctoral sans le soutien d'un grand nombre de personnes dont la générosité, la bonne humeur et l'intérêt manifestés à l'égard de ma recherche m'ont permis de progresser dans cette phase délicate de "l'apprenti-chercheur".

En premier lieu, je tiens à remercier mes directeurs de thèse, Monsieur Paul HONEINE et Monsieur Clovis FRANCIS, pour la confiance qu'ils m'ont accordée en acceptant d'encadrer ce travail doctoral, pour leurs multiples conseils et pour toutes les heures qu'ils ont consacrées à diriger cette recherche. Leur compétence, leur rigueur scientifique et leur clairvoyance m'ont beaucoup appris. Ils ont été et resteront des moteurs de mon travail de chercheur. Je suis ravie d'avoir travaillé en leur compagnie car outre leur appui scientifique, ils ont toujours été là pour me soutenir et me conseiller au cours de l'élaboration de cette thèse.

Monsieur Fahed ABDALLAH et Madame Latifa OUKHELLOU m'ont fait l'honneur d'être rapporteurs de ma thèse, et je les en remercie, de même que pour leur participation au Jury. Ils ont également contribué par leurs nombreuses remarques et suggestions à améliorer la qualité de ce mémoire, et je leur en suis très reconnaissante. Je tiens à remercier Madame Joumana FARAH et Madame Farah MOURAD-CHEHADE pour avoir accepté de participer à mon jury de thèse et pour leur participation scientifique ainsi que le temps qu'elles ont consacré à ma recherche. Je remercie également Monsieur Régis LENGELLE pour l'honneur qu'il me fait d'être dans mon jury de thèse. Monsieur Maxime BERAR m'a fait l'honneur de participer au Jury de soutenance ; je l'en remercie profondément.

J'adresse aussi mes remerciements à Madame Zeinab SAAD, Professeur à l'Université Libanaise et ancienne Doyenne de l'école doctorale de l'UL, pour avoir mis en place la convention de thèse en cotutelle du réseau UT-INSA avec l'UL, permettant ainsi des séjours entre le Liban et la France et une collaboration entre les deux pays et précisément les deux universités, et pour son équipe talentueuse qui a su gérer nos demandes.

Mes remerciements vont aussi à mes amis, Thérèse, Tatiana, Leila, Mirna, qui, avec cette question récurrente, "quand est-ce que tu la soutiens cette thèse ?", bien qu'angoissante en période fréquente de doutes, m'ont permis de ne jamais dévier de mon objectif final. Je remercie encore toutes les personnes formidables que j'ai rencontrées l'UTT pour la bonne ambiance de travail mais également pour les nombreux bons moments

passés ensembles : Johnny, Joyce, Sandy, Patric, Elie, Roy, Zeinab, Abdo, Heping. Ils m'ont permis d'oublier momentanément le travail dans des soirées, repas, sorties vélo ou autres.

Enfin, les mots les plus simples étant les plus forts, j'adresse toute mon affection à ma famille : ma mère Noha, mon Père Adel, ma sœur Lara et mon frère Elias. Malgré mon éloignement, leur confiance, leur tendresse, leur amour me portent et me guident tous les jours. Merci pour avoir fait de moi ce que je suis aujourd'hui.

Merci à tous et à toutes.

Soyons reconnaissants aux personnes qui nous
donnent du bonheur ; elles sont les charmants jardiniers
par qui nos âmes sont fleuries.
Marcel Proust (1871-1922).

Troyes, le 14 décembre 2015.

Titre Fusion de l'information dans les réseaux de capteurs : application à la surveillance de phénomènes physiques

Résumé Cette thèse apporte des solutions clés à deux problèmes omniprésents dans les réseaux de capteurs sans fil, à savoir la précision des mesures acquises dans les régions à faible couverture et la dimensionnalité sans cesse grandissante des données collectées. La première contribution de cette thèse est l'amélioration de la couverture de l'espace à surveiller par le biais de la mobilité des capteurs. Nous avons recours aux méthodes à noyaux en apprentissage statistique pour modéliser un phénomène physique tel que la diffusion d'un gaz. Nous décrivons plusieurs schémas d'optimisation pour améliorer les performances du modèle résultant. Nous proposons plusieurs scenarios de mobilité des capteurs. Ces scénarios définissent d'une part l'ensemble d'apprentissage du modèle et d'autre part le capteur mobile. La seconde contribution de cette thèse se situe dans le contexte de la réduction de la dimensionnalité des données collectées par les capteurs. En se basant sur l'analyse en composantes principales, nous proposons à cet effet des stratégies adaptées au fonctionnement des réseaux de capteurs sans fil. Nous étudions également des problèmes intrinsèques aux réseaux sans fil, dont la désynchronisation entre les nœuds et la présence de bruits de mesures et d'erreurs de communication. Des solutions adéquates avec l'approche Gossip et les mécanismes de lissage sont proposées. L'ensemble des techniques développées dans le cadre de cette thèse est validé sur un réseau de capteurs sans fil qui estime le champ de diffusion d'un gaz.

Mots-clés
- Réseaux de capteurs (technologie)
- Apprentissage automatique
- Réduction des données (statistique)
- Analyse en composantes principales
- Traitement du signal

Title Information aggregation in sensor networks : application to monitoring of physical activities

Abstract This thesis investigates two major problems that are challenging the wireless sensor networks (WSN) : the measurements accuracy in the regions with a low density of sensors and the growing volume of data collected by the sensors. The first contribution of this thesis is to enhance the collected measurements accuracy, and hence to strengthen the monitored space coverage by the WSN, by means of the sensors mobility strategy. To this end, we address the estimation problem in a WSN by kernel-based machine learning methods, in order to model some physical phenomenon, such as a gas diffusion. We propose several optimization schemes to increase the relevance of the model. We take advantage of the sensors mobility to introduce several mobility scenarios. Those scenarios define the training set of the model and the sensor that is selected to perform mobility based on several mobility criteria. The second contribution of this thesis addresses the dimensionality reduction of the set of collected data by the WSN. This dimensionality reduction is based on the principal component analysis techniques. For this purpose, we propose several strategies adapted to the restrictions in WSN. We also study two well-known problems in wireless networks : the non-synchronization problem between nodes of the network, and the noise in measures and communication. We propose appropriate solutions with Gossip-like algorithms and smoothing mechanisms. All the techniques developed in this thesis are validated in a WSN dedicated to the monitoring of a physical species leakage such as the diffusion of a gas.

Keywords
- Sensor networks
- Machine learning
- Data reduction
- Principal components analysis
- Signal processing

Table des matières

Préface

Sommaire

Les réseaux de capteurs sans fil ont reçu une attention considérable au cours de la dernière décennie en raison de leur faible coût, de la facilité de leur déploiement et de leur aptitude à mettre en œuvre des techniques efficaces de surveillance. Initialement destinés à des applications militaires, les réseaux des capteurs sans fil couvrent maintenant un large spectre d'applications dans des zones industrielles et civiles (Conner et al. 2004, Czubak et Wojtanowski 2009, Steere et al. 2000). Parmi les nombreuses applications, on peut citer la surveillance de l'environnement (Mainwaring et al. 2002), le suivi de patients (Hande et al. 2006), la détection des feux de forêt (Hefeeda et Bagheri 2007), le suivi de cible (Teng et al. 2010), etc (Puccinelli et Haenggi 2005a). Les réseaux de capteurs sans fil sont constitués de *capteurs intelligents* déployés en grand nombre en vue de collecter et transmettre les données environnementales, d'une manière autonome. L'autonomie des capteurs dans un réseau sans fil impose plusieurs contraintes, avec des réductions de la complexité du calcul, de la consommation de l'énergie et du débit de communication. Un nouveau défi clé concerne l'émergence des réseaux de capteurs mobiles, qui attirent beaucoup d'attention de nos jours, comme c'est le cas d'une flotte de drones par exemple. Un capteur est alors apte à se déplacer vers une nouvelle position pour y extraire une information pertinente. Dans ce contexte plusieurs aspects sont importants : la collaboration entre les capteurs, la détection et l'isolation des capteurs défaillants, la pertinence de l'information délivrée par le capteur, la complexité calculatoire, le flux de données sans cesse grandissant,... Nous sommes ainsi à la recherche de l'information pertinente, tout en respectant des contraintes de couverture, d'énergie de déplacement, d'énergie de communication et de limitation du débit sans cesse grandissant des données.

FIGURE 1 – *Exemple d'un capteur sans fil (source : (LibeliumWorld 2009)).*

Ce mémoire apporte des solutions à un ensemble de problèmes omniprésents dans les réseaux de capteurs sans fil. Dans ce but, nous commençons le présent chapitre par décrire en premier lieu les réseaux de capteurs sans fil et leurs applications. Ensuite nous introduisons le problème de mobilité pour optimiser la couverture des réseaux de capteurs mobiles. Ensuite, nous exposons un problème majeur dans les réseaux de capteurs qui est la grande dimension des données acquises et nous signalons l'importance de la réduction de dimensionnalité. Nous exposons enfin l'organisation du document.

Réseau de capteurs sans fil

Les réseaux de capteurs sans fil (RCSF) proposent une solution économique et facilement déployable pour la surveillance à distance de l'environnement. Cette technologie repose sur les progrès réalisés dans les domaines de la microélectronique, de la micromécanique, et des technologies de communication sans fil. Un RCSF est constitué d'un grand nombre de nœuds, dits micro-capteurs ou capteurs intelligents (Akyildiz et al. 2002a, Vieira et al. 2003, Sohraby et al. 2007). Il existe deux grandes familles de réseaux de capteurs : structurés et non structurés. Un RCSF non structuré est formé d'une forte densité de nœuds de capteurs déployés d'une manière ad hoc, c'est-à-dire selon une distribution aléatoire, dans le champ sous surveillance. Dans un RCSF non structuré, la maintenance du réseau, tel que la gestion de connectivité et la détection des défaillances, est difficile à cause du grand nombre de nœuds à distribution aléatoire. Dans un RCSF structuré, les nœuds sont déployés d'une manière pré-planifiée, à des endroits fixes pré-déterminés. L'avantage d'un réseau structuré est que le coût d'entretien et de gestion des nœuds est souvent plus faible. De

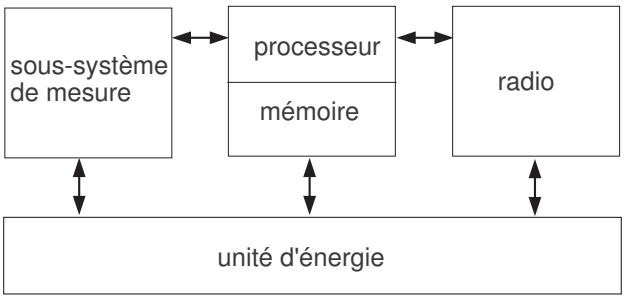

FIGURE 2 – *Architecture d'un capteur sans fil.*

plus, un nombre réduit de nœuds peut être déployé puisque les nœuds sont placés à des endroits précis pour fournir une couverture totale de la région, tandis que le déploiement ad hoc peut avoir des régions non couvertes.

Chaque nœud est un petit appareil capable de mesurer, de traiter, de stocker, et de transmettre des données de l'environnement. Ces données peuvent correspondre à des mesures de pression, température, concentration d'un gaz ou d'un polluant, son, vibration, etc. Il est constitué principalement des composantes suivantes :

- un système de mesure est une unité d'acquisition qui lie le nœud avec le monde physique, en mesurant des grandeurs physiques,
- un calculateur est une unité de traitement, par exemple un microprocesseur, qui est capable de traiter les données numériques,
- une mémoire est une unité pour le stockage des informations, par exemple des données et des algorithmes,
- un module de télécommunication qui permet de transmettre et de recevoir des signaux d'autres nœuds, par exemple une radio de courte portée,
- un système d'alimentation d'énergie qui alimente tous les modules, très souvent formé d'une pile non échangeable. Une alimentation secondaire qui récolte l'énergie de l'environnement, par exemple en utilisant des panneaux solaires, induit un coût monétaire significatif.

La FIGURE 2 schématise ces différentes composantes. La miniaturisation impose plusieurs contraintes aux micro-capteurs. Avec une pile inéchangeable, leur autonomie impose des contraintes d'économie d'énergie. Ils ne possèdent pas de grandes capacités de stockage et de calcul. En outre, la communication radio est une tâche consommatrice d'énergie et elle est identifiée dans de nombreux déploiements comme le facteur principal de l'épuisement de la batterie du capteur (Akyildiz et al. 2002b). L'émission ou la réception d'un paquet est un consommateur d'énergie plus important que quelques opérations de calcul élémentaires. La réduction du montant des transmissions de données a donc été reconnue comme une question centrale dans la conception de réseaux de capteurs sans

fil (Ilyas et al. 2004). Les contraintes de limitation de ressources, d'énergie en particulier, sont au fondement d'un nouveau paradigme : une optimisation collaborative, où chaque capteur résout localement un sous-problème d'optimisation, en collaborant de proche-en-proche avec ses voisins (Puccinelli et Haenggi 2005b). Des algorithmes à faible complexité de calcul doivent y être implémentés.

Notons que les contraintes de conception dépendent largement de l'application envisagée dans l'environnement à surveiller. Cet environnement joue un rôle essentiel dans la détermination de la taille du réseau, du système de déploiement et de la topologie du réseau. Pour les environnements intérieurs, moins de nœuds sont nécessaires pour former un réseau dans un espace limité, alors que les environnements extérieurs nécessitent plus de nœuds pour couvrir une plus grande surface. Un déploiement ad hoc est préféré au déploiement pré-planifiée lorsque l'environnement est inaccessible par les humains ou lorsque le réseau est composé de centaines à des milliers de nœuds. Les obstructions dans l'environnement peuvent également limiter la communication entre les nœuds qui, à son tour, affecte la connectivité (ou la topologie) du réseau. Les travaux de recherche sur les réseaux de capteurs visent à répondre aux contraintes susmentionnées en développant de nouveaux algorithmes adaptés, soit à partir de protocoles inédits ou en réexaminant des protocoles existants.

Applications

Les domaines d'applications du RCSF peuvent être essentiellement classés en deux catégories : surveillance et suivi. Les applications de surveillance comprennent la surveillance intérieure/extérieure de l'environnement, la surveillance du bien-être et de la santé, l'automatisation des processus et des usines, et la surveillance sismique et structurelle. Les applications de suivi comprennent la localisation et le suivi de cibles mobiles, telles que des animaux, des humains ou des véhicules. Nous décrivons ci-dessous des exemples d'applications qui ont été testées dans un environnement réel.

Domaine militaire

Comme pour la plupart des technologies, les RCSF étaient initialement destinés à des applications militaires. Ils sont utilisés par les militaires pour un certain nombre de fins telles que la surveillance de l'activité militaire dans des régions étendues et la protection des forces déployées. Étant équipés de capteurs appropriés, ces réseaux peuvent permettre la détection de mouvement de l'ennemi, l'identification des forces ennemies et l'analyse de leur mouvement et leur progrès. Dans ce domaine d'application, les réseaux de capteurs sans fil ont tout à fait un avantage sur les autres réseaux, car les attaques ennemies peuvent endommager ou détruire une partie des nœuds, mais l'échec des nœuds dans les RCSF n'affecte pas l'ensemble du réseau grâce au caractère collaboratif qui permet une auto-adaptation du réseau. Parmi les utilisations possibles de RCSF dans le domaine militaire on trouve (Lee et al. 2009, Watthanawisuth et al. 2011) :

- Suivi de l'ennemi et classification des cibles : les objets en mouvement peuvent être détectés en utilisant des capteurs spécialement conçus. Ce système aide surtout dans la détection des soldats et des véhicules armés.
- Surveillance du champ de bataille : des zones critiques et des frontières sont étroitement surveillées en utilisant les réseaux de capteurs pour obtenir des informations sur toute activité des ennemis dans ce domaine.
- Evaluation des dégâts du champ de bataille : les réseaux de capteurs peuvent être déployés après la bataille ou les attaques afin de recueillir des informations pour estimer les dégâts.
- Détection des attaques nucléaires, radiologiques, biologiques et chimiques (NRBC) : des réseaux de capteurs équipés de capteurs appropriés sont déployés afin de détecter et alerter contre des attaques de type NRBC.
- Système de ciblage : des capteurs peuvent être intégrés dans des armes. Des informations sur la cible comme la distance et l'angle sont alors collectées et envoyées au tireur dans le but de mieux viser.
- Déminage : ce problème est devenu un problème mondial suite à la contamination de vastes territoires par des mines explosives. Des capteurs biologiques sensibles aux espèces chimiques composant ces mines sont en cours de développement. Le déploiement de ces capteurs contribue au déminage de ces territoires et de sauver des vies.

Domaine environnemental

Les RCSF jouent un rôle fondamental dans le suivi environnemental. Ils sont déployés pour la surveillance de l'habitat, la détection d'inondation, la détection des incendies de forêt, etc. Parmi les différentes applications, on peut citer :

- Détection des incendies de forêt : des millions de capteurs sont déployés pour détecter le feu, estimer son point de départ et prédire son évolution. Les réseaux de capteurs sans fil permettent une aide à la décision rapide, avant que le feu ne devienne incontrôlable (Yu et al. 2005).
- Détection d'inondation : les informations recueillies par des capteurs pour la pluie, le niveau de l'eau et la météo peuvent prévoir les menaces possibles d'inondation fournissant ainsi de l'aide pour la gestion des catastrophes.
- Surveillance de la pollution : des réseaux de capteurs sont déployés pour détecter des gaz toxiques et examiner leurs évolutions spatio-temporelles (De Vito et Fattoruso 2012).
- Surveillance de la faune : des RCSF sont déployés pour la recherche sur les activités de la faune. Le comportement des espèces menacées est ainsi étudié pour contribuer à leur préservation. Autres activités étudiées à l'aide de capteurs sont la localisation avec des cartes de migration des oiseaux et d'autres animaux, les habitudes alimentaires, le comportement de chasse...(Xu 2002). Les RCSF ont plusieurs avantages par rapport aux méthodes traditionnelles, puisqu'ils fournissent une large couverture (terrestre et aquatique) et des

longues périodes de surveillance, avec des données disponibles directement aux chercheurs en surveillance continue.
– Autres applications de gestion des catastrophes : les réseaux de capteurs sans fil sont également utilisés pour la gestion des diverses catastrophes comme les tremblements de terre et la surveillance des glissements de terrain.

Domaine industriel

En environnement industriel, l'utilisation de capteurs câblés est souvent trop lourde à mettre en œuvre. En outre, les fils conducteurs sont contraignantes, en particulier lorsque des pièces mobiles sont impliquées. Les capteurs sans fil permettent une installation rapide d'équipements, tout en atteignant des endroits qui ne seraient pas accessibles par des câbles fixes. Les RCSF sont utilisés en industrie comme un moyen de réduire les coûts et améliorer les performances de la machine et la maintenabilité. Ils permettent, à titre d'exemple :

– Suivi de l'état de fonctionnement des machines : la "santé" de la machine est surveillée grâce à des mesures de vibrations ou d'usure au niveau de lubrification. Les RCSF permettent l'insertion de capteurs dans des régions qui sont peu accessibles par les techniciens comme les machines tournantes et les zones dangereuses ou restreintes (Hou et Bergmann 2012).
– Surveillance de l'environnement industriel : les RCSF sont utilisés pour la surveillance de l'état de l'environnement dans l'industrie, comme la surveillance du niveau d'eau dans les réservoirs d'une centrale nucléaire et de la température dans les réfrigérateurs, ou encore la quantité de gaz ou de polluants dans l'atmosphère.
– Gestion du contrôle d'inventaire : chaque produit dans un entrepôt peut avoir un capteur attaché qui communique avec d'autres capteurs ou un centre de traitement, ce qui permet aux utilisateurs de trouver l'emplacement exact de l'article à l'aide d'algorithmes d'auto-localisation de capteur et avoir d'autres informations quantitatives et qualitatives des articles stockés.

Domaine médical

Les réseaux de capteurs sont aussi largement utilisés dans le secteur des soins de santé. Les services fournies rendent les choses plus faciles pour l'équipe médicale et améliorent la qualité de vie des patients grâce aux réseaux de capteurs portés (Honeine et al. 2011). Dans le domaine médical, les RCSF peuvent :

– Suivre les patients : les capteurs portés par le patient (parfois implémentés dans son corps) permettent de surveiller des données physiologiques du patient, telles que son mouvement ou encore les fréquences cardiaque et respiratoire, afin de détecter des anomalies comme une crise cardiaque, une perte de conscience ou une chute. Les RCSF permettent ainsi de surveiller à distance les patients, qui peuvent désormais se déplacer librement à domicile (Zakrzewski et al. 2009, Suryadevara et Mukhopadhyay 2012).

- Contrôler l'administration de médicaments : les capteurs intelligents peuvent intégrer également un module qui permet d'injecter des médicaments, par voie veineuse par exemple.
- Interagir avec des dispositifs implantés dans l'organisme : c'est le cas en particulier avec des stimulateurs cardiaques (ou *pacemaker*) de nouvelle génération, afin de mieux suivre et contrôler les troubles de la conduction cardiaque.

Domaines divers

Les RCSF couvrent un large spectre d'applications autres que celles décrites ci-dessus. Nous citons parmi ces applications :

- Surveillance des infrastructures : des RCSF sont déployés dans le cadre de la sécurité des infrastructures en contrôlant les vibrations et les conditions du materiel (Ong et al. 2008). En outre, les bâtiments critiques, les monuments, et les stades peuvent être protégés contre les attaques terroristes à l'aide de réseaux de capteurs, en utilisant par exemple un réseau de caméras collaboratives.
- Domotique : des RCSF sont utilisés pour des applications domestiques, dont le contrôle à distance via Internet des appareils ménagers comme la machine à laver, les climatiseurs, les aspirateurs, les portails et fenêtres, etc. Ils permettent également de contrôler la consommation de l'énergie et de l'eau (Chen et Wang 2006, Trinchero et al. 2011).
- Domaine automobile : des capteurs sont implémentés dans des véhicules pour fournir un dispositif afin de détecter et de suivre les vols de voitures, ou encore afin de contrôler la circulation,... (Chong et Kumar 2003).
- Agriculture de précision : les RCSF permettent l'étude de l'environnement local afin d'optimiser l'agriculture (Callaway et al. 2002). Ils sont utilisés pour surveiller les conditions comme la température, l'humidité, l'humidité du sol, la vitesse et la direction du vent, ...

Réseaux de capteurs mobiles

Les capteurs sans fil, n'ayant pas une infrastructure fixe, sont capables d'être mobiles. Les capteurs mobiles (robots) permettent d'explorer des environnements inconnus et effectuer une variété de tâches. Ils sont classés en trois grands groupes selon l'environnement en question :

- Robots terrestres : ce type de robots est conçu pour fonctionner tout en maintenant un contact permanent avec le sol (Gerhart et Abbott 1998, Chen 2007).
- Drones volant : les drones volants permettent d'éviter les problèmes des obstacles physiques rencontrés par les robots terrestres, tout simplement en les survolant. Ce type de robots mobiles est souvent utilisé pour les applications de surveillance et de détection/suivi de cibles (Burdakov et al. 2010, Maza et al. 2010, Doitsidis et al. 2012).
- Robots aquatiques : ce type de robots introduit un ensemble spécifique de défis en raison des caractéristiques de l'environnement

de déploiement. Les robots sous-marins sont confrontés aux problèmes de la communication dans le milieu aquatique et aux problèmes de localisation et de mobilité qui nécessitent des techniques adaptées aux robots en immersion dans l'eau (Nimbalkar et Pompili 2008, Yoon et Qiao 2011, Yuh 2000).

Dans de nombreux déploiements de réseaux de capteurs, une répartition optimale est inconnue jusqu'à ce que les nœuds de capteurs commencent à collecter et traiter les données. Ce déploiement optimal est généralement impossible sans l'ajout de la mobilité. Le déplacement des capteurs dans la région sous surveillance permet de bénéficier des avantages suivants :

- Longue durée de vie du réseau : en se déplaçant, les capteurs rendent la transmission plus dispersée, et l'énergie moins dissipée. En effet, dans les réseaux qui sont clairsemés, ou lorsque des nœuds stationnaires meurent, les nœuds mobiles peuvent manœuvrer pour relier les voies de communication perdus ou faibles. Cela est impossible avec les réseaux de capteurs statiques, dans lequel les données à partir des nœuds morts ou déconnectés seraient tout simplement perdus. De même, dans les réseaux centralisés fixes, les nœuds les plus proches du centre de fusion vont mourir plus tôt, parce qu'ils doivent transmettre et relier également les messages de données provenant des nœuds plus loins. En utilisant des capteurs mobiles, ce problème est éliminé et la durée de vie du réseau est prolongée (Gandham et al. 2003).
- Amélioration de couverture et du ciblage : puisque les capteurs sont le plus souvent déployés aléatoirement, la mobilité permet une meilleure couverture, ou une proximité de la cible (Wang et al. 2005, Amundson et al. 2008). La mobilité permet également des applications de détection plus polyvalentes. Par exemple, en surveillant les feux de forêt, les capteurs mobiles sont capables de maintenir une distance de sécurité du périmètre de feu, et de fournir des mises à jour régulières aux pompiers.
- Amélioration des performances : l'aspect de la communication sans fil est de plus en plus important dans les systèmes multi-robots pour améliorer leurs performances globales (Tiderko et al. 2008). Pour décider son prochain mouvement efficace, un robot mobile peut avoir besoin d'informations en provenance d'autres robots. La communication permet non seulement la fusion de données, mais contribue également à développer une vue individuelle du réseau et de l'environnement physique.
- Meilleure fidélité de données : un nœud mobile peut être utilisé pour transporter des données à un point destiné. Il est utile lorsque le canal de transmission sans fil est en mauvais état, ou si l'épuisement prématuré de l'énergie est possible (Li et Mohapatra 2007). Le nombre réduit de communications dû à la mobilité va augmenter la probabilité de transmissions réussies.

Nous différencions trois types de mobilité (Labiod 2006, Shorey et Choon 2005) :

- Mobilité non contrôlée : dans ce premier cas, les capteurs du réseau ne possèdent aucun pouvoir de se déplacer. En outre, il n'y a pas des

entités qui pourraient déplacer les capteurs dans le réseau. La mobilité des capteurs est incontrôlée suite à une force externe, comme par exemple le vent soufflant qui déplace les capteurs dans la région sous surveillance.
- Mobilité assistée : ce second type de mobilité suppose un réseau de capteurs composé de capteurs statiques qui sont incapables de se déplacer de manière autonome. Cependant, ces capteurs sont habituellement montés sur différents types d'agents mobiles (robots, véhicules, animaux, personnes, etc.) qui leur fournissent la mobilité. Bien que ce type de mouvement ne soit pas contrôlable, il peut éventuellement être modélisé mathématiquement.
- Mobilité contrôlée : le troisième type de mobilité suppose que le réseau est entièrement ou partiellement composé de capteurs mobiles (robots mobiles) qui peuvent être déplacés suite à une commande interne ou externe. Cette propriété donne la possibilité d'améliorer les performances du réseau dans différentes applications.

Dans cette thèse, nous profitons de la mobilité contrôlée en utilisant des robots pouvant se déplacer dans différentes directions ; le choix de la direction et la distance parcourue sont alors optimisés pour une estimation d'un champ de diffusion d'un gaz.

Dimension de données dans un réseau de capteurs

Un réseau de capteurs sans fil est composé généralement de plusieurs nœuds de capteurs autonomes, à faible coût et à faible puissance. Ces nœuds recueillent des données de l'environnement et collaborent pour transmettre les données acquises dans le but d'un traitement ultérieur. Les nœuds pourraient être équipés de nombreuses modalités hétérogènes, telles que des capteurs thermiques, acoustiques, chimiques, optiques, infrarouge ou sismique, pour surveiller les volcans (Werner-Allen et al. 2006), détecter les intrus (Zong et al. 2006, Zhu et Huang 2007) ou effectuer de nombreuses autres tâches d'estimation, de détection et de classification (Doherty et al. 2001, Nevat et al. 2015, Lee et Choi 2008, Jawhar et al. 2011). En raison de cette diversité, les RCSF ont un énorme potentiel pour développer des applications puissantes, chacune avec ses propres caractéristiques et besoins. Le développement d'algorithmes efficaces qui conviennent à de nombreux scénarios d'applications différentes est une tâche difficile, surtout avec l'augmentation du nombre d'observations acquises avec le temps. Les mesures des capteurs contiennent beaucoup de redondance, soit dans les dimensions de mesure d'un seul capteur, soit entre les dimensions de mesure des différents capteurs en raison de la corrélation spatiale, soit dans la dimension temporelle des mesures.

Réduire la complexité en termes de communication et de calcul est essentiel puisque les réseaux de capteurs sont des systèmes qui fonctionnent avec des ressources extrêmement limitées (Cetin et al. 2006). Les ressources peuvent être conservées si les capteurs ne transmettent pas de données non pertinentes ou redondantes. Malheureusement, il est généralement inconnu à l'avance quelles dimensions de mesure ou combinaison de dimensions sont les plus utiles pour la tâche de détection ou de classification en question. La transmission de données non pertinentes et

redondantes peut être évitée grâce à la réduction de la dimensionnalité. Ainsi, une donnée de petite dimension représente-t-elle les mesures. Cette donnée réduite est transmise par les capteurs à un centre de fusion ou à un autre capteur selon le type du réseau, et elle est la base du calcul de la tâche considérée.

Les techniques mathématiques les plus connues pour la réduction de dimensionnalité sont classées sous deux approches :

- Une approche basée sur la projection des données de grandes dimensions sur des sous-espaces de représentation bien choisis. Dans cette classe on retrouve l'analyse en composantes principales (Jolliffe 1986, Abdi et Williams 2010), les projections préservées localement (He et Niyogi 2003), le positionnement multidimensionnel (Cox et Cox 2000), l'analyse en composantes principales à noyaux (Schölkopf et al. 1999, Kallas et al. 2010),...
- Une approche basée sur la représentativité dans l'espace de mesure en exploitant des critères de complémentarité et de redondance d'information, tels que l'entropie, le contraste, la corrélation, ou encore l'information mutuelle. Dans cette classe, on retrouve la propagation d'affinité (Jia et al. 2008), la sélection progressive des bandes spectrales (Fisher et Chang 2010),...

Dans cette thèse, nous utilisons l'analyse en composantes principales pour réduire la dimension des données mesurées par les capteurs. Nous proposons des stratégies de calcul adaptées aux contraintes imposées dans les réseaux de capteurs sans fil.

Organisation du document

Dans ce mémoire, nous étudions deux problèmes majeurs dans les réseaux de capteurs sans fil, et proposons des solutions adaptées. Le premier problème concerne la surveillance et le suivi de la propagation d'un phénomène physique tel la température, l'émission d'un gaz ou d'un polluant, la pression, etc. Nous proposons une estimation d'un champ de diffusion par les réseaux de capteurs sans fil, à partir de mesures spatio-temporelles acquises par les capteurs. Cette approche se base sur la mise en œuvre des méthodes d'apprentissage statistique, et en particulier les méthodes à noyaux. Nous proposons différents scénarios de mobilité de capteurs afin d'augmenter les performances. Le second problème concerne la réduction de dimensionnalité. Ayant des contraintes de communications et de calcul, il est indispensable de réduire la dimensionnalité de la grande base de données collectées par les capteurs. Dans ce but, nous proposons plusieurs stratégies à la volée, c'est-à-dire avec des communications de proche en proche entre les capteurs. La pertinence des algorithmes proposés est présentée dans le cadre de réseaux de capteurs sans restreindre le large spectre d'applications qui peuvent tirer profit de l'étude présentée dans ce document. Le contenu de chaque chapitre est rappelé en quelques lignes ci-dessous.

Chapitre 1 : Méthodes à noyaux

Ce chapitre introduit brièvement les méthodes à noyaux en apprentissage statistique. L'idée principale de l'apprentissage statistique est de déterminer des modèles et des relations à partir des données recueillies, sans aucune connaissance du système étudié. Les méthodes dites à noyaux sont des méthodes non linéaires qui reposent sur la mise en œuvre de modèles linéaires dans un espace transformé (grâce aux noyaux reproduisants). Nous décrivons le concept de noyaux reproduisants et les espaces de Hilbert associés. Nous présentons l'astuce du noyau et le Théorème de Représentation. Nous concluons ce chapitre par deux exemples clés de méthodes à noyaux, la régression par moindres carrés et l'analyse en composantes principales, qui seront réexaminés et revisités tout au long du manuscrit dans le cadre des RCSF.

Chapitre 2 : Estimation d'un champ de diffusion et mobilité

Ce chapitre traite le problème de modélisation de la distribution spatiale d'un champ de diffusion. D'une manière générale, on s'intéresse aux phénomènes de diffusion classiques, avec des applications sur la diffusion d'un gaz, d'un polluant, ou encore de la température. Dans le but d'estimer une telle distribution aux endroits dépourvus de capteurs, et vu les applications considérées, il est nécessaire d'avoir recours à des méthodes d'apprentissage adéquates. Nous utilisons les méthodes à noyaux pour la modélisation et nous profitons de la mobilité de certains capteurs afin de diminuer l'erreur d'estimation. Plusieurs scénarios de mobilités sont présentés dans ce chapitre et leurs performances analysées.

Chapitre 3 : Analyse en composantes principales dans les RCSF

Dans ce chapitre, nous traitons la réduction de dimension des données acquises par les capteurs, en utilisant l'analyse en composantes principales. Nous étudions le problème d'estimer l'axe principal le plus pertinent, tout en évitant le calcul de la matrice de covariance des données. Nous proposons, selon la topologie du réseau, des stratégies non coopératives et coopératives adaptées aux réseaux de capteurs sans fil, en minimisant à la volée une fonction coût appropriée. Nous présentons plusieurs fonctions coût possibles et nous étudions la convergence des stratégies proposées.

Chapitre 4 : Généralisation des stratégies de l'ACP en réseau

Dans ce chapitre, nous étendons notre étude sur la mise en œuvre de l'ACP dans le cadre des réseaux de capteurs sans fil. D'abord, nous généralisons le problème en considérant l'estimation de plusieurs axes principaux. Ensuite, nous étudions en particulier deux problèmes bien connus dans les réseaux, qui sont la désynchronisation entre les nœuds dans un réseau décentralisé et la présence de bruits de mesures et de communication. Nous proposons des solutions adéquates avec l'approche Gossip en communication asynchrone et les mécanismes de lissage.

Publications

Cette thèse a fait l'objet de plusieurs publications :

- **N. Ghadban,** P. Honeine, C. Francis, F. Mourad-Chehade, J. Farah, and M. Kallas, Estimation locale d'un champ de diffusion par modèles à noyaux. Actes de la 14-ème conférence ROADEF de la Société Française de Recherche Opérationnelle et Aide à la Décision (ROADEF), Troyes, France, 13–15 Février, 2013.
- **N. Ghadban,** P. Honeine, C. Francis, F. Mourad-Chehade, J. Farah, and M. Kallas, Mobilité d'un réseau de capteurs sans fil basée sur les méthodes à noyau. Actes du 24-ème Colloque GRETSI sur le Traitement du Signal et des Images (GRETSI), Brest, France, 3–6 September, 2013.
- **N. Ghadban,** P. Honeine, C. Francis, F. Mourad-Chehade and J. Farah, Diffusion Strategies For In-Network Principal Component Analysis. Proc. 24th IEEE workshop on Machine Learning for Signal Processing (MLSP), Reims, France, 21–24 September, 2014.
- **N. Ghadban,** P. Honeine, C. Francis, F. Mourad-Chehade, and J. Farah, Mobility using first and second derivatives for kernel-based regression in wireless sensor networks. Proc. 21st International Conference on Systems, Signals and Image Processing (IWSSIP), Dubrovnik, Croatia, 12–15 May, 2014.
- **N. Ghadban,** P. Honeine, C. Francis, F. Mourad-Chehade, and J. Farah, Strategies for principal component analysis in wireless sensor networks. Proc. eighth IEEE Sensor Array and Multichannel Signal Processing Workshop (SAM), A Coruña, Spain, 22–25 June, 2014.
- **N. Ghadban,** P. Honeine, C. Francis, F. Mourad-Chehade, and J. Farah,Gossip algorithms for principal component analysis in networks. Proc. 23th European Conference on Signal Processing (EUSIPCO), Nice, France, pp. 1–5, 31 Aug.–4 Sept., 2015.
- **N. Ghadban,** P. Honeine, F. Mourad-Chehade, C. Francis, and J. Farah, In-Network Principal Component Analysis With Diffusion Strategies. International Journal of Wireless Information Networks, submitted.

Première partie

Mobilité de capteurs sans fil basée sur les méthodes à noyaux

Introduction

Sommaire

L'apprentissage statistique a reçu une attention considérable au cours des deux dernières décennies, principalement pour le traitement des problèmes non linéaires. Plusieurs applications, en reconnaissance des formes, extraction de caractéristiques, analyse de séries temporelles, régression, détection et discrimination à deux ou plusieurs classes, mettent à disposition des données très complexes. Ainsi, les modèles linéaires traditionnels ne s'appliquent-ils pas ; d'où la nécessité de modèles non linéaires et d'algorithmes appropriés pour estimer leurs paramètres. Grâce à l'astuce du noyau et la théorie des noyaux reproduisants, les méthodes à noyaux permettent de transformer implicitement les données dans un espace de caractéristiques non linéaires de grande dimension, ce qui permet de construire des modèles et des règles de décision non linéaires avec essentiellement le même coût de calcul que celles des cas linéaires. Les méthodes à noyaux ont montré des performances remarquables pour un large spectre d'applications.

Dans cette partie, nous profitons des récentes avancées des méthodes à noyaux, notamment dans le domaine de la régression non linéaire. Nous traitons le problème de l'adaptation des échantillons de l'ensemble d'apprentissage. Nous prenons comme application la modélisation de la distribution d'un champ de diffusion mesuré par un réseau de capteurs sans fil (RCSF). Un modèle est alors construit afin d'estimer cette distribution en tout point de la région sous surveillance. L'objectif de cette partie est de traiter le problème de mobilité de capteurs afin d'améliorer la précision du modèle de régression, en optimisant le ré-échantillonnage. Bien que nous nous intéressons à la mobilité dans un RCSF pour optimiser l'ensemble d'apprentissage, de nombreux domaines d'application peuvent grandement bénéficier de ce travail, notamment dans les domaines de l'automatique et l'excitation physique ou chimique, où les échantillons d'entrée pourraient être modifiées par l'utilisateur afin d'exploiter de nouveaux régimes de fonctionnement du système étudié.

Mobilité

Dans cette thèse, nous traitons le problème de l'estimation dans un réseau de capteurs sans fil. Les défis des réseaux de capteurs ont donné lieu à un domaine de recherche très actif en apprentissage statistique, avec des méthodes à noyaux pour la régression et la reconnaissance de formes, y compris des problèmes tels que la localisation (Nguyen et al. 2005a), la détection (Nguyen et al. 2005b), et la régression (Honeine et al. 2009; 2010). Les capteurs sont souvent aléatoirement déployés dans la région sous surveillance. Ce déploiement aléatoire risque de ne pas assurer une bonne couverture de la région. Afin de surmonter ce problème, des capteurs mobiles se déplacent d'une manière qui optimise le modèle afin d'améliorer l'estimation de la grandeur physique. Dans ce document, certains capteurs sont supposés être des robots ayant une mobilité contrôlée, et donc ils se déplacent de façon à recueillir des informations plus pertinentes.

Dans ce qui suit, nous décrivons les modèles de mobilité contrôlée des capteurs/robots dans les RCSF. Pour cette raison, nous considérons N capteurs déployés dans une région $\mathbb{X}$. Soient $\boldsymbol{x}_i \in \mathbb{X}$ la position du capteur i, $1 \leq i \leq N$ et $y_i \in \mathbb{R}$ la mesure acquise par ce capteur de la quantité physique étudiée, comme par exemple la température ou la concentration de gaz. Soit $\mathcal{V}_i$ le voisinage du capteur i, c'est-à-dire l'ensemble des indices des capteurs voisins du capteur i.

Modèle aléatoire

La mobilité aléatoire représente un mouvement irrégulier des capteurs. Un capteur mobile part de sa position actuelle vers une nouvelle position avec une vitesse et une direction aléatoires. Les mouvements se produisent soit pour une durée constante, soit pour une distance parcourue constante. Plusieurs modèles de mobilité aléatoire existent. Par exemple, dans (Bergamo et Lopes 2012), un capteur i, à un instant t, se déplace de la position $\boldsymbol{x}_{i,t-1}$ à la position $\boldsymbol{x}_{i,t}$ selon la règle

$$\boldsymbol{x}_{i,t} = \boldsymbol{x}_{i,t-1} + \boldsymbol{q}_{i,t},$$

où $\boldsymbol{q}_{i,t}$ est une variable aléatoire qui suit la loi gaussienne de moyenne nulle et de variance $1/N^2$. Un autre exemple est le modèle de mobilité Manhattan (Buruhanudeen et al. 2007) qui utilise une topologie de route en grille. Ce modèle est principalement proposé pour le mouvement dans les zones urbaines, où les rues sont organisées et les nœuds mobiles sont autorisés à se déplacer uniquement dans deux directions orthogonales. A chaque intersection, le nœud mobile peut tourner à gauche, à droite ou aller tout droit avec une certaine probabilité. Nous trouvons aussi le modèle *Reference Point Group Mobility (RPGM)* proposé par Jayakumar et Ganapathi (2008), où les nœuds sont divisés en groupes et chaque groupe a un chef (ou *leader*). La mobilité du chef est aléatoire et les membres de son groupe le suivent avec un certain écart. A chaque instant, chaque nœud choisit aléatoirement une vitesse et une direction déviées de celle du chef de groupe.

Les performances de la mobilité peuvent être améliorées en remplaçant le critère aléatoire par des règles de décision bien précises. Les modèles

cités dans ce qui suit reposent sur des règles d'optimisation afin d'exécuter des tâches bien déterminées : suivi d'une cible, régression et couverture de l'espace.

Modèles pour le suivi de cible

Une majeur application des RCSF est le suivi de cible. Elle consiste à estimer instantanément la position d'une cible mobile. Elle est d'une grande importance en surveillance et sécurité, notamment dans les applications militaires. La mobilité contrôlée des capteurs permet d'améliorer l'estimation de position. Différentes techniques ont été proposées pour le suivi de cibles. Zou et Chakrabarty (2007) ont proposé un système de gestion de la mobilité basé sur la théorie de l'estimation bayésienne. Les nœuds peuvent se déplacer à une nouvelle position choisie dans un ensemble de positions candidates, et qui est à un pas de la position actuelle. Le mouvement est effectuée si la nouvelle position permettra d'améliorer la qualité de suivi. Dans un scénario différent, Chin et al. (2010) ont considéré le problème d'une cible mobile, appelée souris, essayant d'éviter la détection par des capteurs mobiles, appelés chats. Récemment, Mourad et al. (2012) ont proposé une nouvelle méthode pour le suivi de cible en utilisant l'algorithme de colonies de fourmis (Dorigo et Gambardella 1997). Ayant une cible en mouvement à chaque pas de temps, la technique consiste à estimer la position actuelle de la cible et ensuite prédire la position suivante à l'aide d'un modèle de prédiction du second ordre. Un repositionnement de capteurs est ensuite effectué afin d'optimiser la localisation de la cible pour le pas de temps suivant.

Modèles pour la régression et la couverture de l'espace

Un RCSF permet la surveillance et le suivi de l'évolution d'un champ de diffusion. Il peut s'agir de la diffusion d'un gaz, d'une espèce biochimique ou encore de la pollution, pour ne citer que ceux-ci. Cette application peut bénéficier de la mobilité des capteurs. Différentes techniques ont été proposées pour gérer cette mobilité. Ces techniques sont principalement portées sur l'amélioration de la topologie du réseau, ce qui améliore la couverture de la région sous surveillance. Parmi ces techniques, on retrouve la mobilité par balayage, la stratégie de Lambrou *et al.* et la stratégie de Lu *et al.*. Ces différentes techniques sont détaillées dans la suite.

Modèle de mobilité par balayage

Le modèle de mobilité par balayage (Prabhakaran et Sankar 2006) représente un ensemble de nœuds mobiles qui se déplacent dans une certaine direction fixe. Ce modèle de mobilité permet un balayage de tout l'espace. Lorsque le nœud mobile arrive à la frontière de la région sous surveillance, la direction du mouvement est retournée de 180 degrés en décrivant un arc de cercle. Ainsi, le nœud mobile se déplace-t-il dans le nouveau sens.

***Stratégie de Lambrou* et al.**

Lambrou et al. (2010) ont proposé une stratégie qui a pour but de couvrir tout l'espace étudié. A cette fin, N capteurs sont fixes et M capteurs sont mobiles, tous déployés dans un espace rectangulaire $\mathbb{X} \subset \mathbb{R}^2$ de taille $(X_1 \times X_2)$. En considérant une plage prédéterminée de la communication, deux nœuds sont connectés quand ils sont écartés de moins de δ unités de distance ; le voisinage d'un nœud i est alors défini par :

$$\mathcal{V}_i = \{l \ : \ \|\boldsymbol{x}_i - \boldsymbol{x}_l\| < \delta\}.$$

La région sous surveillance $\mathbb{X}$ est divisée en de plus petites régions régulières (cellules), de côté dx chacune. Ainsi, un capteur i occupe la cellule $c_i = (l, k)$, avec $l = \lceil \boldsymbol{x}_{i1}/dx \rceil$ et $k = \lceil \boldsymbol{x}_{i2}/dx \rceil$, où $\boldsymbol{x}_i = [\boldsymbol{x}_{i1} \quad \boldsymbol{x}_{i2}]$ et $\lceil \cdot \rceil$ désigne la partie entière par excès de son argument. Soit la matrice $\boldsymbol{G}$ de taille $(G_1 \times G_2)$, avec $G_1 = \lceil X_1/dx \rceil$ et $G_2 = \lceil X_2/dx \rceil$, et d'éléments $\boldsymbol{G}_{lk}$ tels que :

$$\boldsymbol{G}_{lk} = \begin{cases} 1, & \text{si la cellule } (l,k) \text{ est occupée par un capteur ;} \\ 0, & \text{sinon.} \end{cases}$$

Chaque élément de $\boldsymbol{G}$ représente la "confiance" qu'un événement se produisant dans la zone correspondante du champ sera détecté par le réseau de capteurs.

A chaque itération t, le nœud mobile i determine sa nouvelle position suivant la minimisation sur $\boldsymbol{x}$ d'une fonction coût de la forme :

$$J(\boldsymbol{x}) = \sum_j \omega_j J_j(\boldsymbol{x}),$$

où $J_j(\cdot)$ est une fonction coût particulière et les ω_j sont des poids non négatifs tels que $\sum_j \omega_j = 1$. Deux fonctions coût particulières sont utilisées selon $J(\boldsymbol{x}) = \omega_1 J_1(\boldsymbol{x}) + \omega_2 J_2(\boldsymbol{x})$ avec :

- $J_1(\cdot)$ qui pénalise les positions qui sont loins des grands trous de couverture, selon
$$J_1(\boldsymbol{x}) = \frac{\|\boldsymbol{x} - \boldsymbol{t}_i\|}{\delta},$$
où $\boldsymbol{t}_i$ désigne la position du centre du plus grand trou de couverture dans le voisinage $\mathcal{V}_i$. Cette position est déterminée par l'algorithme "zoom" (Lambrou et Panayiotou 2007) en cherchant la zone appartenant au $\mathcal{V}_i$ et dont la somme des éléments de $\boldsymbol{G}$ correspondants est minimale.
- $J_2(\cdot)$ qui pénalise les positions qui sont proches de capteurs statiques, selon
$$J_2(\boldsymbol{x}) = \max_{j \in \mathcal{V}_i} \exp\Big(- \frac{\|\boldsymbol{x} - \boldsymbol{x}_j\|^2}{r_s^2} \Big),$$
où r_s désigne la plage de detection des capteurs. Cette fonction se comporte comme une force de répulsion locale. Elle repousse le nœud mobile de son capteur voisin le plus proche.

Supposons qu'à l'itération t, le capteur mobile à la position $\boldsymbol{x}_{i,t-1}$ se dirige selon une direction θ. Les prochaines positions possibles sont les R points $(\boldsymbol{x}^{(1)}, \cdots, \boldsymbol{x}^{(R)})$, uniformément réparties sur l'arc de centre $\boldsymbol{x}_{i,t-1}$, de

rayon d et d'angle au centre entre $\theta - \phi$ et $\theta + \phi$. Le nœud mobile évalue la fonction coût $J(\boldsymbol{x}^{(l)})$ pour les R positions candidates et se déplace vers la position qui la minimise.

***Stratégie de Lu* et al.**

Les méthodes précédentes sont indépendantes du problème d'estimation en question, mais reposent uniquement sur de la topologie des capteurs. Lu et al. (2010) proposent une stratégie de mobilité basée sur l'estimation du champ étudié. L'algorithme proposé comprend deux étapes majeures à chaque itération :

- La première étape consiste à déterminer la fonction de régression $\psi(\cdot)$ qui estime le champ de diffusion dans la région sous surveillance $\mathbb{X}$.
- La deuxième étape consiste à mettre en œuvre l'optimisation de la localisation à l'aide de l'algorithme de Lloyd.

La suite de cette section présente l'algorithme de Lloyd distribué pour le contrôle de la couverture basée sur la fonction estimée du champ de diffusion.

A chaque capteur i est définie une région de Voronoï V_i qui est l'ensemble des points les plus proches de i que de tout autre point de $\mathbb{X}$. En d'autre terme, la région de Voronoï V_i représente en quelque sorte la "zone d'influence" du capteur, avec

$$V_i = \{\boldsymbol{x} \in \mathbb{X} \ : \ \|\boldsymbol{x} - \boldsymbol{x}_i\| \leq \|\boldsymbol{x} - \boldsymbol{x}_j\| \quad \forall j = 1, 2, \cdots, N, \ \text{ et } j \neq i\}.$$

L'algorithme consiste à déplacer le capteur mobile i vers le centre de masse de la région de Voronoï. Donc, à l'itération t, le capteur i se déplace de $\boldsymbol{x}_{i,t-1}$ à $\boldsymbol{x}_{i,t}$ selon :

$$\boldsymbol{x}_{i,t} = \frac{L_{V_i}}{M_{V_i}}$$

avec

$$M_{V_i} = \int_{V_i} \psi(\boldsymbol{x}) d\boldsymbol{x}$$

et

$$L_{V_i} = \int_{V_i} \boldsymbol{x}\psi(\boldsymbol{x}) d\boldsymbol{x}$$

Dans le cas où la communication entre les capteurs est limitée sur le voisinage, la région de Voronoï V_i du capteur i doit se réduire à :

$$W_i = V_i \cap \mathcal{V}_i.$$

En effet, le calcul de V_i nécessite l'évaluation de ψ dans le voisinage du capteur i, c'est-à-dire $\mathcal{V}_i$. Cependant, les capteurs ont accès seulement aux information fournis dans $\mathcal{V}_i$. D'où la réduction de V_i à W_i.

Plan de cette partie

La première partie de ce manuscrit est formée de deux chapitres. Dans le premier chapitre, nous donnons un bref aperçu de l'apprentissage statistique et des méthodes à noyaux. Nous introduisons la notion du noyau

reproduisant et ses caractéristiques. Ensuite nous présentons des théorèmes bien connus qui sont étudiés dans cette partie. Nous présentons deux exemples clés sur les méthodes à noyaux qui sont la régression par moindres carrés à noyaux et l'analyse en composantes principales à noyaux. Dans le second chapitre, nous traitons le problème de modélisation de la distribution d'un champ. A cet effet, nous proposons un modèle basé sur les noyaux que nous mettons à jour à l'aide des informations provenant des capteurs mobiles. Nous montrons que le cadre proposé permet de tirer un régime efficace pour la mobilité des capteurs, en minimisant l'erreur d'approximation. Il s'avère que le problème résultant est lié à un problème bien connu en apprentissage statistique qui n'est autre que le problème de pré-image (Honeine et Richard 2011a;b). Nous profitons de l'évolution récente dans ce domaine pour proposer des schémas d'optimisation afin de résoudre le problème de la mobilité.

Méthodes à noyaux

1

Sommaire

L'identification de système a toujours été un problème difficile en traitement du signal et en apprentissage statistique. L'estimation fonctionnelle basée sur l'espace de Hilbert à noyau reproduisant fournit des méthodes efficaces de régression non linéaire comme extensions de celles linéaires, en remplaçant dans les algorithmes le produit scalaire par un noyau reproduisant (Aronszajn 1950). Cette astuce permet de cartographier implicitement les données dans un espace caractéristique non linéaire, où le modèle original est appliqué (Shawe-Taylor et Cristianini 2004).

En méthodes à noyaux pour l'apprentissage statistique, le modèle sous-jacent est défini par une combinaison linéaire de fonctions noyau centrées sur les échantillons d'apprentissage. Telle est l'essence du Théorème de Représentation (Schölkopf et al. 2001). Il énonce clairement les différents défis qui ont émergé depuis la fin des années 90 en apprentissage statistique à base de noyaux : d'abord, les chercheurs se sont intéressés à l'estimation des coefficients de la combinaison linéaire pour un critère d'optimisation tel que dans les Machines à Vecteurs de Support (SVM) et la régression par moindres carrés à noyaux (Schölkopf et Smola 2001, Shawe-Taylor et Cristianini 2004). Ensuite, un effort important a été porté sur les noyaux, aussi bien sur la mise en œuvre de nouvelles classes de noyaux pour de "nouveaux types" de données telles que les graphes et les documents, que sur le choix d'un noyau approprié et son réglage optimal pour une tâche donnée, notamment avec l'apprentissage par noyaux multiples (Gönen et Alpaydin 2011). En outre, le domaine de représentations parcimonieuses est apparue avec des modèles ayant moins de termes d'expansion, d'abord étudiés dans le cadre des SVM et plus récemment par pénalisation ou avec des critères de parcimonies (Honeine 2012). Ce

dilemme est lié à la sélection d'un sous-ensemble d'apprentissage actif où un critère "statistiquement optimal" est étudié dans (MacKay 1992, Hoi et Jin 2008).

Ce chapitre couvre tout d'abord la notion des noyaux reproduisants. Ensuite, les caractéristiques de tels noyaux sont données. Après, nous introduisons les deux éléments fondamentaux des méthodes à noyaux qui sont le Théorème de Représentation et l'astuce du noyau. Nous concluons ce chapitre par deux principaux exemples de méthodes à noyaux, avec la régression par moindres carrés à noyaux et l'analyse en composantes principales à noyaux, qui seront réexaminés tout au long du manuscrit dans le cadre des RCSF.

1.1 Apprentissage statistique

En théorie de l'apprentissage statistique (Vapnik 1995), on cherche à déterminer un modèle qui traduit le mieux possible une relation entre les observations recueillies sur un système, ou encore entre ses entrées et sorties, à partir d'un ensemble d'apprentissage. Les performances du modèle résultant devraient se généraliser à de nouvelles observations. Afin de réaliser cette condition, une certaine connaissance a priori du comportement du système doit être incorporée grâce à un choix approprié de l'espace d'hypothèses duquel la solution est recherchée. Les méthodes en apprentissage statistique peuvent être regroupées en deux classes principales : supervisées et non supervisées (Hastie et al. 2003).

En apprentissage supervisé, on cherche une fonction $\psi(\cdot)$ qui détermine la relation entre l'espace des échantillons ou d'observations $\mathbb{X} \in \mathbb{R}^d$ et l'espace de réponse $\mathbb{Y} \in \mathbb{R}$. La fonction optimale sur toutes les fonctions est donnée par la minimisation d'une fonctionnelle de risque réelle, selon

$$\arg\min_{\psi} \int_{\mathbb{X}\times\mathbb{Y}} L(\psi(\boldsymbol{x}), y) P(\boldsymbol{x}, y) d\boldsymbol{x} dy,$$

où L est la fonction coût qui mesure l'erreur commise entre la sortie désirée y et la sortie estimée $\psi(\boldsymbol{x})$, et $P(\boldsymbol{x}, y)$ est la distribution de probabilité définie pour tout couple $(\boldsymbol{x}, y) \in \mathbb{X} \times \mathbb{Y}$. Toutefois, cette distribution $P(\boldsymbol{x}, y)$ est inconnue. Pour surmonter cet inconvénient, la fonction optimale est souvent obtenue à partir d'un ensemble d'apprentissage, qui est un ensemble fini de N réalisations formé de $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_N, y_N)\} \subset \mathbb{X} \times \mathbb{Y}$. Ainsi, aboutissons-nous à la minimisation du risque d'apprentissage, ou risque empirique, selon

$$\arg\min_{\psi} \frac{1}{N} \sum_{i=1}^{N} L(\psi(\boldsymbol{x}_i), y_i). \tag{1.1}$$

A noter que lorsque les y_i prennent des valeurs discrètes, il s'agit d'un problème de classification, par exemple en classification binaire où on cherche à attribuer à chaque échantillon une étiquette $+1$ ou -1, tandis que lorsque les y_i prennent des valeurs réelles, il s'agit d'un problème de régression.

En apprentissage non supervisé, les seules observations disponibles sont les échantillons de l'espace d'entrée $\mathbb{X}$; les échantillons ne sont pas

associées à des étiquettes. On cherche alors la relation entre les éléments de l'espace $\mathbb{X}$ sans les informations données par les étiquettes comme dans le cas de l'apprentissage supervisé. Le problème d'optimisation dans ce cas est le suivant :

$$\arg\min_{\psi} \int_{\mathbb{X}} L(\psi(\boldsymbol{x}))P(\boldsymbol{x})d\boldsymbol{x},$$

où $P(\boldsymbol{x})$ est la distribution de probabilité de $\boldsymbol{x} \in \mathbb{X}$, et L une fonction coût donnée. Comme dans le cas de l'apprentissage supervisé, la distribution $P(\boldsymbol{x})$ est inconnue. Pour surmonter cette difficulté, la fonction optimale est donnée par un ensemble d'apprentissage, qui est un ensemble fini de N réalisations formé de $\{\boldsymbol{x}_1, \boldsymbol{x}_2, , \cdots, \boldsymbol{x}_N\} \subset \mathbb{X}$ en minimisant le risque empirique, selon

$$\arg\min_{\psi} \frac{1}{n} \sum_{i=1}^{n} L(\psi(\boldsymbol{x}_i)). \tag{1.2}$$

Une infinité de solutions minimisent le risque empirique dans les deux cas. En d'autres termes, il existe une infinité de solutions qui vérifient (1.1) pour l'apprentissage supervisé ou (1.2) pour l'apprentissage non supervisé. Il s'agit alors d'un problème mal-posé puisqu'il n'est pas possible de reconstituer d'une manière unique la fonction recherchée à partir de l'ensemble d'apprentissage. Pour surmonter ce problème, on a recours à la régularisation de Tikhonov (Tikhonov et Arsenin 1977a), en restreignant l'espace de fonctions candidates $\mathcal{H}$ à un espace de fonctions régulières. Soient $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ le produit scalaire dans $\mathcal{H}$ et $\| \cdot \|_{\mathcal{H}}$ la norme correspondante. Le problème d'optimisation avec une fonctionnelle de risque régularisée à la Tikhonov est de la forme :

$$\psi^* = \arg\min_{\psi} \frac{1}{N} \sum_{i=1}^{N} L(\psi(\boldsymbol{x}_i), y_i) + \nu \|\psi\|^2_{\mathcal{H}}. \tag{1.3}$$

où ν contrôle le compromis entre, d'une part le premier terme qui représente le risque empirique mesurant l'adéquation entre les sorties estimées et les sorties désirées et, d'autre part le second terme qui représente la pénalisation permettant d'obtenir des solutions plus régulières. Sans la pénalisation, le problème d'optimisation serait mal-posé puisqu'il existerait alors une infinité de fonctions qui minimisent le premier terme. La théorie de la régularisation à la Tikhonov des méthodes d'apprentissage statistiques a connu diverses avancées, notamment avec les travaux de Poggio et Girosi (1989) et Vapnik (1995). Une généralisation de cette pénalisation a également été considérée en remplaçant le terme $\|\psi\|^2_{\mathcal{H}}$ par $\mathcal{R}(\|\psi\|^2_{\mathcal{H}})$, où $\mathcal{R}(\cdot)$ est une fonction monotone croissante sur $\mathbb{R}^+$. Le problème d'optimisation s'écrit alors sous la forme

$$\psi^* = \arg\min_{\psi} \frac{1}{N} \sum_{i=1}^{N} L(\psi(\boldsymbol{x}_i), y_i) + \nu \mathcal{R}(\|\psi\|^2_{\mathcal{H}}). \tag{1.4}$$

Un choix particulier de l'espace fonctionnel $\mathcal{H}$ est l'espace de Hilbert à noyau reproduisant (Aronszajn 1950).

1.2 Noyau reproduisant et espace de Hilbert associé

Les méthodes à noyaux assurent le passage des données de l'espace des observations à l'espace dit de Hilbert à noyau reproduisant (RKHS

pour *Reproducing Kernel Hilbert Space*). Ainsi, pour découvrir des relations cachées existantes dans les échantillons, des algorithmes linéaires sont-ils appliqués dans le RKHS (Shawe-Taylor et Cristianini 2004, Vert et al. 2004).

1.2.1 Noyau défini positif

Un noyau est une fonction de similarité utilisée pour comparer deux échantillons qui appartiennent au même domaine d'entrée $\mathbb{X}$:

Définition 1.1 (Noyau) *Un noyau désigne une fonction symétrique de* $\mathbb{X} \times \mathbb{X}$ *dans* $\mathbb{R}$, *c'est-à-dire* $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \kappa(\boldsymbol{x}_j, \boldsymbol{x}_i)$.

Définition 1.2 (Noyau défini positif) *Un noyau est dit défini positif si pour tout entier fini* N :

$$\sum_{i=1}^{N}\sum_{j=1}^{N} c_i c_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0, \quad \forall (\boldsymbol{x}_i, c_i) \in \mathbb{X} \times \mathbb{R}, i \in \{1, 2, \cdots, N\}.$$

Définition 1.3 (Matrice de Gram) *Etant donné un noyau* $\kappa(\cdot, \cdot)$ *et* N *observations* $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N$, *la matrice de Gram* $\boldsymbol{K}$ *de taille* $(N \times N)$ *est définie de la façon suivante :*

$$\boldsymbol{K}_{ij} = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j), \quad \forall (\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathbb{X} \times \mathbb{X}, 1 \leq i, j \leq N$$

Cette matrice est symétrique, puisque, par définition, nous avons $\kappa(x_i, x_j) = \kappa(x_i, x_j)$ pour tout $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{X}$. En outre, elle est définie positive, car

$$\boldsymbol{c}^\top \boldsymbol{K} \boldsymbol{c} \geq 0, \quad \forall \boldsymbol{c} \in \mathbb{R}^N.$$

1.2.2 Cadre fonctionnel

Nous présentons les espaces de Hilbert à noyau reproduisant comme espaces d'hypothèses en vue de faire de l'estimation de fonction. Pour cela, nous commençons par définir tout d'abord les notions d'espace de Hilbert et de noyau reproduisant.

Définition 1.4 (Espace de Hilbert) *Un espace de Hilbert* $\mathcal{H}$ *est un espace vectoriel complet, muni d'un produit scalaire* $\langle \cdot, \cdot \rangle_{\mathcal{H}}$, *dont on tire une norme.*

On dit que $\mathcal{H}$ est un espace complet si toute suite de Cauchy $\{h_n\}_{n \geq 1}$ de $\mathcal{H}$ a une limite dans $\mathcal{H}$. Une suite de Cauchy satisfait :

$$\sup_{m>n} \|h_n - h_m\| \to 0, \quad \text{lorsque} \quad n \to \infty.$$

Définition 1.5 (Noyau reproduisant) *Soit* $\mathcal{H}$ *un espace de Hilbert de fonctions de* $\mathbb{X}$ *dans* $\mathbb{R}$. *La fonction* $\kappa : \mathbb{X} \times \mathbb{X} \longrightarrow \mathbb{R}$ *est appelée noyau reproduisant de* $\mathcal{H}$ *si :*

1. *Pour tout élément* $\boldsymbol{x} \in \mathbb{X}$, *la fonction* $\kappa(\boldsymbol{x}, \cdot) : \boldsymbol{x}_j \mapsto \kappa(\boldsymbol{x}, \boldsymbol{x}_j)$ *appartient à* $\mathcal{H}$.
2. *Pour tout* $\boldsymbol{x} \in \mathbb{X}$, *et* $\psi \in \mathcal{H}$, *la propriété reproduisante est vérifiée :*

$$\psi(\boldsymbol{x}) = \langle \psi, \kappa(\boldsymbol{x}, \cdot) \rangle_{\mathcal{H}}.$$

En particulier, pour toute paire $(\boldsymbol{x}, \boldsymbol{x}_j) \in \mathbb{X} \times \mathbb{X}, \langle \kappa(\boldsymbol{x}, \cdot), \kappa(\boldsymbol{x}_j, \cdot) \rangle_{\mathcal{H}} = \kappa(\boldsymbol{x}, \boldsymbol{x}_j)$. Ainsi l'évaluation du noyau en deux échantillons de $\mathbb{X}$ peut-elle s'écrire sous la forme d'un produit scalaire dans $\mathcal{H}$. L'espace $\mathcal{H}$ est dit espace de Hilbert à noyau reproduisant si un noyau reproduisant κ existe.

1.3 Astuce du noyau

Tout noyau défini positif $\kappa(\cdot,\cdot)$ peut s'écrire comme un produit scalaire dans un espace de Hilbert.

Théorème 1.1 (Astuce du noyau) *Un noyau κ est défini positif si et seulement si il existe un espace de Hilbert $\mathcal{H}$ muni d'un produit scalaire $\langle\cdot,\cdot\rangle_{\mathcal{H}}$ et une fonction $\phi : \mathbb{X} \longrightarrow \mathcal{H}$ tels que :*

$$\forall(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \mathbb{X} \times \mathbb{X}, \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle\phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j)\rangle_{\mathcal{H}}.$$

On peut définir la transformation ϕ par l'intermédiaire du noyau κ :

$$\begin{aligned}\phi : \mathbb{X} &\longrightarrow \mathcal{H}\\ \boldsymbol{x} &\longmapsto \phi(\boldsymbol{x}) = k(\boldsymbol{x}, \cdot).\end{aligned}$$

Afin de démontrer ce théorème, on considère au début un espace vectoriel, avec $\mathbb{X} = \mathbb{R}^d$, $d \in \mathbb{N}^+$, où deux vecteurs peuvent être comparés par leur produit scalaire conventionnel $\langle \boldsymbol{x}_i, \boldsymbol{x}_j\rangle$. On peut facilement voir que ce produit scalaire entre les vecteurs est un noyau. En effet, il est symétrique puisque $\langle \boldsymbol{x}_i, \boldsymbol{x}_j\rangle = \langle \boldsymbol{x}_j, \boldsymbol{x}_i\rangle$, et il est défini positif puisque, pour tout N :

$$\sum_{i=1}^{N}\sum_{j=1}^{N} c_i c_j \langle \boldsymbol{x}_i, \boldsymbol{x}_j\rangle = \left\|\sum_{i=1}^{N} c_i \boldsymbol{x}_i\right\|^2 \geq 0, \quad \forall(\boldsymbol{x}_i, c_i) \in \mathbb{X} \times \mathbb{R}, i \in \{1, 2, \cdots, N\}. \tag{1.5}$$

Ce produit scalaire est le *noyau linéaire*. Il est limité par le fait qu'il ne peut être utilisé que pour analyser les vecteurs. En général, $\mathbb{X}$ n'est pas nécessairement un espace vectoriel. Nous avons alors recours à représenter chaque observation $\boldsymbol{x} \in \mathbb{X}$ avec $\phi(\boldsymbol{x})$ où ϕ est une fonction de $\mathbb{X}$ dans $\mathcal{H}$, et définir un noyau pour tout $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{X}$ par :

$$\kappa(x_i, x_j) = \langle\phi(x_i), \phi(x_j)\rangle_{\mathcal{H}}, \tag{1.6}$$

En procédant comme dans (1.5), on peut facilement vérifier que κ est un noyau valide sur l'espace $\mathbb{X}$, qui n'est pas nécessairement un espace vectoriel. L'avantage ici est que le produit scalaire est calculé directement à partir des données d'entrée en utilisant la fonction du noyau κ, donc sans connaissance explicite de la fonction ϕ. On peut alors résoudre un problème non linéaire en transformant l'espace de représentation des données d'entrée en un espace de plus grande dimension, appelé *espace caractéristique, espace de description ou encore espace d'attributs*, et en utilisant un modèle linéaire dans cet espace. Par conséquent, on peut effectuer un algorithme qui utilise uniquement le produit scalaire entre les vecteurs d'entrée implicitement dans un espace de Hilbert, en remplaçant chaque produit scalaire par l'évaluation de la fonction noyau. L'astuce du noyau permet ainsi d'opérer dans un espace de plus grande dimension sans avoir à calculer explicitement les coordonnées des données dans cet espace. La Figure 1.1 donne une illustration de ce principe pour un problème de classification binaire avec deux classes qui ne sont pas linéairement séparables.

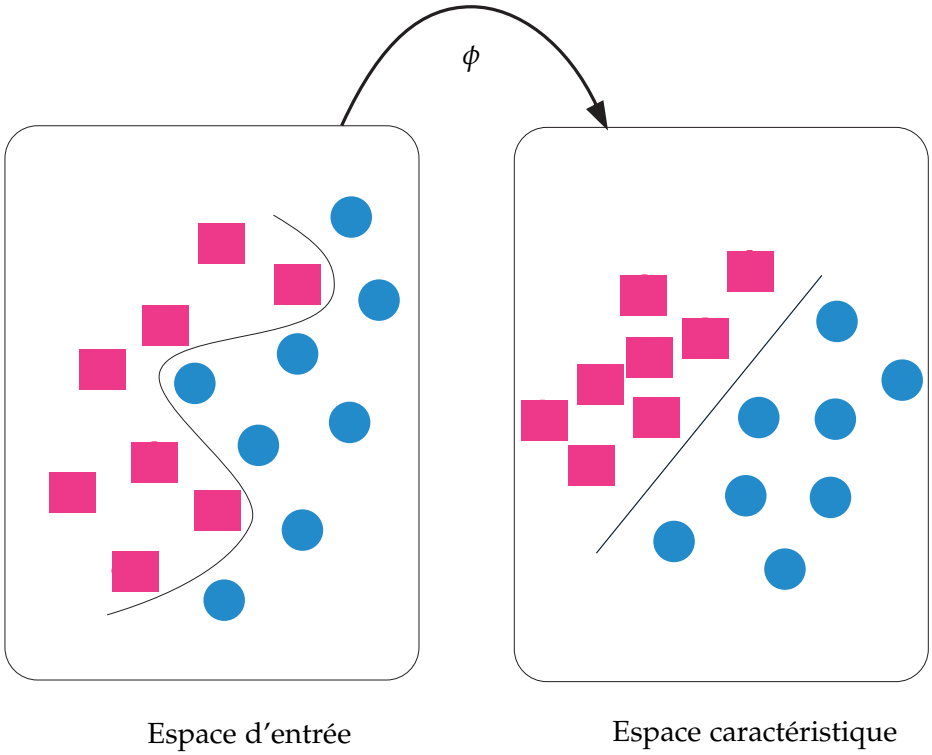

FIGURE 1.1 – *Passage des données de l'espace d'entrée vers un espace caractéristique où elles sont linéairement séparables.*

1.3.1 Théorème de Moore-Aronszajn

Le théorème de Moore-Aronszajn (Aronszajn 1950) permet d'établir le lien entre les noyaux définis positifs et les espaces de Hilbert à noyau reproduisant.

Théorème 1.2 (Théorème de Moore-Aronszajn) *A tout noyau défini positif $\kappa : \mathbb{X} \times \mathbb{X} \longrightarrow \mathbb{R}$, il correspond un RKHS unique admettant κ comme noyau reproduisant, et réciproquement.*

Démonstration. Soit κ un noyau défini positif. On définit l'espace $\mathcal{H}_0$ construit à partir de l'ensemble des fonctions de la forme :

$$f(\cdot) = \sum_{i=1}^{N} \alpha_i \phi(\boldsymbol{x}_i).$$

avec $N \in \mathbb{N}^+$, $\alpha_i \in \mathbb{R}$, $\boldsymbol{x}_i \in \mathbb{X}$ et ϕ la transformation non linéaire de l'espace des observations $\mathbb{X}$ à l'espace fonctionnel $\mathcal{H}_0$. On considère deux fonctions quelconque de $\mathcal{H}_0$:

$$f(\cdot) = \sum_{i=1}^{N} \alpha_i \phi(\boldsymbol{x}_i) \qquad et \qquad g(\cdot) = \sum_{j=1}^{N} \beta_j \phi(\boldsymbol{x}_j).$$

Le produit scalaire dans $\mathcal{H}_0$ entre ces fonctions est donné par :

$$\langle f, g \rangle_{\mathcal{H}_0} = \langle \sum_{i=1}^{N} \alpha_i \phi(\boldsymbol{x}_i), \sum_{j=1}^{N} \beta_j \phi(\boldsymbol{x}_j) \rangle_{\mathcal{H}}.$$

En utilisant l'astuce du noyau, l'expression du produit scalaire devient :

$$\langle f, g \rangle_{\mathcal{H}_0} = \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \beta_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j).$$

Muni de ce produit scalaire, l'espace ainsi construit est un espace pré-Hilbertien. L'espace de Hilbert admettant κ comme noyau reproduisant est alors obtenu en complétant $\mathcal{H}_0$ conformément à (Aronszajn 1950) de sorte que toute suite de Cauchy y converge.

Réciproquement, pour démontrer que tout noyau reproduisant est défini positif, il suffit de constater que $\sum_i \sum_j \alpha_i \alpha_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \| \sum_i \alpha_i \phi(\boldsymbol{x}_i) \|^2$ ne peut être négatif. □

La relation associant l'espace de Hilbert à noyau reproduisant à un noyau donné est décrite par le biais du théorème de Moore-Aronszajn. La Table 1.1 montre les noyaux définis positifs les plus courants dans la littérature.

Table 1.1 – *Les noyaux définis positifs les plus courants, avec les paramètres* $b \geq 0$, $c > 0$, $\sigma \in \mathbb{R}^+$ *et* $p \in \mathbb{N}^+$.

noyau	**Expression de** $k(\boldsymbol{x}_i, \boldsymbol{x}_j)$
Noyau linéaire	$\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$
Noyau polynomial	$\left(\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle + b \right)^p$
Noyau Gaussian	$\exp\left(- \frac{\left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|^2}{2\sigma^2} \right)$
Noyau Laplacian	$\exp\left(- \frac{\left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|}{c} \right)$
Noyau quadratique rationnel	$1 - \frac{\left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|^2}{\left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|^2 + c}$

1.4 Théorème de Représentation

L'objectif en apprentissage statistique est de déterminer une fonction qui caractérise la relation entre les données disponibles du système étudié. En reprenant le problème de la minimisation du risque empirique régularisé (1.4) en se plaçant dans le cadre d'un RKHS, la solution optimale admet une forme explicite donnée par le Théorème de Représentation. Ce théorème a été introduit dans les problèmes d'interpolation par spline par Kimeldorf et Wahba dans (Kimeldorf et Wahba 1971). Il a été récemment généralisé à d'autres problèmes d'apprentissage dans (Schölkopf et al. 2000, Cucker et Smale 2002, Kurkova et Sanguineti 2005). Selon ce théorème, les solutions d'une grande classe de problèmes d'optimisation régularisées peuvent être exprimées comme une combinaison linéaire finie de fonctions noyau évaluées sur les échantillons de l'ensemble d'apprentissage.

Théorème 1.3 (Théorème de Représentation) *Soient* $\kappa : \mathbb{X} \times \mathbb{X} \longrightarrow \mathbb{R}$ *un noyau reproduisant,* $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_N, y_N)\} \subset \mathbb{X} \times \mathbb{Y}$ *un ensemble d'apprentissage et* L

une fonction coût quelconque. Soit $\mathcal{H}$ l'espace de Hilbert induit par le noyau reproduisant κ et $\mathcal{R}(\cdot)$ une fonction strictement croissante sur $\mathbb{R}^+$. Alors toute fonction $\psi^* \in \mathcal{H}$ minimisant le risque empirique régularisé :

$$\frac{1}{N}\sum_{i=1}^{N} L(\psi(\boldsymbol{x}_i), y_i) + \nu\mathcal{R}(\|\psi\|^2_{\mathcal{H}}),$$

peut s'écrire sous la forme :

$$\psi^*(\cdot) = \sum_{i=1}^{N} \alpha_i \kappa(\boldsymbol{x}_i, \cdot). \tag{1.7}$$

Démonstration. Soit $\mathcal{H}_n$ le sous-espace de $\mathcal{H}$ engendré par les fonctions $\{\kappa(\boldsymbol{x}_1, \cdot), \cdots, \kappa(\boldsymbol{x}_n, \cdot)\}$, c'est-à-dire :

$$\mathcal{H}_n = \{\psi \in \mathcal{H}; \psi(\cdot) = \sum_{i=1}^{N} \alpha_i \kappa(\boldsymbol{x}_i, \cdot), \alpha_1, \cdots, \alpha_N \in \mathbb{R}\}.$$

Toute fonction ψ de $\mathcal{H}$ admet une et une seule décomposition en deux contributions, l'une appartenant à l'espace $\mathcal{H}_n$ et l'autre qui lui est orthogonale :

$$\psi(\cdot) = \psi^*(\cdot) + \psi^{\perp}(\cdot),$$

avec $\psi^* \in \mathcal{H}_n$ et $\psi^{\perp}$ la composante orthogonale telle que $\langle \psi(\cdot)^{\perp}, \kappa(x_i, \cdot)\rangle_{\mathcal{H}} = 0$, $\forall i \in \{1, 2, \cdots, N\}$. La propriété reproduisante permet d'évaluer ψ en tout $\boldsymbol{x}_j$, $j \in \{1, 2, \cdots, N\}$, selon l'expression :

$$\begin{aligned}
\psi(x_j) &= \langle \psi(\cdot), \kappa(x_j, \cdot)\rangle_{\mathcal{H}} \\
&= \sum_{i=1}^{N} \alpha_i \, \kappa(x_i, x_j) + \langle \psi^{\perp}(\cdot), \kappa(x_j, \cdot)\rangle_{\mathcal{H}} \\
&= \sum_{i=1}^{N} \alpha_i \, \kappa(x_i, x_j).
\end{aligned}$$

L'évaluation de ψ en chaque élément de l'ensemble d'apprentissage ne dépend que des coefficients α_i, $i = 1, 2, \cdots, N$. Ainsi la minimisation du risque empirique (1.4) est-elle indépendante de la composante orthogonale $\psi^{\perp}$. En outre, puisque $\psi^{\perp}(\cdot)$ est orthogonale à $\sum_{i=1}^{N} \alpha_i \, \kappa(x_i, \cdot)$, et $\mathcal{R}$ est une fonction strictement croissante, le théorème de Pythagore donne :

$$\begin{aligned}
\mathcal{R}(\|\psi\|^2_{\mathcal{H}}) &= \mathcal{R}\Big(\|\sum_{i=1}^{N} \alpha_i \, \kappa(x_i, \cdot)\|^2_{\mathcal{H}} + \|\psi^{\perp}(\cdot)\|^2_{\mathcal{H}}\Big) \\
&\geq \mathcal{R}\Big(\|\sum_{i=1}^{N} \alpha_i \, \kappa(x_i, \cdot)\|^2_{\mathcal{H}}\Big).
\end{aligned}$$

La fonction qui minimise l'expression ci-dessus doit donc vérifier $\|\psi^{\perp}(\cdot)\|^2_{\mathcal{H}} = 0$. Ceci conclut la preuve du théorème. □

Le Théorème de Représentation établit ainsi que la solution minimisant le risque empirique régularisé est une combinaison linéaire de la fonction noyau appliquée sur les données d'apprentissage. On peut convertir alors le problème de recherche dans un espace $\mathcal{H}$ à un problème d'estimation de N coefficients α_i.

1.5 Régression par moindres carrés à noyaux

La régression dite Ridge (Hoerl et Kennard 1970, Tikhonov et Arsenin 1977b) est une méthode de régression à modèle linéaire et une fonction coût quadratique. Nous présentons dans ce paragraphe la méthode des moindres carrés à noyaux (ou *Kernel Ridge Regression* en anglais) qui permet de définir des modèles de régression non linéaires en se plaçant dans le cadre des méthodes à noyaux.

Nous considérons l'ensemble d'apprentissage avec des couples $(\boldsymbol{x}_i, y_i)$, $i \in \{1, 2, \cdots, N\}$. Nous cherchons à déterminer une fonction $\psi^*(\cdot)$ qui minimise l'erreur quadratique moyenne entre les sorties estimées $\psi(\boldsymbol{x}_i)$ et les sorties désirées y_i, en remplaçant la fonction coût L du risque régularisé (1.4) par la perte quadratique $(y_i - \psi(\boldsymbol{x}_i))^2$ et le terme de régularisation $\mathcal{R}(\|\psi\|^2_{\mathcal{H}})$ par sa forme la plus simple $\|\psi\|^2_{\mathcal{H}}$. Nous pouvons alors écrire :

$$\psi^* = \arg\min_{\psi \in \mathcal{H}} \frac{1}{N} \sum_{i=1}^{N} (y_i - \psi(\boldsymbol{x}_i))^2 + \nu \|\psi\|^2_{\mathcal{H}}, \tag{1.8}$$

où ν est un paramètre de régularisation qui contrôle le compromis entre la justesse aux données disponibles et la douceur de la solution.

La solution de ce problème est donnée par le Théorème de Représentation sous la forme d'une combinaison linéaire des $\kappa(\boldsymbol{x}_j, \cdot)$, pour $j \in \{1, 2, \cdots, N\}$, selon

$$\psi^*(\cdot) = \sum_{j=1}^{N} \alpha_j \kappa(\boldsymbol{x}_j, \cdot). \tag{1.9}$$

Puisque le modèle est linéaire avec les coefficients α_j, ces derniers peuvent être facilement estimés comme suit. Soit le vecteur $\boldsymbol{\alpha}$ qui regroupe les coefficients inconnus α_j pour $j \in \{1, 2, \cdots, N\}$. En combinant les équations (1.8) et (1.9), nous obtenons :

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{y} - \boldsymbol{K}\boldsymbol{\alpha}\|^2 + \nu \boldsymbol{\alpha}^T \boldsymbol{K} \boldsymbol{\alpha}, \tag{1.10}$$

où $\boldsymbol{K}$ est la matrice de Gram dont les éléments sont $\kappa(\boldsymbol{x}_j, \boldsymbol{x}_l)$ pour $j, l \in \{1, 2, \cdots, N\}$, et $\boldsymbol{y}$ est le vecteur regroupant les y_j pour $j \in \{1, 2, \cdots, N\}$. La solution de ce problème d'optimisation est donnée grâce à la méthode des moindres carrés, en annulant le gradient par rapport à $\boldsymbol{\alpha}$ de la fonction coût dans (1.10). Soit $F(\boldsymbol{\alpha}) = \|\boldsymbol{y} - \boldsymbol{K}\boldsymbol{\alpha}\|^2 + \nu \boldsymbol{\alpha}^T \boldsymbol{K} \boldsymbol{\alpha}$ cette fonction coût, alors

$$\nabla_{\boldsymbol{\alpha}} F(\boldsymbol{\alpha}) = 2\boldsymbol{K}^\top (\boldsymbol{K}\boldsymbol{\alpha} - \boldsymbol{y}) - 2\nu \boldsymbol{K}\boldsymbol{\alpha}.$$

La matrice de Gram $\boldsymbol{K}$ étant symétrique définie positive, alors $\nabla_{\boldsymbol{\alpha}} F(\boldsymbol{\alpha}) = 0$ est une condition nécessaire et suffisante pour le calcul de $\boldsymbol{\alpha}$ optimal. La solution optimale est obtenue par la résolution du système linéaire suivant :

$$(\boldsymbol{K} + \nu \mathbf{I})\, \boldsymbol{\alpha}^* = \boldsymbol{y},$$

où $\mathbf{I}$ est la matrice identité de taille $(N \times N)$. Notons que pour une valeur appropriée du paramètre de régularisation ν, la matrice entre parenthèses est toujours non singulière. En conséquence, le vecteur de solution optimale $\boldsymbol{\alpha}^*$ est obtenu par :

$$\boldsymbol{\alpha}^* = (\boldsymbol{K} + \nu \mathbf{I})^{-1} \boldsymbol{y}. \tag{1.11}$$

1.6 Analyse en composantes principales à noyaux

L'analyse en composantes principales (ACP), étudiée en détail dans la deuxième partie de cette thèse, est un outil mathématique puissant pour extraire des informations pertinentes à partir de données multivariées. Cette approche ne tient compte d'aucune connaissance préalable du système, à l'exception de sa linéarité. Elle consiste à déterminer un sous-espace qui retient la plus grande variance des données. Ce sous-espace est engendré par des axes dits axes principaux. On peut montrer que les axes principaux sont les vecteurs propres associés aux plus grandes valeurs propres de la matrice de covariance $\mathfrak{C}$ des échantillons $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\}$, avec $\mathfrak{C} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i \boldsymbol{x}_i^\top$ où les échantillons sont supposés de moyenne nulle.

L'analyse en composantes principales à noyaux (Schölkopf et al. 1999, Kallas et al. 2010) est une extension de l'ACP en utilisant des techniques de méthodes à noyaux. Ainsi, les opérations linéaires de l'ACP sont-elles réalisées dans un espace de Hilbert à noyau reproduisant. Bien que les vecteurs propres résultants soient obtenus par une technique linéaire dans l'espace fonctionnel, ils décrivent des relations non linéaires dans l'espace des observations. L'astuce du noyau permet de ne pas calculer explicitement la fonction de transformation non linéaire.

Soit ϕ une transformation non linéaire de l'espace des observations $\mathbb{X}$ à l'espace de Hilbert à noyau reproduisant $\mathcal{H}$ qui fait correspondre à chaque $\boldsymbol{x}_i$ son image $\phi(\boldsymbol{x}_i)$. On souhaite résoudre l'ACP à noyaux, en termes de produit scalaire dans l'espace fonctionnel, $\langle \phi(\boldsymbol{x}_i), \phi(\boldsymbol{x}_j) \rangle_{\mathcal{H}}$, pour tout $i, j = 1, 2, \cdots, N$. La matrice de covariance dans $\mathcal{H}$ est donnée par :

$$\mathfrak{C}^{\phi} = \frac{1}{N} \sum_{i=1}^{N} \phi(\boldsymbol{x}_i) \phi(\boldsymbol{x}_i)^\top$$

Les r axes principaux, $\boldsymbol{w}_k \in \mathcal{H}$, pour $k = 1, 2, \cdots, r$, correspondent aux vecteurs propres ayant des valeurs propres λ_k tels que :

$$\mathfrak{C}^{\phi} \boldsymbol{w}_k = \lambda_k \boldsymbol{w}_k. \tag{1.12}$$

Toutes les solutions $\boldsymbol{w}_k$ se trouvent dans l'espace engendré par les images des échantillons par la fonction $\phi(\cdot)$. Cela signifie qu'il existe des coefficients $\alpha_{k,1}, \alpha_{k,2}, \cdots, \alpha_{k,N}$ de sorte que

$$\boldsymbol{w}_k = \sum_{i=1}^{N} \alpha_{k,i} \phi(\boldsymbol{x}_i). \tag{1.13}$$

En remplaçant l'expression de $\mathfrak{C}^{\phi}$ et l'équation (1.13) dans l'équation (1.12), nous obtenons :

$$N \lambda_k \boldsymbol{\alpha}_k = \boldsymbol{K} \boldsymbol{\alpha}_k, \tag{1.14}$$

où $\boldsymbol{K}$ est la matrice de taille $(N \times N)$ d'éléments $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$, et $\boldsymbol{\alpha}_k = [\alpha_{k,1}, \alpha_{k,2}, \cdots, \alpha_{k,N}]^\top$.

Conclusion

Dans ce chapitre, nous avons introduit les principaux concepts de l'apprentissage statistique utiles à cette thèse. Nous avons montré que les mé-

thodes à noyaux assurent le passage des échantillons provenant de l'espace d'entrée à un espace caractéristique (RKHS). Les algorithmes appliqués sur les échantillons dans les RKHS sont exprimés en fonction du produit scalaire entre les paires d'échantillons. Les produits scalaires sont évalués par les fonctions noyau, sans qu'il soit nécessaire d'expliciter la fonction non linéaire. Nous avons également présenté les méthodes à noyaux pour la régression par moindres carrés à noyaux et l'ACP à noyaux. Dans le chapitre suivant, nous utiliserons ce modèle afin de modéliser un champ de diffusion, et nous améliorons la modélisation grâce à l'ajustement des échantillons d'entrée.

Estimation d'un champ de diffusion et mobilité

2

Sommaire

Dans ce chapitre, nous traitons le problème de modélisation de la distribution d'un champ. D'une manière générale, nous nous intéressons aux phénomènes de diffusion classiques, avec des applications sur le gaz, la température,... Notre but est d'estimer cette distribution aux endroits dépourvus de capteurs. A cette fin, nous utilisons des capteurs mobiles. Ces derniers se déplacent dans la région sous surveillance, assurent sa couverture et améliorent l'estimation du champ diffusé.

Soient N capteurs déployés dans une région $\mathbb{X}$, où $\mathbb{X} \subset \mathbb{R}^2$ pour un espace de dimension 2 ou bien $\mathbb{X} \subset \mathbb{R}^3$ comme dans le cas des drones navigant dans un espace de dimension 3. Soient $\boldsymbol{x}_i \in \mathbb{X}$ la position du capteur i, $1 \leq i \leq N$, et $y_i \in \mathbb{R}$ la mesure prise par le capteur i de la quantité physique étudiée comme par exemple la concentration de gaz. Le champ de diffusion est estimé en utilisant les informations des capteurs, à savoir leurs positions et les mesures acquises. L'objectif est de déterminer un modèle ψ tel que :

$$\psi(\boldsymbol{x}_j) \approx y_j \qquad \text{pour tout } j = 1, 2, \cdots, N.$$

Nous proposons d'utiliser le formalisme des méthodes à noyaux pour déterminer ce modèle. Ces méthodes d'apprentissage ne nécessitent aucune

connaissance des paramètres régissant le phénomène physique, comme par exemple l'évolution de la source ou encore les conditions aux bords (Honeine et al. 2009; 2010, Ghadban et al. 2013a). Ensuite nous proposons des schémas d'optimisation du modèle en déplaçant les capteurs selon plusieurs stratégies de mobilité.

2.1 Méthode à noyaux pour l'estimation dans les RCSF

Dans le but d'estimer le champ de diffusion, les capteurs transmettent leurs informations à un centre de fusion (CF) où un calcul centralisé est mené. Le CF détermine la fonction $\psi(\cdot)$ en considérant le formalisme des méthodes à noyaux (Schölkopf et al. 2001). Soit $\mathcal{H}$ un espace de Hilbert à noyau reproduisant constitué de fonctions de $\mathbb{X}$ dans $\mathbb{R}$. On désigne par $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ son produit scalaire et par $\| \cdot \|_{\mathcal{H}}$ la norme correspondante. Le problème de minimisation de l'erreur quadratique entre les sorties du modèle $\psi(\boldsymbol{x}_i)$ et les réponses souhaitées y_i, mesurées par les capteurs, nous emmène au problème d'optimisation suivant :

$$\psi^*(\cdot) = \arg\min_{\psi \in \mathcal{H}} \sum_{j=1}^{N} |y_j - \psi(\boldsymbol{x}_j)|^2 + \nu \|\psi\|_{\mathcal{H}}^2, \tag{2.1}$$

où ν est un paramètre de régularisation qui contrôle le compromis entre le raccord aux données disponibles et la douceur de la solution.

La solution de ce problème est donnée par le Théorème de Représentation, comme présenté dans le Chapitre 1, section 1.5 :

$$\psi^*(\cdot) = \sum_{j=1}^{N} \alpha_j \kappa(\boldsymbol{x}_j, \cdot). \tag{2.2}$$

et

$$\boldsymbol{\alpha}^* = (\boldsymbol{K} + \nu \mathbf{I})^{-1} \boldsymbol{y}. \tag{2.3}$$

où $\boldsymbol{K}$ est la matrice de Gram, et $\boldsymbol{y}$ est le vecteur contenant les y_j où $j \in \{1, 2, \cdots, N\}$. Notons que la complexité de calcul est de l'ordre $\mathcal{O}(N^3)$ comme nous inversons une matrice de taille $(N \times N)$.

Nous avons présenté dans la Table 1.1 différents types de noyaux. Dans notre travail, nous utilisons les noyaux radiaux de la forme

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = g(\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2), \tag{2.4}$$

où $g(\cdot)$ est une fonction réelle positive. Les noyaux radiaux sont adaptés au problème de la diffusion de champ puisque ces noyaux dépendent uniquement de la distance et ils sont invariants par rapport à une translation. Soit $g'(\cdot)$ la première dérivée de la fonction $g(\cdot)$ par rapport à son argument, c'est-à-dire $g'(z) = \partial g(z)/\partial z$. Le noyau radial le plus utilisé est le noyau gaussian, où $g(\cdot) = \exp^{-\frac{1}{2\sigma^2}(\cdot)}$, avec σ la largeur de bande. Dans ce cas, nous avons $g'(z) = -\frac{1}{2\sigma^2} g(z)$.

2.1.1 Approche séquentielle

L'intégration d'un nouveau capteur, ainsi que la substitution des capteurs suite à un dépannage, nécessitent une inversion matricielle. L'inversion de la matrice de Gram de taille $(N \times N)$ par la méthode de cofacteurs a pour complexité de calcul $\mathcal{O}(N^3)$. Plus le nombre de capteurs N est grand, plus cette procédure prend du temps et consomme de l'énergie. Par conséquent, il faut avoir recours à des méthodes d'inversion plus rapide telle la méthode d'inversion par bloc et cela dans les cas d'ajout ou d'élimination d'un capteur, comme développé dans la suite. Cette méthode est également utilisée dans le cadre d'un traitement décentralisé.

Ajout d'un capteur

Soit $\boldsymbol{K}_{(N)} = (\boldsymbol{K} + \nu\mathbf{I})$, où $\boldsymbol{K}$ est la matrice de Gram de taille $(N \times N)$. L'inverse de $\boldsymbol{K}_{(N)}$ est supposé connu. Lorsqu'un capteur est ajouté, nous aboutissons à la matrice $\boldsymbol{K}_{(N+1)}$ de taille $((N+1) \times (N+1))$. L'inverse de cette dernière matrice est obtenue par l'inversion par bloc. En effet, la matrice $\boldsymbol{K}_{(N+1)}$ peut s'écrire en forme de bloc :

$$\boldsymbol{K}_{(N+1)} = \begin{pmatrix} \boldsymbol{K}_{(N)} & \boldsymbol{b} \\ \boldsymbol{c} & d \end{pmatrix}.$$

En se référant à Watt (2006), nous déterminons l'inverse de $\boldsymbol{K}_{(N+1)}$ à partir de celui de $\boldsymbol{K}_{(N)}$, selon :

$$\boldsymbol{K}^{-1}_{(N+1)} = \begin{pmatrix} \boldsymbol{K}^{-1}_{(N)} + \frac{1}{S_{\boldsymbol{K}_{(N)}}} \boldsymbol{K}^{-1}_{(N)} \boldsymbol{b}\boldsymbol{c} \boldsymbol{K}^{-1}_{(N)} & -\frac{1}{S_{\boldsymbol{K}_{(N)}}} \boldsymbol{K}^{-1}_{(N)} \boldsymbol{b} \\ -\frac{1}{S_{\boldsymbol{K}_{(N)}}} \boldsymbol{c}\boldsymbol{K}^{-1}_{(N)} & \frac{1}{S_{\boldsymbol{K}_{(N)}}} \end{pmatrix}. \tag{2.5}$$

Notons que $S_{\boldsymbol{K}_{(N)}} = (d - \boldsymbol{c}\boldsymbol{K}^{-1}_{(N)}\boldsymbol{b})$ désigne le complément de Schur de la matrice $\boldsymbol{K}_{(N+1)}$ par rapport à $\boldsymbol{K}_{(N)}$.

Le but principal est d'adapter les coefficients $\boldsymbol{\alpha}^*$ définis dans (2.3). En effet, une fois que nous trouvons l'inverse de $\boldsymbol{K}_{(N+1)}$, nous pouvons déterminer $\boldsymbol{\alpha}^*_{(N+1)}$ à partir de $\boldsymbol{\alpha}^*_{(N)}$ en revenant à l'équation (2.3) :

$$\boldsymbol{\alpha}^*_{(N+1)} = \begin{pmatrix} \boldsymbol{\alpha}^*_{(N)} - \frac{1}{S_{\boldsymbol{K}_{(N)}}} \boldsymbol{K}^{-1}_{(N)} \boldsymbol{b} y_{N+1} \\ -\frac{1}{S_{\boldsymbol{K}_{(N)}}} \boldsymbol{c}\boldsymbol{K}^{-1}_{(N)} \boldsymbol{y}_{(N)} + \frac{1}{S_{\boldsymbol{K}_{(N)}}} y_{N+1} \end{pmatrix}, \tag{2.6}$$

où $\boldsymbol{y}_{(N+1)} = [\boldsymbol{y}_{(N)} \quad y_{N+1}]^\top$. Nous utilisons cette méthode pour intégrer de nouveaux capteurs. Nous pouvons même l'utiliser pour construire la matrice $\boldsymbol{K}_{(N)}$ et déterminer les coefficients $\boldsymbol{\alpha}^*$ en intégrant les capteurs l'un après l'autre. Donc, en partant d'un seul élément, nous construisons au fur et à mesure la matrice $\boldsymbol{K}_{(N)}$ et les coefficients $\boldsymbol{\alpha}^*$ de tout le réseau. Ainsi, dans le cadre d'une stratégie séquentielle (dite non coopérative dans le chapitre suivant), la matrice $\boldsymbol{K}^{-1}_{(N)}$ et les coefficients $\boldsymbol{\alpha}^*$ sont transmis d'un capteur à un autre et adaptés "à la volée".

Elimination d'un capteur

Nous nous intéressons à l'élimination d'un capteur. C'est le cas, par exemple, lorsqu'un capteur j tombe en panne, il sera éliminé de la matrice $\boldsymbol{K}_{(N)}$. Nous procédons alors selon l'algorithme suivant.

1. Réarranger la matrice $\boldsymbol{K}_{(N)}^{-1}$ de sorte à avoir la dernière ligne et la dernière colonne, celles qui correspondent au capteur j. Nous obtenons ainsi la matrice $\boldsymbol{K}_{arrangée}^{-1}$.
2. Utiliser l'inversion par bloc pour éliminer la dernière ligne et la dernière colonne de $\boldsymbol{K}_{arrangée}^{-1}$. Nous retrouvons ainsi la matrice de Gram inversée $\boldsymbol{K}_{sans\ j}^{-1}$. En effet, d'après l'équation (2.6) :

$$\boldsymbol{K}_{arrangée}^{-1} = \begin{pmatrix} \boldsymbol{K}_{sans\ j}^{-1} + \frac{1}{S_{\boldsymbol{K}_{sans\ j}}} \boldsymbol{K}_{sans\ j}^{-1} \boldsymbol{b}\boldsymbol{c} \boldsymbol{K}_{sans\ j}^{-1} & -\frac{1}{S_{\boldsymbol{K}_{sans\ j}}} \boldsymbol{K}_{sans\ j}^{-1} \boldsymbol{b} \\ -\frac{1}{S_{\boldsymbol{K}_{sans\ j}}} \boldsymbol{c} \boldsymbol{K}_{sans\ j}^{-1} & \frac{1}{S_{\boldsymbol{K}_{sans\ j}}} \end{pmatrix}.$$

Soit $\boldsymbol{K}_{arrangée}^{-1} = \begin{pmatrix} \boldsymbol{U} & \boldsymbol{v} \\ \boldsymbol{s} & t \end{pmatrix}$.

Alors :

$$\begin{aligned} \boldsymbol{U} - \frac{\boldsymbol{v}\boldsymbol{s}}{t} &= \boldsymbol{K}_{sans\ j}^{-1} + \frac{1}{S_{\boldsymbol{K}_{sans\ j}}} \boldsymbol{K}_{sans\ j}^{-1} \boldsymbol{b}\boldsymbol{c} \boldsymbol{K}_{sans\ j}^{-1} \\ &\quad - \frac{1}{S_{\boldsymbol{K}_{sans\ j}}} \boldsymbol{K}_{sans\ j}^{-1} \boldsymbol{b} \times \frac{1}{S_{\boldsymbol{K}_{sans\ j}}} \boldsymbol{c} \boldsymbol{K}_{sans\ j}^{-1} \times S_{\boldsymbol{K}_{sans\ j}} \\ &= \boldsymbol{K}_{sans\ j}^{-1}, \end{aligned}$$

De même, soit $\boldsymbol{\alpha}_{arrangée}^{*} = [\boldsymbol{a} \quad \alpha_j]^\top$, alors

$$\boldsymbol{\alpha}_{sans\ j}^{*} = \boldsymbol{a} - \boldsymbol{v} y_j,$$

ce qui permet de déterminer l'inverse de la sous-matrice $\boldsymbol{K}_{sans\ j}$ à partir de l'inverse de la matrice $\boldsymbol{K}$ et $\boldsymbol{\alpha}_{arrangée}^{*}$ à partir de $\boldsymbol{\alpha}^{*}$.

2.2 Schémas d'optimisation

Afin d'améliorer l'estimation du champ, les capteurs sont capables de se déplacer. Ce sont des capteurs robots. Dans ce document, il s'agit d'une mobilité contrôlée où les capteurs mobiles se déplacent suite à une commande interne ou externe. Le but est de déterminer cette commande de la mobilité afin de pouvoir mieux couvrir la région sous surveillance. A cette fin, une fois le modèle construit à partir des capteurs avant mobilité, appelé modèle fixe, chaque capteur mobile peut se déplacer de sa position actuelle, désignée par $\boldsymbol{x}_t^+$ à l'instant t, à une nouvelle position $\boldsymbol{x}_{t+1}^+$. Le déplacement simultané de tous les capteurs mobiles est un problème insoluble, notamment lorsqu'il s'agit d'un réseau de capteurs à grande échelle. Notre objectif est de réduire la consommation d'énergie à cause des contraintes d'énergie et de calcul. Pour surmonter ce problème, les capteurs sont déplacés d'une manière séquentielle, l'un après l'autre.

L'objectif de la méthode proposée consiste à adapter de manière itérative la position des capteurs dans le but d'améliorer le modèle de régression. Notons que le capteur mobile ne doit pas intervenir dans l'estimation du modèle, c'est-à-dire dans l'ensemble d'apprentissage, mais a posteriori dans son ajustement. Cette considération évite le problème de sur-apprentissage. Soit ψ_t le modèle de régression à l'itération t. L'erreur

de validation du modèle en toute paire $(\boldsymbol{x}^+, y^+) \in \mathbb{X} \times \mathbb{Y}$ est définie par l'écart quadratique entre la sortie souhaitée y^+ et la sortie estimée du modèle $\psi_{t-1}(\boldsymbol{x}^+)$ à l'itération $t-1$:

$$\begin{aligned}\varepsilon_t(\boldsymbol{x}^+) &= (y^+ - \psi_{t-1}(\boldsymbol{x}^+))^2 \\ &= (y^+ - \sum_{j=1}^{N} \alpha_{j,t-1} \kappa(\boldsymbol{x}_{j,t-1}, \boldsymbol{x}^+))^2. \end{aligned} \tag{2.7}$$

Le capteur mobile adapte ensuite sa position selon l'un des scénarios détaillés dans la section 2.3. Il se déplace à la nouvelle position $\boldsymbol{x}_t^+$ de façon à réduire l'erreur de validation du modèle. Notons que les coefficients estimés $\alpha_{j,t}$ annulent l'erreur de validation du modèle estimé ψ_{t-1} en toute paire de l'ensemble d'apprentissage. Mais, cette erreur peut être plus élevée dans des endroits dépourvus de capteurs. D'où la mobilité, en déplaçant un capteur mobile vers cette zone afin d'y réduire l'erreur de validation. En effet, le capteur mobile, arrivant à cette région mal couverte, adapte le modèle de régression en utilisant sa nouvelle mesure et sa nouvelle position en apprentissage. Ainsi, l'erreur de validation du modèle diminue-t-elle lorsque le capteur mobile se déplace vers la région où l'erreur est la plus importante. La nouvelle position est alors obtenue par la résolution du problème d'optimisation suivant :

$$\boldsymbol{x}_t^+ = \arg\max_{\boldsymbol{x} \in \mathbb{X}} \varepsilon_t(\boldsymbol{x}). \tag{2.8}$$

Ce problème est un problème difficile, car il s'agit d'un problème d'optimisation non linéaire et non convexe, principalement à cause de la nature du noyau considéré. Afin de surmonter ces difficultés, nous proposons dans la suite plusieurs schémas itératifs d'optimisation de premier et second ordre : la méthode du point fixe, l'algorithme du gradient et la méthode de Newton. Plusieurs critères d'arrêt peuvent être définis : le nombre maximal d'itérations, ou l'erreur résiduelle minimale entre deux itérations successives. Notons que ces méthodes ne sont pas optimales pour résoudre un tel problème non linéaire et non convexe. Mais rien n'empêche d'avoir recourt à ces méthodes pour améliorer la solution initiale. Alors que les méthodes proposées peuvent être appliquées pour tout type de noyau, la formulation est donnée ici pour les noyaux radiaux, c'est-à-dire de la forme (2.4). Dans ce cas, le gradient de la fonction d'erreur (2.7) est donné par

$$\begin{aligned}\nabla\varepsilon_t(\boldsymbol{x}) = &-4\Big(y^+ - \sum_{j=1}^{N} \alpha_{j,t-1} \kappa(\boldsymbol{x}_{j,t-1}, \boldsymbol{x})\Big) \\ &\times \sum_{j=1}^{N} \alpha_{j,t-1}\, g'(\|\boldsymbol{x} - \boldsymbol{x}_{j,t-1}\|^2)(\boldsymbol{x} - \boldsymbol{x}_{j,t-1}).\end{aligned}$$

Dans cette expression, nous supposons que $\boldsymbol{x}$ est dans le voisinage de $\boldsymbol{x}^+$ où la sortie souhaitée reste constante et égale à y^+.

2.2.1 Méthode du point fixe

Le maximum de la fonction de coût quadratique (2.7) s'obtient lorsque son gradient s'annule. Par conséquent, la valeur optimale $\boldsymbol{x}^+$ est obtenue

par $\nabla\varepsilon_t(\boldsymbol{x}^+) = 0$, donnant lieu à une méthode itérative dite du point fixe (Cadariu et Radu 2008) de la forme $\boldsymbol{x}^+ = h(\boldsymbol{x}^+)$. Il suffit alors de substituer, à chaque itération t, la solution candidate $\boldsymbol{x}^+_{t-1}$ par la nouvelle solution candidate $\boldsymbol{x}^+_t = h(\boldsymbol{x}^+)$. En utilisant un noyau radial, nous avons :

$$-4\Big(y^+ - \sum_{j=1}^{N} \alpha_{j,t-1}\kappa(\boldsymbol{x}_{j,t-1}, \boldsymbol{x}^+)\Big) \sum_{j=1}^{N} \alpha_{j,t-1}\, g'(\|\boldsymbol{x}^+ - \boldsymbol{x}_{j,t-1}\|^2)(\boldsymbol{x}^+ - \boldsymbol{x}_{j,t-1}) = 0.$$

Puisque $\Big(y^+ - \sum_{j=1}^{N} \alpha_{j,t-1}\kappa(\boldsymbol{x}_{j,t-1}, \boldsymbol{x}^+)\Big) = 0$ emmène à $\varepsilon_t(\boldsymbol{x}^+) = 0$, nous en déduisons que :

$$\sum_{j=1}^{N} \alpha_{j,t-1}\, g'(\|\boldsymbol{x}^+ - \boldsymbol{x}_{j,t-1}\|^2)\,(\boldsymbol{x}^+ - \boldsymbol{x}_{j,t-1}) = 0.$$

L'expression du point fixe avec la règle de mise à jour de la solution candidate $\boldsymbol{x}^+_{t-1}$ à la nouvelle solution candidate $\boldsymbol{x}^+_t$ est la suivante :

$$\boldsymbol{x}^+_t = \frac{\sum_{j=1}^{N} \alpha_{j,t-1}\, g'(\|\boldsymbol{x}^+_{t-1} - \boldsymbol{x}_{j,t-1}\|^2)\, \boldsymbol{x}_{j,t-1}}{\sum_{j=1}^{N} \alpha_{j,t-1}\, g'(\|\boldsymbol{x}^+_{t-1} - \boldsymbol{x}_{j,t-1}\|^2)}. \tag{2.9}$$

La méthode du point fixe peut être instable et risque parfois de ne pas converger. Ces problèmes sont probablement dus à l'absence d'un paramètre de pas d'adaptation pour permettre un contrôle de la convergence de l'algorithme. Cette méthode n'impose pas de contrainte sur la distance parcourue par le capteur en question, c'est-à-dire $\|\boldsymbol{x}^+_t - \boldsymbol{x}^+_{t-1}\|$, ce qui la rend moins adaptée aux exigences des capteurs à énergie limitée. Dans ce qui suit nous présentons des méthodes mieux adaptées aux RCSF en contrôlant le pas, soit en utilisant un paramètre de pas, soit en suivant la courbure de $\varepsilon_t(\boldsymbol{x}^+_{t-1})$.

2.2.2 Algorithme du gradient

L'algorithme du gradient est une technique d'optimisation du premier ordre. Elle se base sur le gradient de la fonction coût $\varepsilon_t(\boldsymbol{x})$ par rapport à $\boldsymbol{x}$ (Fletcher et Powell 1963, Battiti 1992). En partant d'un point initial, cette technique adapte la solution dans la direction opposée au gradient, selon la règle de mise à jour à l'itération t suivante :

$$\boldsymbol{x}^+_t = \boldsymbol{x}^+_{t-1} + \eta_t \nabla\varepsilon_t(\boldsymbol{x}^+_{t-1}), \tag{2.10}$$

où $\eta_t > 0$ est le pas, qui dépend de l'itération t. L'utilisation du pas ici permet de contrôler la convergence, par opposition à la méthode du point fixe, le prix à payer étant le choix approprié de sa valeur. Puisque notre fonction coût est non convexe et non linéaire, des maxima locaux sont présents. Pour assurer une convergence appropriée, le pas η_t devrait être bien choisi. Une grande valeur de η_t rend l'algorithme instable. Une trop petite valeur de η_t nécessite un grand nombre d'itérations pour converger vers la solution, et le risque de converger vers une solution locale est plus important. Une solution est de faire diminuer la valeur de η_t à chaque itération (Robbins et Monro 1951). Un choix commun du pas est $\eta_t = \eta_0/t$,

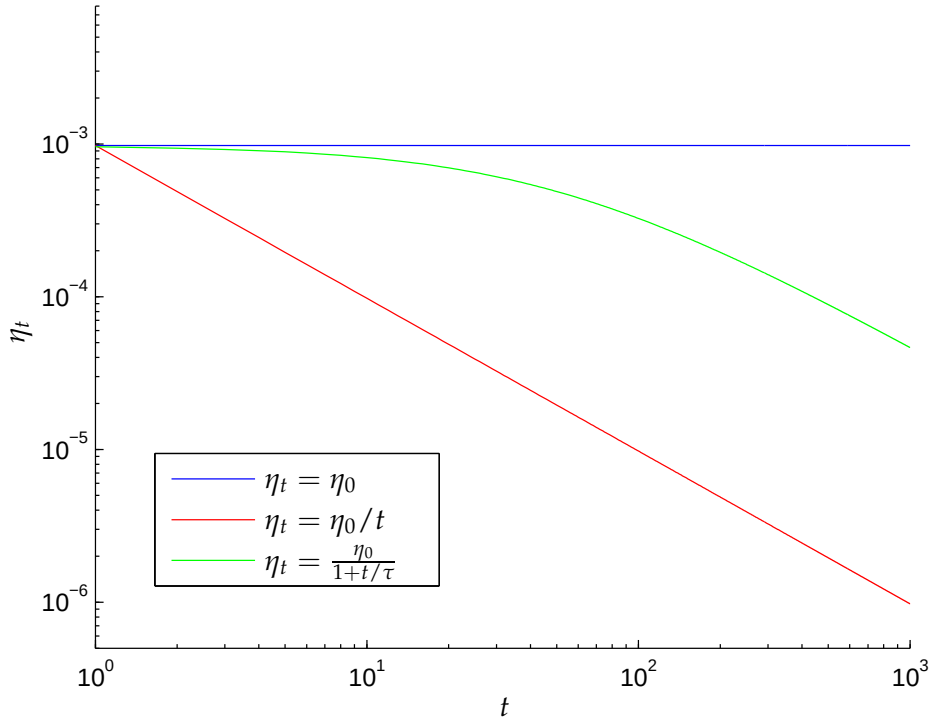

FIGURE 2.1 – *Comparaison entre les différents choix de* η_t *avec* $\eta_0 = 2^{-10}$ *et* $\tau = 50$.

où η_0 est un paramètre constant. Une autre possibilité est l'approche "rechercher puis converger" (Honeine 2012), avec $\eta_t = \frac{\eta_0}{1+t/\tau}$. Dans ce cas, le paramètre de retard τ détermine la durée de la phase de recherche initiale, avec $\eta_t \simeq \eta_0$ lorsque $t \ll \tau$, avant une phase de convergence où η_t diminue selon η_0/t lorsque $t \gg \tau$. Ceci est bien adapté à l'hypothèse de l'énergie limitée des capteurs puisque le pas diminue lorsqu'on s'approche de la solution. La FIGURE 2.1 compare les différents choix du paramètre η.

2.2.3 Méthode de Newton

La méthode de Newton (Grippo et al. 1986, Battiti 1992) est une technique d'optimisation du second ordre. Elle se base sur le gradient et la matrice hessienne de la fonction coût $\varepsilon_t(\boldsymbol{x})$ par rapport à $\boldsymbol{x}$. Le pas η_t est pris en se basant sur la matrice hessienne $\boldsymbol{H}_t$ estimée à l'instant t. Ainsi, nous obtenons :

$$\boldsymbol{x}_t^+ = \boldsymbol{x}_{t-1}^+ + \boldsymbol{H}_t^{-1} \nabla \varepsilon_t(\boldsymbol{x}_{t-1}^+). \tag{2.11}$$

Cette méthode permet de suivre la courbure de $\varepsilon_t(\boldsymbol{x}_{t-1}^+)$ en tenant compte de ses dérivées partielles secondes. De cette manière, la taille du pas est imposée par la courbure de la fonction de coût. Cependant, cette méthode a une plus grande complexité de calcul par rapport à l'algorithme du gradient.

Par définition, la matrice hessienne est :

$$\boldsymbol{H}_t = \begin{pmatrix} \frac{\partial^2 \varepsilon_t(\boldsymbol{x}_{t-1}^+)}{\partial x_{t-1,1}^{+2}} & \frac{\partial^2 \varepsilon_t(\boldsymbol{x}_{t-1}^+)}{\partial x_{t-1,1}^+ \partial x_{t-1,2}^+} & \cdots & \frac{\partial^2 \varepsilon_t(\boldsymbol{x}_{t-1}^+)}{\partial x_{t-1,1}^+ \partial x_{t-1,d}^+} \\ \frac{\partial^2 \varepsilon_t(\boldsymbol{x}_{t-1}^+)}{\partial x_{t-1,2}^+ \partial x_{t-1,1}^+} & \frac{\partial^2 \varepsilon_t(\boldsymbol{x}_{t-1}^+)}{\partial x_{t-1,2}^{+2}} & \cdots & \frac{\partial^2 \varepsilon_t(\boldsymbol{x}_{t-1}^+)}{\partial x_{t-1,2}^+ \partial x_{t-1,d}^+} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 \varepsilon_t(\boldsymbol{x}_{t-1}^+)}{\partial x_{t-1,d}^+ \partial x_{t-1,1}^+} & \frac{\partial^2 \varepsilon_t(\boldsymbol{x}_{t-1}^+)}{\partial x_{t-1,d}^+ \partial x_{t-1,2}^+} & \cdots & \frac{\partial^2 \varepsilon_t(\boldsymbol{x}_{t-1}^+)}{\partial x_{t-1,d}^{+2}} \end{pmatrix}$$

où $\boldsymbol{x}_{t-1}^+ = [x_{t-1,1}^+ \, x_{t-1,2}^+ \cdots x_{t-1,d}^+]$, d étant la dimension de l'espace. Dans le cas d'un espace de dimension deux, la matrice hessienne $\boldsymbol{H}_t$ est de la forme :

$$\boldsymbol{H}_t = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix},$$

où

$$\begin{aligned} h_{11} =& -2(y^+ - \sum_{l=1}^N \alpha_{l,t-1} \kappa(\boldsymbol{x}_{t+1}^+, \boldsymbol{x}_{l,t-1})) \sum_{j=1}^N \alpha_{j,t-1} \frac{\partial^2 \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{j,t-1})}{\partial x_{t-1,1}^{+2}} \\ & + 2\Big(\sum_{j=1}^N \alpha_{j,t-1} \frac{\partial \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{j,t-1})}{\partial x_{t-1,1}^+} \Big)^2 \end{aligned}$$

$$\begin{aligned} h_{12} &= h_{12} \\ &= -2(y^+ - \sum_{l=1}^N \alpha_{l,t-1} \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{l,t-1})) \sum_{j=1}^N \alpha_{j,t-1} \frac{\partial^2 \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{j,t-1})}{\partial x_{t-1,1}^+ \partial x_{t-1,2}^+} \\ &\quad + 2 \sum_{j=1}^N \alpha_{j,t-1} \frac{\partial \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{j,t-1})}{\partial x_{t-1,1}^+} \sum_{l=1}^N \alpha_{l,t-1} \frac{\partial \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{l,t-1})}{\partial x_{t-1,2}^+} \end{aligned}$$

$$\begin{aligned} h_{22} =& -2(y^+ - \sum_{l=1}^N \alpha_{l,t-1} \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{l,t-1})) \sum_{j=1}^N \alpha_{j,t-1} \frac{\partial^2 \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{j,t-1})}{\partial x_{t-1,2}^{+2}} \\ & + 2\Big(\sum_{j=1}^N \alpha_{j,t-1} \frac{\partial \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{j,t-1})}{\partial x_{t-1,2}^+} \Big)^2 \end{aligned}$$

Pour un noyau radial, nous obtenons :

$$\begin{aligned} \nabla \varepsilon_t(\boldsymbol{x}_{t-1}^+) =& -4(y^+ - \sum_{l=1}^N \alpha_{l,t-1} \kappa(\boldsymbol{x}_{t-1}^+, \boldsymbol{x}_{l,t-1})) \\ & \sum_{j=1}^N \alpha_{j,t-1} g^{(1)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}\|^2) \, (\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}), \end{aligned}$$

$$\begin{aligned} h_{11} =& -4(y^+ - \sum_{l=1}^N \alpha_{l,t-1} g(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{l,t-1}\|^2)) \Big[\sum_{j=1}^N \alpha_{j,t-1} g^{(1)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}\|^2) \\ & + 2 \sum_{j=1}^N \alpha_{j,t-1} g^{(2)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}\|^2)(x_{t-1,1}^+ - x_{j,t-1,1})^2 \Big] \\ & + 8\Big(\sum_{j=1}^N \alpha_{j,t-1} g^{(1)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}\|^2)(x_{t-1,1}^+ - x_{j,t-1,1}) \Big)^2 \end{aligned}$$

$$
\begin{aligned}
h_{12} &= h_{21} \\
&= -8(y^+ - \sum_{l=1}^{N} \alpha_{l,t-1} g(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{l,t-1}\|^2)) \\
&\quad \Big[\sum_{j=1}^{N} \alpha_{j,t-1} g^{(2)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}\|^2)(x_{t-1,1}^+ - x_{j,t-1,1})(x_{t-1,2}^+ - x_{j,t-1,2}) \Big] \\
&+ 8\Big(\sum_{j=1}^{N} \alpha_{j,t-1} g^{(1)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}\|^2)(x_{t-1,1}^+ - x_{j,t-1,1}) \Big) \\
&\quad \Big(\sum_{l=1}^{N} \alpha_{l,t-1} g^{(1)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{l,t-1}\|^2)(x_{t-1,2}^+ - x_{l,t-1,2}) \Big)
\end{aligned}
$$

$$
\begin{aligned}
h_{22} = &- 4(y^+ - \sum_{l=1}^{N} \alpha_{l,t-1} g(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{l,t-1}\|^2)) \Big[\sum_{j=1}^{N} \alpha_{j,t-1} g^{(1)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}\|^2) \\
&+ 2 \sum_{j=1}^{N} \alpha_{j,t-1} g^{(2)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}\|^2)(x_{t-1,2}^+ - x_{j,t-1,2})^2 \Big] \\
&+ 8\Big(\sum_{j=1}^{N} \alpha_{j,t-1} g^{(1)}(\|\boldsymbol{x}_{t-1}^+ - \boldsymbol{x}_{j,t-1}\|^2)(x_{t-1,2}^+ - x_{j,t-1,2}) \Big)^2
\end{aligned}
$$

où $g^{(r)}(z)$ correspond à la $r^{\text{ème}}$ dérivée de g par rapport à z.

2.3 Scénarios de mobilité

Dans la section précédente, nous avons détaillé les différents schémas d'optimisation pour réduire l'erreur de validation et déplacer un capteur mobile. Dans cette section, nous présentons plusieurs scénarios pour choisir le capteur mobile et adapter l'ensemble d'apprentissage. Dans chaque scénario, nous définissons l'ensemble d'apprentissage pour construire le modèle, et l'ensemble des capteurs mobiles et leur stratégie pour améliorer les performances du modèle (Ghadban et al. 2013b; 2014). Il est à noter que dans les trois scénarios un seul capteur se déplace afin que la comparaison des performances obtenues soit utile.

2.3.1 Scénario A : mobilité par modèles locaux

Dans ce scénario, nous proposons d'effectuer une estimation locale pour chaque capteur, à partir d'informations collectées des capteurs voisins. C'est une approche décentralisée, n'ayant pas besoin d'un centre de fusion. Chaque capteur construit son propre modèle localement en communiquant avec ses voisins. Ainsi, la complexité de calcul et de communication se réduit-elle. Nous introduisons le voisinage d'un capteur i, $\mathcal{N}_i$, qui est l'ensemble des indices des nœuds voisins au nœud i, c'est-à-dire les nœuds qui y sont directement connectés. Nous considérons que le nœud i n'est pas adjacent à lui-même, c'est-à-dire $i \notin \mathcal{N}_i$. Soient

$$
\mathcal{A}_{i,t} = \{(\boldsymbol{x}_{n_{i,1},t}, y_{n_{i,1},t}), (\boldsymbol{x}_{n_{i,2},t}, y_{n_{i,2},t}), \cdots, (\boldsymbol{x}_{n_{i,k},t}, y_{n_{i,k},t})\} \tag{2.12}
$$

l'ensemble d'apprentissage à l'itération t associé au capteur i, où $n_{i,1}, n_{i,2}, \cdots, n_{i,k} \in \mathcal{N}_i$ et $\psi_{i,t}(\cdot)$ le modèle associé au capteur i, déterminé conformément à la section 2.1 en utilisant son ensemble d'apprentissage $\mathcal{A}_{i,t}$, à savoir

$$\psi_{i,t}(\cdot) = \sum_{j \in \mathcal{N}_i} \alpha_{i,j,t}\, \kappa(\boldsymbol{x}_{j,t}, \cdot), \tag{2.13}$$

où les coefficients $\alpha_{i,j,t}$ sont obtenus en utilisant l'expression (2.3) avec l'ensemble $\mathcal{A}_{i,t}$.

L'optimisation conjointe de toutes les positions $\boldsymbol{x}_{j,t}$ et tous les coefficients $\alpha_{i,j,t}$, pour $j \in \mathcal{N}_i$, est un problème insoluble. Ceci est principalement à cause de la non-linéarité de $\psi_{i,t}(\cdot)$ par rapport à chacun des échantillons $\boldsymbol{x}_{j,t}$, alors qu'il reste linéaire par rapport aux coefficients. Pour surmonter ces difficultés, nous proposons de mettre à jour une seule position d'un capteur de l'ensemble d'apprentissage à chaque itération. Plusieurs stratégies sont adoptées pour choisir le capteur à déplacer à chaque itération :

- la randomisation avec une sélection aléatoire d'un capteur parmi tous les autres capteurs. C'est une sélection avec ou sans synchronisation.
- la sélection cyclique où les capteurs sont sélectionnées séquentiellement selon un ordre prédéfini.
- la sélection qui considère le capteur qui a la plus grande erreur de validation. Cette sélection nécessite la transmission des erreurs de validation à un centre de fusion.

Il est à noter que la dernière stratégie de sélection surpasse les autres critères de sélection puisque la plus grande erreur est ainsi réduite. Elle est donc considérée dans nos méthodes.

La stratégie d'optimisation proposée opère selon les étapes suivantes à chaque itération t :

1. Définir les N sous-ensembles d'apprentissage $\mathcal{A}_{i,t}$, pour $i = 1, 2, \cdots, N$, à partir des voisinages $\mathcal{N}_i$.
2. Estimer, pour $i = 1, 2, \cdots, N$, les N modèles locaux $\psi_{i,t}(\cdot)$ en utilisant les sous-ensembles d'apprentissage respectifs $\mathcal{A}_{i,t}$, conformément à la section 2.1, avec

$$\psi_{i,t}(\cdot) = \sum_{j \in \mathcal{N}_i} \alpha_{i,j,t}\, \kappa(\boldsymbol{x}_{j,t}, \cdot).$$

3. Calculer l'erreur de validation pour chaque modèle local $\psi_{i,t}(\cdot)$ en utilisant l'échantillon correspondant $(\boldsymbol{x}_{i,t}, y_{i,t})$ pour la validation, pour $i = 1, 2, \cdots, N$. Nous rappelons que le sur-apprentissage est évité, puisque $i \notin \mathcal{N}_i$ par construction, c'est-à-dire les informations $(\boldsymbol{x}_{i,t}, y_{i,t})$ du capteur i interviennent dans l'ajustement et non pas dans l'estimation du modèle associé $\psi_{i,t}$. L'erreur est donnée selon (2.7) comme suit :

$$\varepsilon_{i,t+1}(\boldsymbol{x}_{i,t}) = |y_{i,t} - \psi_{i,t}(\boldsymbol{x}_{i,t})|^2.$$

A noter que la moyenne des erreurs de validations $\varepsilon_{i,t}(\boldsymbol{x}_{i,t-1})$, $i = 1, 2, \cdots, N$, est utilisée pour étudier les performances de la mobilité.

4. Sélectionner, parmi les N capteurs, le capteur m selon la stratégie adaptée : randomisation, sélection cyclique ou en considérant la plus grande erreur de validation.
5. Adapter la position du capteur sélectionné, en le déplaçant de sa position actuelle $\boldsymbol{x}_{m,t}$ à sa nouvelle position $\boldsymbol{x}_{m,t+1}$. Cette dernière est obtenue comme décrit dans la section 2.2, en se déplaçant selon la plus grande erreur de validation, c'est-à-dire :

$$\boldsymbol{x}_{m,t+1} = \arg\max_{\boldsymbol{x}\in\mathbb{X}} \varepsilon_{m,t+1}(\boldsymbol{x}). \tag{2.14}$$

   Tous les autres capteurs restent immobiles, soit $\boldsymbol{x}_{j,t+1} = \boldsymbol{x}_{j,t}$ pour l'ensemble $j = 1, 2, \cdots, N$ et $j \neq m$.
6. Mettre à jour les ensembles $\mathcal{A}_{i,t}$ à $\mathcal{A}_{i,t+1}$ en tenant compte de l'éventuel changement du voisinage de chaque capteur selon (2.12).

2.3.2 Scénario B : mobilité avec un modèle global

Dans ce scénario, nous proposons d'effectuer une estimation globale. C'est une approche centralisée. Les capteurs transmettent leurs informations à un centre de fusion qui construit le modèle à partir de toutes les données. Soient

$$\mathcal{B}_t = \{(\boldsymbol{x}_{1,t}, y_{1,t}), (\boldsymbol{x}_{2,t}, y_{2,t}), \cdots, (\boldsymbol{x}_{N,t}, y_{N,t})\} \tag{2.15}$$

l'ensemble d'apprentissage à l'itération t et $\psi_t(\cdot)$ le modèle déterminé conformément à la section 2.1 à partir de cet ensemble d'apprentissage, à savoir

$$\psi_t(\cdot) = \sum_{j=1}^{N} \alpha_{j,t}\, \kappa(\boldsymbol{x}_{j,t}, \cdot), \tag{2.16}$$

où les coefficients $\alpha_{j,t}$ sont obtenus en utilisant l'expression (2.3) avec l'ensemble d'apprentissage $\mathcal{B}_t$.

Rappelons nous que le modèle ψ_t donné par (2.16) a une erreur optimiste en tout élément de l'ensemble d'apprentissage et que le capteur mobile doit intervenir seulement dans l'ajustement du modèle. Pour cela, afin de déterminer le capteur qui "corrige le plus" l'erreur de validation, nous adoptons la stratégie "leave-one-out". Dans ce schéma, un seul capteur est exclu de l'ensemble d'apprentissage, et l'erreur de validation est calculée en ce capteur.

La stratégie d'optimisation proposée opère dans les étapes suivantes à chaque itération t :

1. Définir les N sous-ensembles d'apprentissage $\mathcal{B}_{i,t}$, pour $i = 1, 2, \cdots, N$, où
$$\mathcal{B}_{i,t} = \mathcal{B}_t \backslash \{(\boldsymbol{x}_{i,t}, y_{i,t})\}.$$
2. Estimer, pour $i = 1, 2, \cdots, N$, tous les N sous-modèles $\psi_{i,t}(\cdot)$ en utilisant les sous-ensembles d'apprentissage respectifs $\mathcal{B}_{i,t}$:

$$\psi_{i,t}(\cdot) = \sum_{\substack{j=1 \\ j\neq i}}^{N} \alpha_{i,j,t}\, \kappa(\boldsymbol{x}_{j,t}, \cdot).$$

3. Calculer, pour chaque $i = 1, 2, \cdots, N$, l'erreur de validation entre le sous-modèle $\psi_{i,t}(\boldsymbol{x}_{i,t})$ et la mesure souhaitée $y_{i,t}$:

$$\varepsilon_{i,t+1}(\boldsymbol{x}_{i,t}) = |y_{i,t} - \psi_{i,t}(\boldsymbol{x}_{i,t})|^2.$$

4. Sélectionner le capteur m, avec $m \in \{1, 2, \cdots, N\}$, selon la stratégie adaptée : randomisation, sélection cyclique ou en considérant la plus grande erreur de validation.
5. Déplacer le capteur sélectionné de sa position actuelle $\boldsymbol{x}_{m,t}$ à sa nouvelle position $\boldsymbol{x}_{m,t+1}$ telle que :

$$\boldsymbol{x}_{m,t+1} = \arg\max_{\boldsymbol{x} \in \mathbb{X}} \varepsilon_{m,t+1}(\boldsymbol{x}). \tag{2.17}$$

   Tous les autres capteurs restent immobiles, soit $\boldsymbol{x}_{j,t+1} = \boldsymbol{x}_{j,t}$ pour l'ensemble $j = 1, 2, \cdots, N$ et $j \neq m$.
6. Mettre à jour l'ensemble $\mathcal{B}_t$ à $\mathcal{B}_{t+1}$ selon :

$$\mathcal{B}_{t+1} = \mathcal{B}_t \backslash \{(\boldsymbol{x}_{m,t}, y_{m,t})\} \cup \{(\boldsymbol{x}_{m,t+1}, y_{m,t+1})\}.$$

2.3.3 Scénario C : mobilité avec des robots en présence de capteurs fixes

Dans ce scénario, nous divisons les capteurs entre capteurs fixes et capteurs mobiles, appelés robots dans la suite pour les distinguer des capteurs fixes. Cette division est motivée par le fait que les capteurs fixes coûtent moins chers que les capteurs mobiles. Les robots se déplacent pour chercher l'information et ils ont la capacité de communiquer entre eux.

Un modèle global est construit en utilisant l'ensemble $\mathcal{B}_t$ défini dans (2.15). Soit $\psi_t(\cdot)$ le modèle global calculé, tel que :

$$\psi_t(\cdot) = \sum_{j=1}^{N} \alpha_{j,t}\, \kappa(\boldsymbol{x}_{j,t}, \cdot), \tag{2.18}$$

où les coefficients $\alpha_{j,t}$ sont obtenus en utilisant l'expression (2.3) avec l'ensemble d'apprentissage $\mathcal{B}_t$.

Nous proposons d'utiliser M capteurs robots. Ces capteurs robots sont désignés par l'ensemble $\mathcal{C}_t = \{(\boldsymbol{x}^*_{1,t}, y^*_{1,t}), (\boldsymbol{x}^*_{2,t}, y^*_{2,t}), \cdots, (\boldsymbol{x}^*_{M,t}, y^*_{M,t})\}$, où $\boldsymbol{x}^*_i \in \mathbb{X}$ désigne la position du robot i et $y^*_i \in \mathbb{R}$ la mesure acquise à cette position, pour $i \in \{1, 2, \cdots, M\}$. Un seul robot est déplacé à chaque itération. La stratégie d'optimisation proposée opère suivant les étapes suivantes à chaque itération t :

1. Estimer le modèle $\psi_t(\cdot)$ en utilisant l'ensemble d'apprentissage $\mathcal{B}_t$, avec

$$\psi_t(\cdot) = \sum_{j=1}^{N} \alpha_{j,t}\, \kappa(\boldsymbol{x}_{j,t}, \cdot).$$

2. Calculer, pour $i = 1, 2, \cdots, M$, l'erreur de validation pour chaque robot $(\boldsymbol{x}^*_{i,t}, y^*_{i,t})$, selon (2.7) avec

$$\varepsilon_{t+1}(\boldsymbol{x}^*_{i,t}) = |y^*_{i,t} - \psi_t(\boldsymbol{x}^*_{i,t})|^2.$$

3. Sélectionner, à partir de tous les M robots, le robot m selon une randomisation, une sélection cyclique ou en considérant la plus grande erreur de validation.
4. Déplacer le robot de $\boldsymbol{x}^*_{m,t}$ à $\boldsymbol{x}^*_{m,t+1}$ afin de se déplacer vers une plus grande erreur de validation en maximisant la fonction $\varepsilon_{t+1}(\cdot)$,

$$\boldsymbol{x}^*_{m,t+1} = \arg\max_{\boldsymbol{x}\in\mathbb{X}} \varepsilon_{t+1}(\boldsymbol{x}). \tag{2.19}$$

   Tous les autres robots restent fixes, c'est-à-dire $\boldsymbol{x}^*_{j,t+1} = \boldsymbol{x}^*_{j,t}$ pour tout $j = 1, 2, \cdots, M$ et $j \neq m$.
5. Mettre à jour l'ensemble $\mathcal{C}_t$ à $\mathcal{C}_{t+1}$ comme suit :

$$\mathcal{C}_{t+1} = \mathcal{C}_t \backslash \{(\boldsymbol{x}^*_{m,t}, y^*_{m,t})\} \cup \{(\boldsymbol{x}^*_{m,t+1}, y^*_{m,t+1})\}. \tag{2.20}$$

6. Mettre à jour l'ensemble $\mathcal{B}_t$ à $\mathcal{B}_{t+1}$ en ajoutant les informations $(\boldsymbol{x}^*_{m,t}, y^*_{m,t})$ à l'ensemble $\mathcal{B}_t$, et en incrémentant l'ordre N du modèle : $\mathcal{B}_{t+1} = \mathcal{B}_t \cup \{(\boldsymbol{x}^*_{m,t}, y^*_{m,t})\}$.

2.3.4 Contrainte sur la solution

Les méthodes proposées fournissent des schémas d'optimisation itératives. Malheureusement, rien n'empêche d'avoir une mise à jour à l'extérieur du domaine admissible $\mathbb{X}$, où $\mathbb{X}$ représente la région à surveiller dans laquelle les capteurs sont déployés, à titre d'exemple $\mathbb{X}$ peut être une région carré. On exige souvent des contraintes sur la solution : une solution bornée par exemple en contrôle automatique, la non-négativité comme en analyse spectrale, ou encore la conservation de flux telle que la contrainte de somme unitaire. Nous proposons d'adapter les techniques itératives susmentionnées pour imposer de telles contraintes. A cette fin, et puisque $\boldsymbol{x}_{i,t-1}$ est supposé admissible, nous imposons à $\boldsymbol{x}_{i,t}$ d'appartenir à $\mathbb{X}$. Ainsi, si le calcul envoie le capteur à l'extérieur de $\mathbb{X}$, le capteur reste à sa place.

2.4 Résultats expérimentaux

Afin d'illustrer les résultats, nous considérons la diffusion d'un gaz à partir d'une source, dans une région carré $\mathbb{X} = [-0.5; 0.5] \times [-0.5; 0.5]$. Cette diffusion est régie par l'équation différentielle suivante :

$$\frac{\partial G(\mathbf{x}, t)}{\partial t} - c\nabla^2_{\mathbf{x}} G(\mathbf{x}, t) = Q(\mathbf{x}, t), \tag{2.21}$$

où $G(\mathbf{x}, t)$ est la densité de gaz dépendant de la position et du temps, $\nabla^2_{\mathbf{x}}$ est l'opérateur spatial de Laplace, $Q(\mathbf{x}, t)$ correspond à la quantité de gaz ajoutée à la position $\boldsymbol{x}$ à l'instant t et c est la conductivité du milieu fixée à 0.1. Une source de gaz placée au centre de la région est activée. Nous utilisons $N = 100$ capteurs déployés aléatoirement dans la région considérée. Le volume des capteurs est supposé très petit par rapport à la dimension de l'espace. Les simulations, effectuées sur Matlab, visent à déterminer la quantité de gaz à tout point de la région sous surveillance.

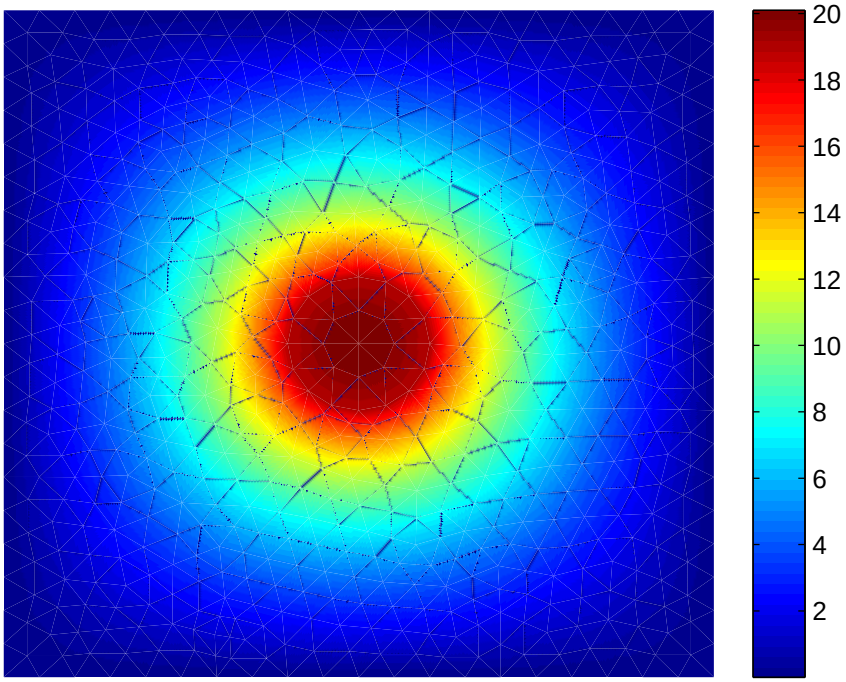

FIGURE 2.2 – *La distribution de gaz dans l'espace.*

Une fois qu'elle a atteint un certain équilibre, la distribution du gaz dans l'espace est considérée stationnaire ; elle est supposée invariante dans le temps. La FIGURE 2.2 représente la distribution de gaz dans la région $\mathbb{X}$. Dans ce qui suit, nous déterminons d'abord les paramètres du modèle fixe $\psi_t(\cdot)$ et le voisinage des capteurs pour l'estimation locale. Ces paramètres sont choisis d'une manière à diminuer l'erreur de validation du modèle fixe. Ensuite nous appliquons les différents scénarios de mobilité et les schémas d'optimisation afin de réduire l'erreur de validation et améliorer le modèle. Enfin, nous étudions les performances de ces méthodes et nous les comparons avec des méthodes de l'état de l'art.

2.4.1 Détermination du modèle fixe

Afin d'illustrer les méthodes proposées, il est nécessaire de régler quelques paramètres. La première étape consiste à déterminer les paramètres du modèle fixe. Dans ce but, nous divisons les $N = 100$ capteurs en $N_a = 66$ capteurs pour l'apprentissage et $N_v = 34$ capteurs pour la validation. Soient ω_a l'ensemble d'apprentissage, et ω_v l'ensemble de validation. Le modèle global dépend alors d'un seul paramètre, la largeur de bande σ du noyau gaussien. D'autre part, le scénario A définit un modèle local pour chaque capteur, en se basant sur les informations collectées par son voisinage. Ceci implique qu'il nous faut déterminer le voisinage $\mathcal{N}_i$ pour $i = 1, 2, \cdots, N$.

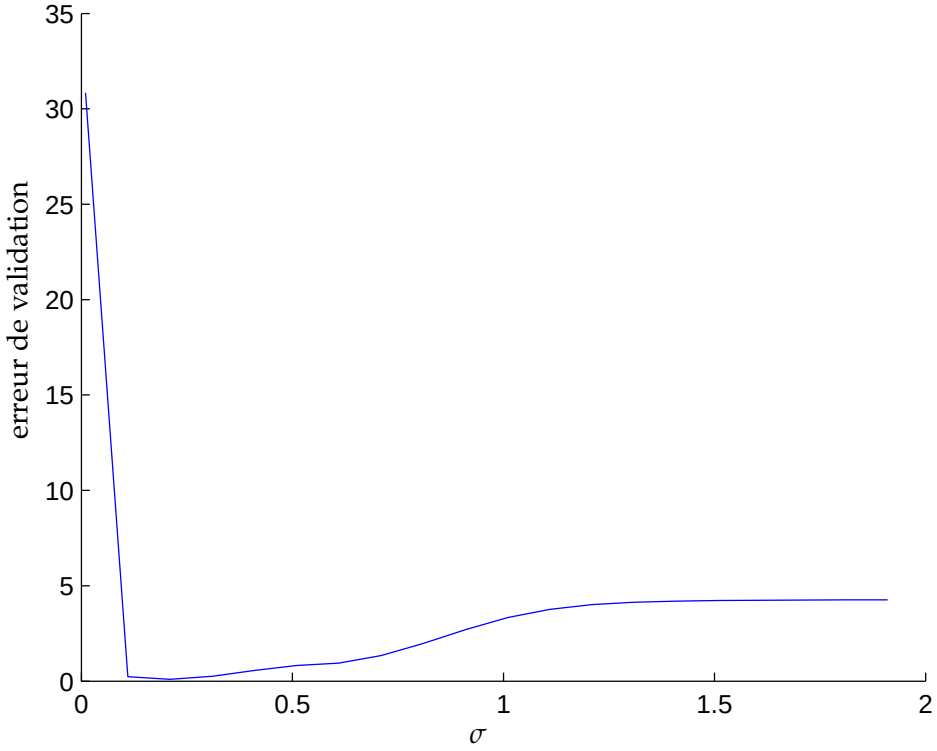

FIGURE 2.3 – *Impact de la valeur de la largeur de bande σ sur le modèle résultant. En utilisant* 20 *simulations de Monte-Carlo pour générer* 100 *capteurs déployés aléatoirement, l'erreur de validation est estimée en utilisant* 34 *capteurs.*

Le paramètre σ

Le modèle dépend de la largeur de bande σ du noyau gaussien. Pour étudier l'effet de ce paramètre, 20 simulations de Monte-Carlo de 100 capteurs déployés aléatoirement sont considérées. En variant σ entre 0.01 et 2 avec un pas de 0.01 dans le modèle fixe, l'erreur quadratique de validation est estimée sur ω_v selon $\frac{1}{N_v}\sum_{i\in\omega_v}(y_i - \psi_i(\mathbf{x}_i))^2$. Les résultats sont donnés dans la FIGURE 2.3. Nous remarquons que pour des petites valeurs de σ, l'erreur est grande suite à un sous-lissage qui provoque l'apparition de détails artificiels. Alors que les grandes valeurs de σ entraînent un fort lissage et la majorité des caractéristiques est effacée, augmentant ainsi l'erreur. Puisque l'erreur de validation est minimale pour $\sigma \simeq 0.1$, nous considérons dans ce qui suit cette valeur de largeur de bande du noyau gaussien.

Le voisinage $\mathcal{N}_i$

Le voisinage $\mathcal{N}_i$ d'un capteur i est défini par les k plus proches capteurs, la valeur de k étant pré-fixée. Le nombre de voisins k dépend de la densité des capteurs déployés dans $\mathbb{X}$ et de la portée des capteurs en communication. En outre, sa valeur affecte fortement la complexité de calcul, car il définit la taille de la matrice de Gram à inverser. Le paramètre k est choisi en minimisant l'erreur de validation moyenne entre les me-

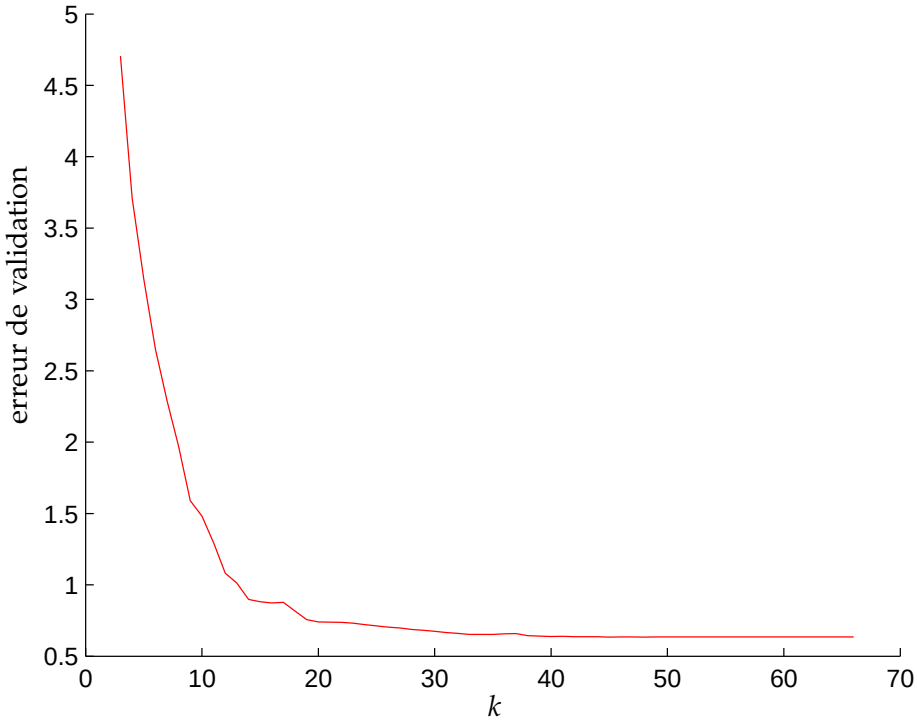

FIGURE 2.4 – *Influence du nombre de voisins k avec 100 capteurs, dont 34 capteurs pour la validation et moyennant sur* 20 *simulations de Monte Carlo.*

sures y_ℓ, $\ell \in \omega_v$, et les sorties de modèles obtenus sans mobilité $\psi_i(\boldsymbol{x}_\ell)$ où $i \in \omega_a$ est l'indice du plus proche capteur de ℓ. Pour cela, 20 simulations de Monte-Carlo de 100 capteurs déployés aléatoirement sont considérées. La FIGURE 2.4 montre l'erreur de validation, moyennée sur les simulations de Monte-Carlo, en fonction du nombre de voisins considérés. Nous constatons que l'erreur diminue avec l'augmentation du nombre de voisins. En effet, un petit nombre de voisins ne permet pas de bien représenter toutes les données, c'est pourquoi l'erreur de validation est grande pour des petites valeurs de k. Par contre, quand k augmente, le nombre des voisins représentant les données est plus grand, entraînant ainsi une meilleure description du réseau. Au delà de $k = 30$, l'erreur sature, ce qui implique que l'information apportée par un plus grand nombre de voisins a une contribution supplémentaire minime. Et comme l'augmentation de k augmente la complexité de calcul, nous choisissons $k = 20$ qui représente un bon compromis entre la précision du modèle et la complexité calculatoire.

2.4.2 Mobilité

Une fois le modèle fixe est déterminé, nous appliquons les scénarios de mobilité décrits dans la section 2.3. Deux cas de réseaux sont pris en considération. Dans le premier cas, les capteurs sont statiques, et chaque capteur détermine un modèle local. Dans le second cas, les capteurs sont mobiles et les modèles locaux sont calculés et utilisés pour la mobilité.

La pertinence du modèle est évaluée à l'aide de l'erreur de valida-

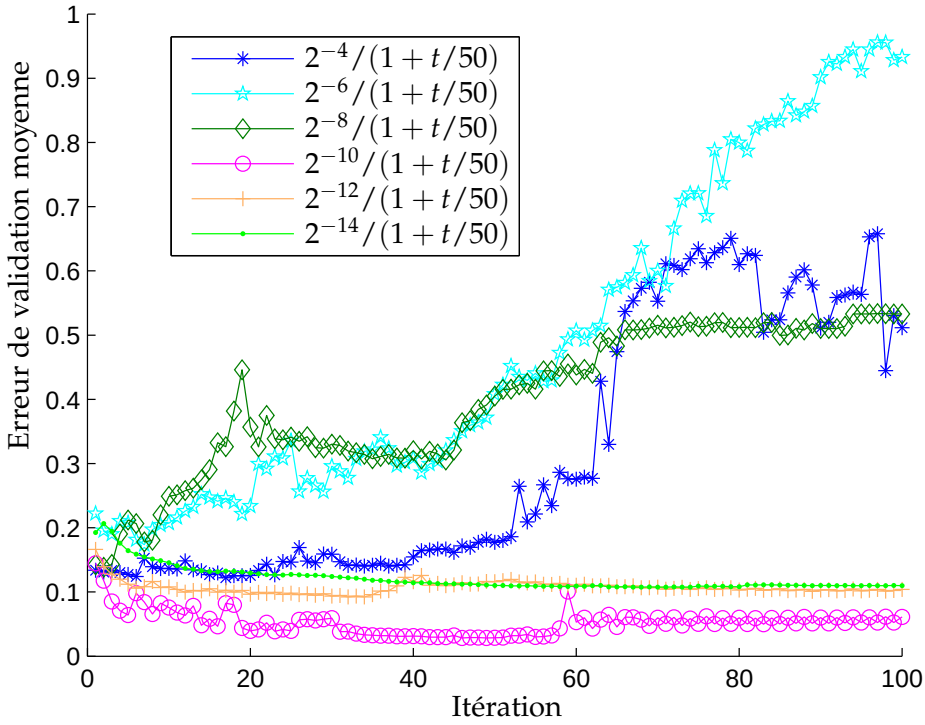

FIGURE 2.5 – *Evolution de l'erreur de validation moyenne pour différentes valeurs du paramètre η_0 du pas dans le cas du scénario A.*

tion estimée sur les capteurs mobiles et moyennée sur 20 simulations de Monte-Carlo. Précisons que le capteur mobile ne participe pas au modèle, de sorte qu'il ne peut servir pour tester l'efficacité du modèle proposé. Au-delà de l'erreur d'approximation, nous sommes souvent concernés par la distance parcourue, puisqu'elle est liée à la consommation d'énergie dans un réseau de capteurs mobiles. En raison de l'énergie limitée des capteurs, il est préférable que la distance parcourue soit minimale. Dans ce qui suit, 20 simulations de Monte-Carlo sont effectuées, chacune étant composée de 100 itérations, c'est-à-dire $t = 1, 2, \cdots, 100$. A chaque itération, un seul capteur est déplacé selon la mobilité de l'une des méthodes indiquées dans la section 2.3.

Pour les scénarios A et B, nous choisissons, parmi tous les capteurs, le capteur i dont le modèle donne la plus grande erreur d'estimation $\varepsilon_{i,t}(\boldsymbol{x}_{i,t-1})$. Contrairement aux autres méthodes, l'algorithme du gradient dépend du pas η_t, qui est pris de la forme $\frac{\eta_0}{1+t/\tau}$, tel que recommandé dans la section 2.2.2. Pour choisir ce paramètre, nous considérons 20 simulations Monté Carlo pour chaque valeur du pas. La FIGURE 2.5 et la FIGURE 2.6 montrent l'évolution, à chaque mouvement, de l'erreur de validation moyenne, pour plusieurs combinaisons des valeurs des paramètres η_0 et τ respectivement, pour le scénario A. La FIGURE 2.7 et la FIGURE 2.8 correspondent à ceux du scénario B. Ces figures montrent de bonnes performances pour les valeurs $\eta_0 = 2^{-10}$ et $\tau = 60$ dans le cas du scénario A et du scénario B. La FIGURE 2.9 et la FIGURE 2.10 illustrent l'erreur de validation à chaque itération pour le modèle fixe et les modèles avec mobilité

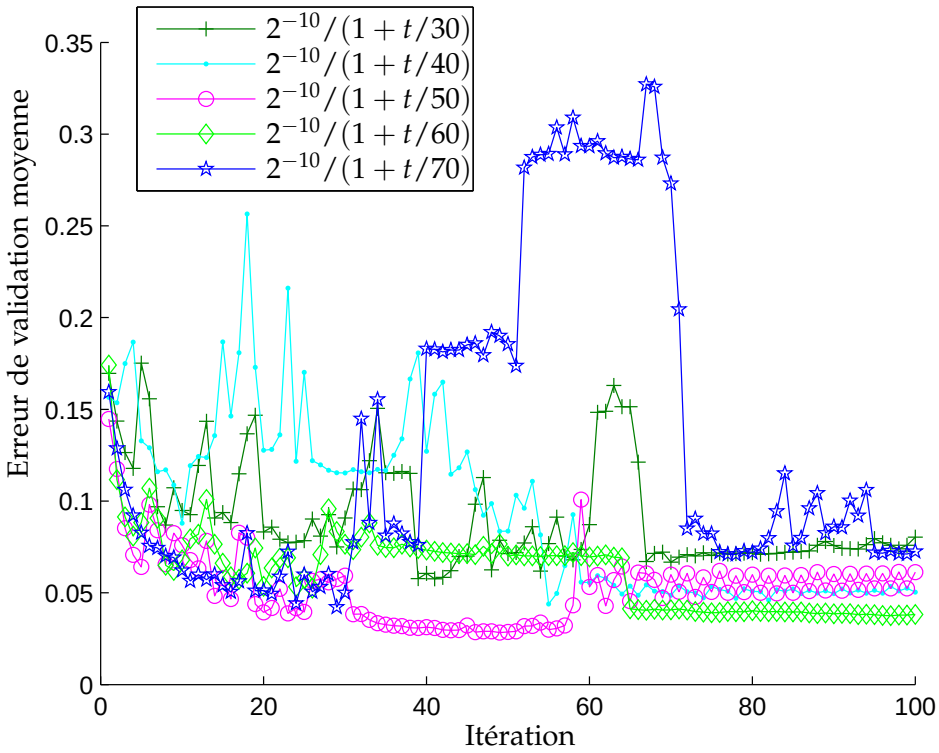

FIGURE 2.6 – *Evolution de l'erreur de validation moyenne pour différentes valeurs du paramètre τ du pas dans le cas du scénario A.*

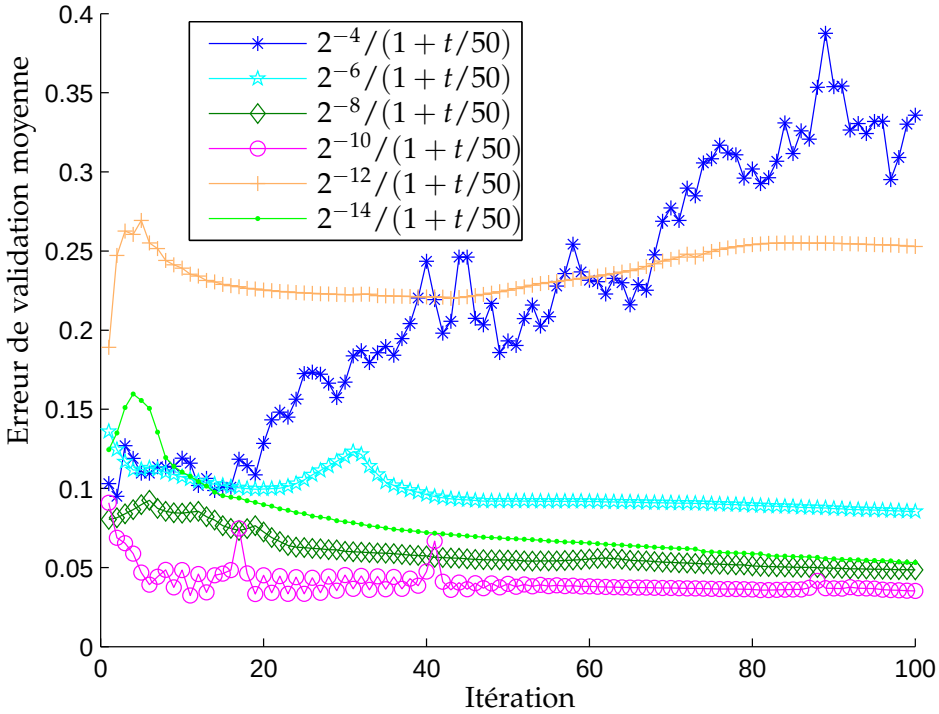

FIGURE 2.7 – *Evolution de l'erreur de validation moyenne pour différentes valeurs du paramètre η_0 du pas dans le cas du scénario B.*

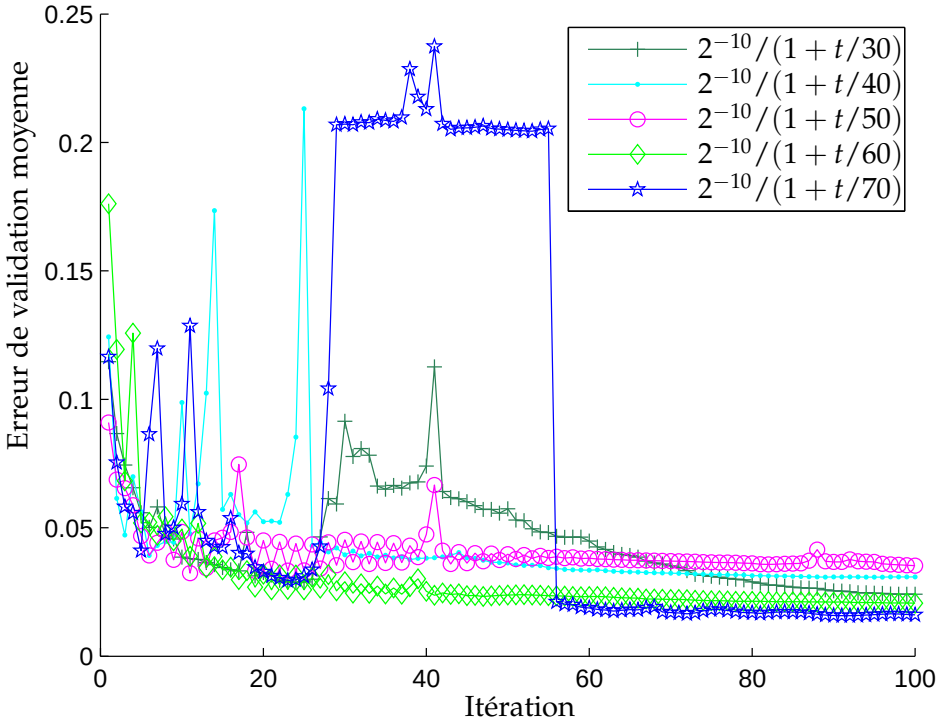

FIGURE 2.8 – *Evolution de l'erreur de validation moyenne pour différentes valeurs du paramètre τ du pas dans le cas du scénario B.*

régie par les trois méthodes d'optimisation proposées pour les scénarios A et B respectivement. Ces figures montrent la pertinence de la mobilité pour diminuer l'erreur. En outre, la méthode de Newton fonctionne mieux que la descente de gradient et du point fixe. Elles montrent aussi que la mobilité aléatoire est surpassée par l'algorithme du gradient et la méthode de Newton. Le scénario B est comparé avec l'algorithme de Lu *et al.* puisque dans les deux stratégies, il s'agit d'un modèle global et d'une mobilité de tous les capteurs. La FIGURE 2.10 montre que l'algorithme du gradient et la méthode de Newton surpassent l'algorithme de Lu *et al.*. D'autre part, nous remarquons que la méthode de Newton dans le scénario B atteint une erreur plus petite que celle dans le scénario A. Ainsi, le modèle global, en utilisant les informations de tout le réseau, surpasse le modèle local qui considère les informations du voisinage seulement. Notons que le modèle local a une complexité de calcul et de stockage plus petite ainsi qu'une plus petite consommation d'énergie en termes de communication, comparée à une approche globale qui nécessite la transmission des informations à un centre de fusion.

Pour le scénario C, nous ajoutons 5 capteurs robots déployés aléatoirement. A chaque itération, le robot qui a la plus grande erreur de validation se déplace. La FIGURE 2.11 et la FIGURE 2.12 montrent l'évolution, à chaque mouvement, de l'erreur de validation, pour plusieurs combinaisons des valeurs des paramètres η_0 et τ du pas dans le cas de l'algorithme de gradient. Ces figures montrent l'intérêt de prendre $\eta_0 = 2^{-10}$ et $\tau = 30$. Nous rappelons que nous avons intérêt à diminuer η_t lorsque nous approchons

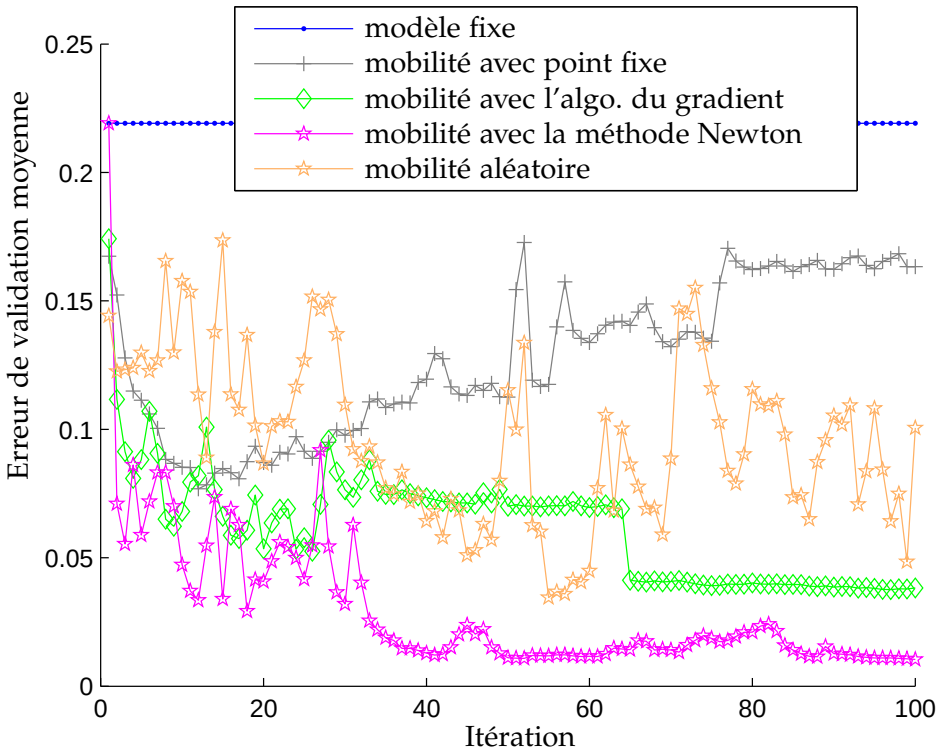

FIGURE 2.9 – *Performances des méthodes de mobilité avec le scénario A.*

de la solution finale. Or à partir de $t = \tau$, la valeur de η_t commence à diminuer significativement. Donc, d'après les valeurs de τ, nous constatons que la convergence du scénario C est plus rapide que celle des scénarios A et B. Afin de comparer avec l'algorithme de Lambrou *et al.*, nous devons déterminer les valeurs des paramètres cités dans la section "Stratégie de Lambrou *et al.*", page 18. Motivés par (Lambrou et al. 2010), nous prenons $dx = 100$, $\omega_1 = \omega_2 = 0.5$ et $\phi = 25°$. La FIGURE 2.13 montre les performances des différentes stratégies de mobilité, en les comparant au modèle statique, algorithme de Lambrou *et al.*, mobilité aléatoire et mobilité par balayage. Nous remarquons que l'algorithme du gradient et la méthode de Newton ont presque les mêmes performances. Alors que la méthode du point fixe atteint la même erreur, elle demeure la moins stable. En comparant ce scénario avec les deux autres, A et B, nous trouvons que l'erreur pour le scénario C est plus petite. En effet, l'augmentation de l'ensemble d'apprentissage à chaque itération avec des robots mobiles permet d'augmenter la précision de l'estimation. De plus, le scénario C coûte moins cher puisque le nombre de capteurs mobiles (robots) utilisés est beaucoup plus faible : il s'agit de seulement 5 capteurs mobiles, contre 100 capteurs mobiles dans les scénarios A et B.

La TABLE 2.1 montre la moyenne des distances parcourues de tous les capteurs mobiles, ainsi que la plus grande distance. Il est facile de voir que la méthode de point fixe fournit de grandes variations, principalement en raison de l'absence d'un paramètre de pas pour contrôler la mobilité. La méthode de Newton a, en moyenne, une meilleure efficacité. Cependant, ce procédé a un coût de calcul plus importante par rapport aux autres.

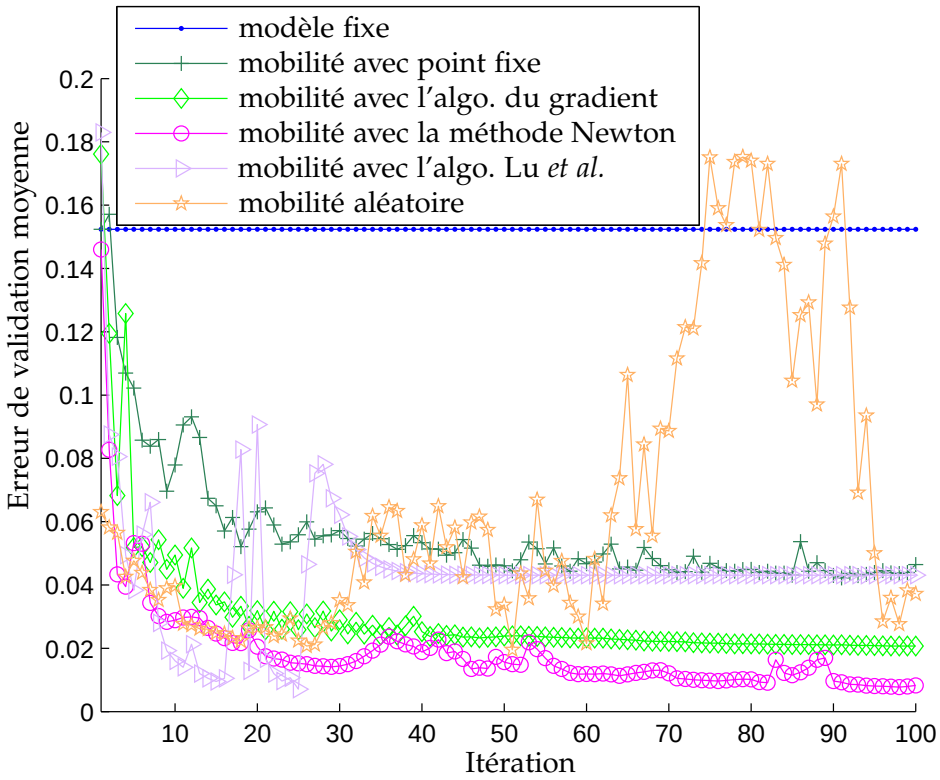

FIGURE 2.10 – *Performances des méthodes de mobilité avec le scénario B.*

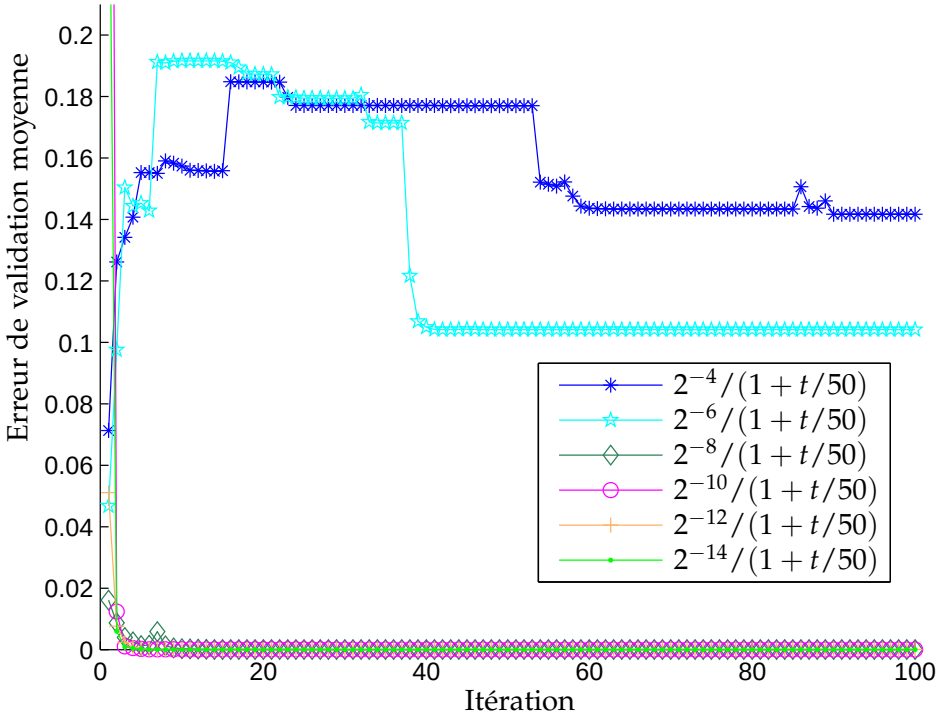

FIGURE 2.11 – *Evolution de l'erreur de validation moyenne pour différentes valeurs du paramètre eta$_0$ du pas dans le cas du scénario C.*

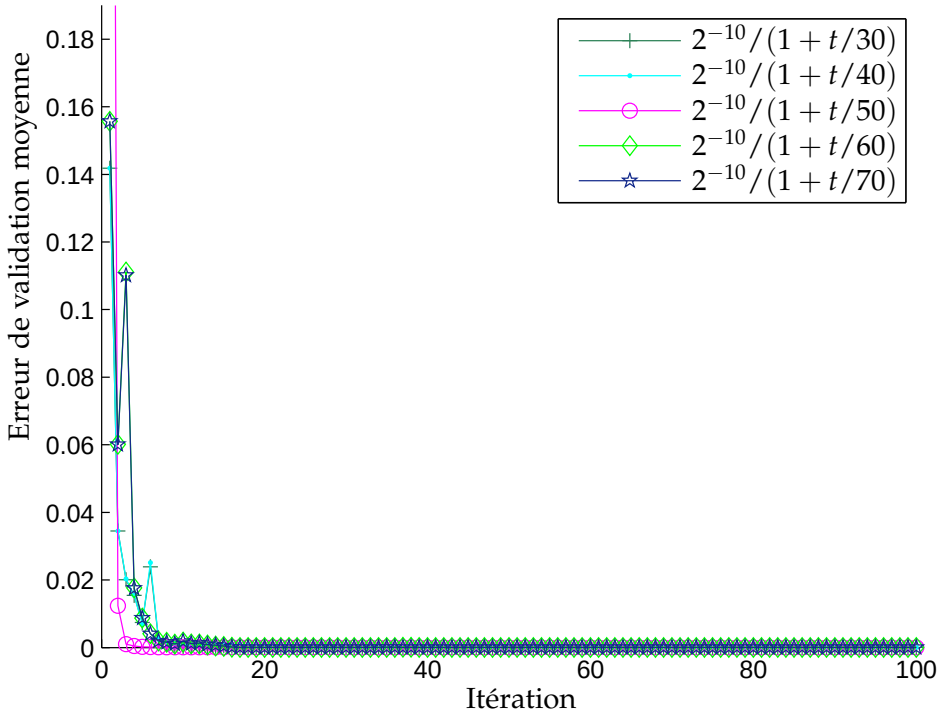

FIGURE 2.12 – *Evolution de l'erreur de validation moyenne pour différentes valeurs du paramètre τ du pas dans le cas du scénario C.*

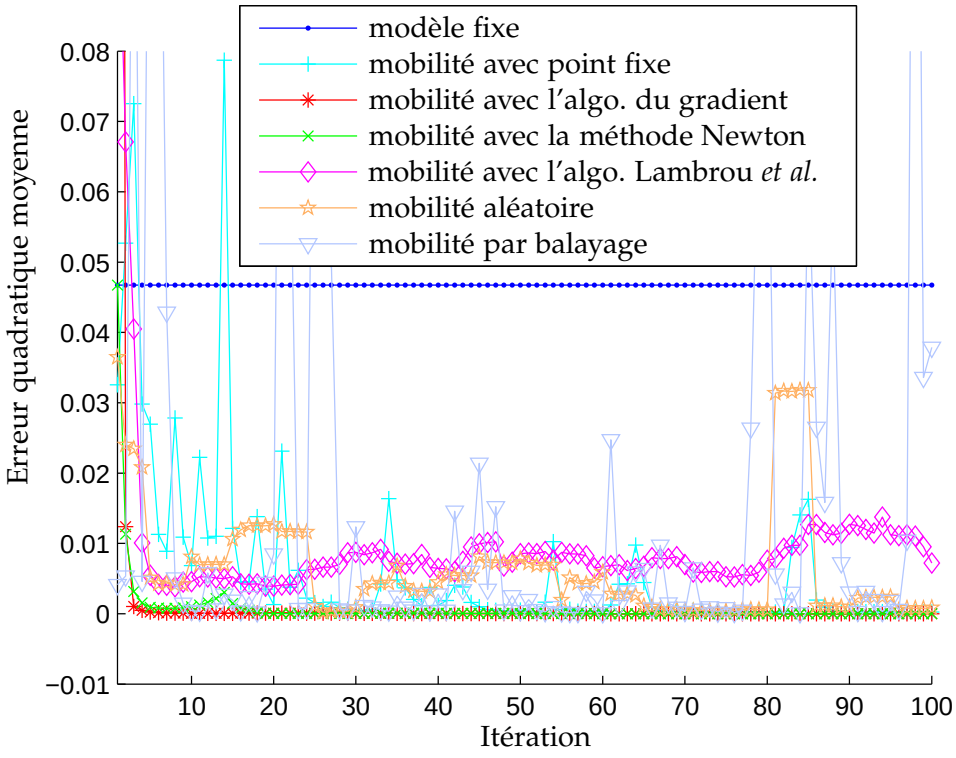

FIGURE 2.13 – *Performances des méthodes de mobilité avec le scénario C.*

TABLE 2.1 – *Distances moyenne et maximale parcourues par les capteurs mobiles.*

	Stratégie de mobilité :	Point fixe	Gradient	Newton
Scénario A	Distance moyenne :	0.0744	0.0449	0.0370
	Distance maximale :	0.4909	0.2681	0.1694
Scénario B	Distance moyenne :	0.0622	0.0439	0.0384
	Distance maximale :	0.3348	0.2232	0.2146
Scénario C	Distance moyenne :	0.2981	0.0343	0.0171
	Distance maximale :	0.3729	0.0644	0.0460

Par conséquent, nous pouvons conclure que, en fonction de l'application en question et des contraintes imposées, nous pouvons choisir l'une ou l'autre de ces méthodes.

Mobilité de plusieurs capteurs

Comme nous avons indiqué, le déplacement simultané de tous les capteurs mobiles est un problème insoluble, notamment lorsqu'il s'agit d'un réseau de capteurs à grande échelle. Mais rien n'empêche à déplacer un petit nombre de capteurs mobiles dans le but d'accélérer le processus de couvrir l'espace. Pour cela, nous avons augmenté le nombre des capteurs pouvant se déplacer à chaque itérations à 4 capteurs. La FIGURE 2.14, la FIGURE 2.15 et la FIGURE 2.16 montrent les performances de la mobilité d'un seul et de quatre capteurs mobiles, avec l'algorithme de gradient et la méthode de Newton, pour les scénarios A, B et C respectivement. Ces figures montrent que lorsque 4 capteurs se déplacent simultanément à chaque itération, l'erreur d'apprentissage diminue et la convergence est plus stable, notamment pour les scénarios A et B. Un autre critère de comparaison est le nombre de pas requis avant d'atteindre la stabilité. Ce paramètre est directement lié à l'énergie dépensée par les capteurs. Prenons par exemple le cas de la mobilité avec l'algorithme du gradient dans le scénario A. Un seul capteur mobile atteint la stabilité après environ 60 itérations, ce qui est équivalent à 60 déplacements, alors que 4 capteurs mobiles atteignent la stabilité après environ 35 itérations, équivalent à $35 \times 4 = 140$ pas. Nous constatons que l'énergie dépensée avec 4 capteurs est plus grande, même si le nombre d'itérations est plus petit. Cette observation est valable encore pour la méthode de Newton. Dans le scénario B, les deux cas d'un seul capteur mobile et de 4 capteurs mobiles ont besoin essentiellement du même nombre d'itérations pour atteindre la stabilité. Ce qui induit une plus grande énergie dépensée avec 4 capteurs mobiles. Dans le scénario C, les deux cas ont presque les mêmes performances. En conclusion, un seul capteur mobile à chaque itération est plus efficace en terme d'énergie dépensée.

CONCLUSION DU CHAPITRE

Dans le cadre des méthodes à noyaux en apprentissage statistique, nous avons modélisé dans ce chapitre un champ de diffusion de gaz mesuré par un réseau de capteurs mobiles. Ces modèles sont déterminés au moyen d'un processus d'apprentissage non linéaire en utilisant le noyau

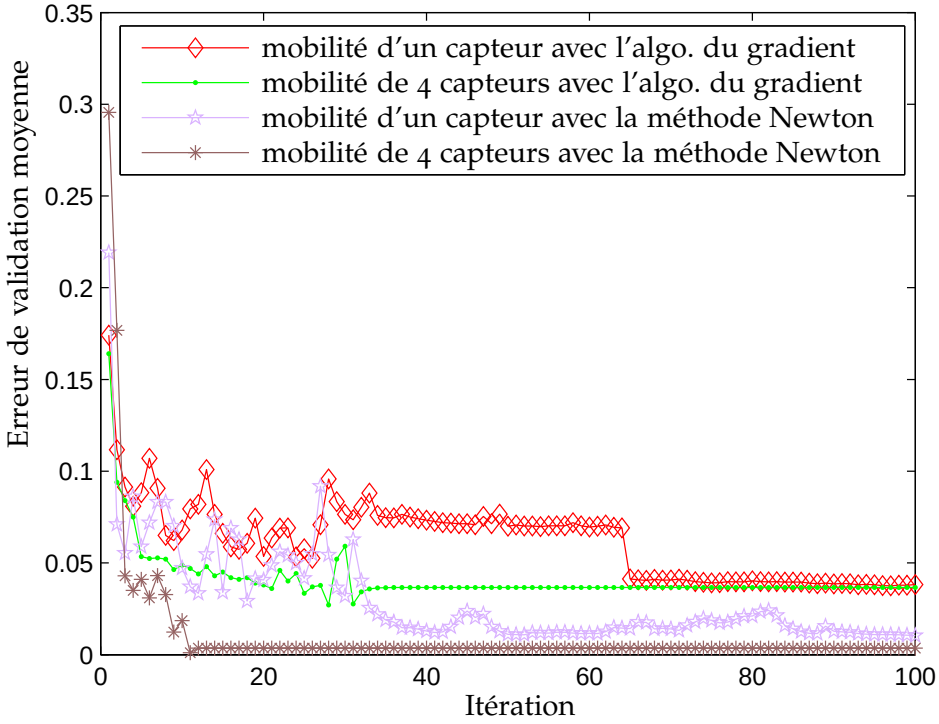

FIGURE 2.14 – *Performances des méthodes de mobilité avec le scénario A pour différents nombre de capteur mobiles par itération.*

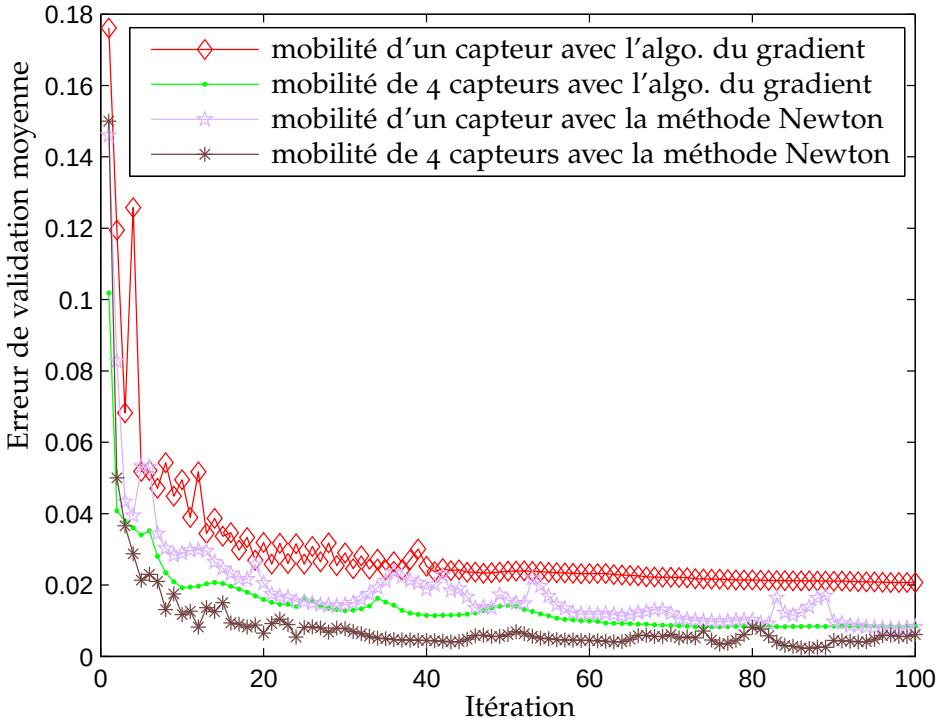

FIGURE 2.15 – *Performances des méthodes de mobilité avec le scénario B pour différents nombres de capteurs mobiles par itération.*

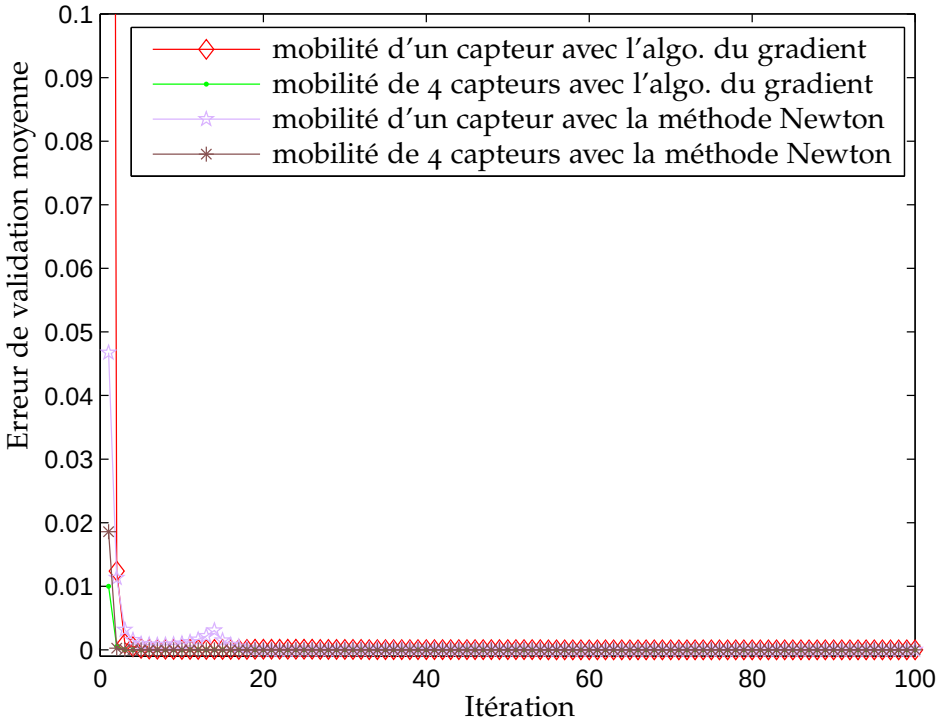

FIGURE 2.16 – *Performances des méthodes de mobilité avec le scénario C pour différents nombre de capteur mobiles par itération.*

gaussien. En proposant différents scénarios, nous avons optimisé les modèles résultants en déplaçant des capteurs mobiles dans la région à surveiller. Trois scénarios ont été envisagés. Dans le premier et le deuxième scénario, tous les capteurs peuvent se déplacer afin d'améliorer les modèles locaux et le modèle global respectivement. Dans le troisième scénario, une partie des capteurs est mobile, et cherche les informations pour adapter l'ensemble d'apprentissage formé initialement par les capteurs fixes. Il est à noter que dans les trois scénarios un seul capteur se déplace à chaque itération. Le capteur mobile se déplace vers la région la moins couverte selon un schéma d'optimisation. Nous avons considéré trois schémas d'optimisation du premier et second ordre : la méthode du point fixe, l'algorithme du gradient et la méthode de Newton. Nous avons montré la pertinence des méthodes proposées en termes d'erreur de validation, par rapport au cas statique et l'état de l'art. Nous avons montré encore que le troisième scénario a surpassé les deux premiers en termes d'erreur et du prix et que la méthode de Newton a donné la plus petite erreur, mais au prix d'une plus grande complexité de calcul.

Deuxième partie

Analyse en composantes principales pour les réseaux de capteurs sans fil

Introduction

Sommaire

Dans un réseau formé de plusieurs nœuds, chaque nœud collecte des données, le plus souvent de grande taille. Une communication entre les nœuds consiste à transmettre et recevoir des informations basées sur ces données collectées. Pour répondre aux contraintes de communication, stockage et traitement de ces informations, il est nécessaire de réduire la dimensionnalité de ces données tout en conservant les informations pertinentes. L'extraction d'informations pertinentes à partir des données multivariées est une tâche difficile. Cette idée émerge dès 1973 avec le théorème de codage de Slepian-Wolf (Slepian et Wolf 1973), qui révèle que deux sources corrélées peuvent être décodées conjointement avec une faible probabilité d'erreur. L'idée de Slepian-Wolf peut être implémentée dans une approche de compression avec ou sans perte (Wyner et Ziv 1976). L'intérêt de ces approches a souvent été soulevé dans la littérature du codage distribué de l'information dans les réseaux de capteurs sans fil (Chou et al. 2003, Barcelo-Llado et al. 2012).

Dans le présent travail, nous nous concentrons sur les approches avec perte utilisées pour des applications qui n'ont pas besoin d'une représentation très précise des données ; c'est le cas souvent dans les réseaux de capteurs sans fil, où l'information acquise est redondante aussi bien dans ses dimensions spatiale que temporelle. De ce fait, la perte d'une faible quantité de données peut ne pas pénaliser de façon conséquente le fonctionnement des RCSF. Les approches avec perte peuvent être classées selon les outils mathématiques qu'elles utilisent. On retrouve des stratégies qui utilisent le clustering pour quantifier les données disponibles, en utilisant différentes fonctions d'agrégation (Krishnamachari et al. 2002, Al-Karaki et al. 2009, Yuan et al. 2014). On retrouve également la classe de stratégies fondées sur l'analyse en composantes principales (ACP) qui permet d'identifier un sous-espace pertinent pour représenter les données (Jolliffe 1986, Abdi et Williams 2010). L'ACP est probablement la plus largement utilisée en apprentissage statistique, reconnaissance des formes,

compression et réduction de dimension. On l'appelle aussi la transformation discrète de Karhunen-Loève en traitement du signal et la transformée de Hotelling en analyse multivariée. Le domaine d'application de la technique ACP est très large. Ses applications concernent l'analyse multivariée, par exemple avec la validation des données, la classification multi-label, la détection de pannes, et le contrôle de qualité (Wise et al. 1989, MacGregor 1989). Elle est très utile dans le traitement du signal et des images (Geladi et al. 1989, Loughlin 1991, Honeine 2012). Par ailleurs, l'ACP fournit un outil puissant dans le cadre de réseaux de capteurs. Elle est étudiée pour l'extraction de caractéristiques à partir d'échantillons bruités (Chitradevi et al. 2011), et pour la compression et le débruitage des séries temporelles de mesures (Chen et al. 2010, Rooshenas et al. 2010), ainsi que pour la détection d'intrusion (Ahmadi Livani et Abadi 2011) ou d'anomalie (Huang et al. 2006).

L'ACP consiste à déterminer un sous-espace qui retient la plus grande variance des données. Ce sous-espace est engendré par les vecteurs propres associés aux plus grandes valeurs propres de la matrice de covariance des données. L'algorithme classique de l'ACP nécessite la décomposition en éléments propres de la matrice de covariance des données. Or, la complexité de calcul d'une telle opération est cubique avec la taille de l'ensemble de données. Plusieurs techniques itératives ont été proposées pour calculer les vecteurs propres d'une matrice, parmi lesquelles nous citons les travaux de Gill et al. (1974), Bunch et Nielsen (1978) et Hall et al. (2000). Cependant le coût de calcul de ces techniques reste élevé. Un autre défit doit être confronté dans les réseaux de capteurs. En effet, l'envoi de toutes les données acquises par les capteurs à un centre de fusion est inapproprié pour les réseaux de grande taille. De récents travaux ont été menés pour la mise en oeuvre de l'ACP dans un réseau, comme décrit dans la suite.

Dans la suite de cette introduction, nous présentons une description du réseau. Après avoir rappelé l'algorithme classique de l'ACP, nous décrivons les algorithmes proposés dans la littérature des réseaux de capteurs.

Description du réseau

Considérons un réseau de N nœuds capteurs connectés (également appelés *agents* dans la littérature). Chaque nœud est capable d'échanger des données avec d'autres nœuds. Nous supposons que deux nœuds sont nécessairement connectés, soit directement s'ils sont voisins, ou par d'autres nœuds intermédiaires. Dans la littérature, il existe principalement deux types de réseaux : réseaux centralisés et réseaux décentralisés ou distribués. Dans le premier cas, les nœuds sont connectés à un centre de fusion (CF) ; dans le second cas, les nœuds sont connectés, soit de manière non coopérative par un chemin de routage donné, soit d'une façon coopérative où chaque nœud communique avec ses nœuds voisins. Nous notons $\mathcal{V}_k$ l'ensemble des indices des nœuds voisins au nœud k, c'est-à-dire les nœuds qui y sont directement connectés. Nous considérons que le nœud k est adjacent à lui-même, c'est-à-dire $k \in \mathcal{V}_k$.

Soit $\boldsymbol{y}_k$ le vecteur de taille $(p \times 1)$ des données recueillies par le nœud k, pour $k = 1, 2, \cdots, N$, où p est le nombre de mesures acquises par chaque nœud. Soit $\mathbb{Y} \subset \mathbb{R}^p$ l'espace de ces données recueillies, avec le produit scalaire conventionnel $\boldsymbol{y}_k^\top \boldsymbol{y}_l$ pour tout $\boldsymbol{y}_k, \boldsymbol{y}_l \in \mathbb{Y}$. Soit $\boldsymbol{Y}$ la matrice de taille $(N \times p)$ de l'ensemble des données, c'est-à-dire $\boldsymbol{Y} = [\boldsymbol{y}_1 \;\; \boldsymbol{y}_2 \;\; \cdots \;\; \boldsymbol{y}_N]^\top$. On suppose que chaque colonne de $\boldsymbol{Y}$ est de moyenne nulle ; si ce n'est pas le cas, on soustrait la moyenne des données d'une manière centralisée ou décentralisée comme décrit dans la section 4.2.1. On désigne par $z_{\boldsymbol{w},k} = \boldsymbol{w}^\top \boldsymbol{y}_k$ le produit scalaire associé à la projection orthogonale de tout $\boldsymbol{y}_k \in \mathbb{Y}$ sur le vecteur $\boldsymbol{w} \in \mathbb{Y}$ et par $z_{\boldsymbol{w}}$ la variable aléatoire prenant les valeurs $z_{\boldsymbol{w},k}$, avec $k = 1, ..., N$. Selon la topologie du réseau étudié, le but ultime est de déterminer l'axe principal $\boldsymbol{w}$ (ou les axes principaux), avec optimalité au sens du vecteur propre associé à la plus grande valeur propre de la matrice de covariance des données $\boldsymbol{y}_k$, pour $k = 1, 2, \cdots, N$, et en conséquence d'utiliser ensuite seulement $z_{\boldsymbol{w},k}$ pour représenter efficacement les données $\boldsymbol{y}_k$. Dans ce qui suit, nous présentons les méthodes dans la littérature pour l'estimation des axes principaux de l'ACP dans les réseaux de capteurs, en commençant par rappeler l'algorithme classique.

Etat de l'art

Avant de décrire en détail dans les chapitres suivants les méthodes que nous proposons pour l'estimation des axes principaux de l'ACP d'une manière décentralisée, nous présentons dans cette section l'algorithme classique de l'ACP et les algorithmes proposés dans la littérature pour sa mise en œuvre dans les réseaux. Ces algorithmes peuvent être classés en deux catégories, centralisés et décentralisés. Dans le chapitre suivant, afin d'avoir une comparaison juste, c'est-à-dire dans les mêmes conditions, nous comparons nos méthodes avec les algorithmes décentralisés décrits dans cette section.

Algorithme classique de l'ACP

Appliquée dans le cadre des réseaux de capteurs, il s'agit d'une stratégie centralisée, où les nœuds sont supposés être connectés à un centre de fusion (CF). Ce dernier reçoit les données recueillies sans aucune transformation locale par les nœuds. Afin d'extraire le premier axe principal désigné par $\boldsymbol{w}$, le CF cherche à maximiser la variance des données projetées, à savoir

$$\max_{\boldsymbol{w}} \mathbb{E}(z_{\boldsymbol{w}}^2)$$

sous contrainte de norme unité, où $\mathbb{E}(\cdot)$ désigne l'espérance selon la distribution des données. En considérant l'estimation empirique sur les échantillons disponibles, $\boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_N$, nous obtenons

$$\max_{\boldsymbol{w}} \boldsymbol{w}^\top \mathfrak{C} \boldsymbol{w},$$

où $\mathfrak{C} = \frac{1}{N} \sum_{k=1}^{N} \boldsymbol{y}_k \boldsymbol{y}_k^\top$ est la matrice de covariance des échantillons. En combinant le théorème min-max de Courant-Fischer au quotient de Rayleigh

(ou encore en utilisant les multiplicateurs de Lagrange), il est bien connu que le problème d'optimisation devient

$$\mathfrak{C}\boldsymbol{w} = \lambda \boldsymbol{w}.$$

Il s'agit du problème de la décomposition en éléments propres où $\boldsymbol{w}$ est le vecteur propre associé à la valeur propre λ de $\mathfrak{C}$. De plus, cette valeur propre λ n'est autre que la variance correspondante aux échantillons projetés sur $\boldsymbol{w}$, étant donné que

$$\frac{1}{N}\sum_{k=1}^{N}\left(\boldsymbol{w}^\top \boldsymbol{y}_k\right)^2 = \boldsymbol{w}^\top \mathfrak{C}\boldsymbol{w} = \boldsymbol{w}^\top \lambda \boldsymbol{w} = \lambda \boldsymbol{w}^\top \boldsymbol{w} = \lambda.$$

Par conséquent, l'axe principal le plus pertinent est le vecteur propre associé à la plus grande valeur propre de $\mathfrak{C}$. En outre, le $\ell^{\text{ième}}$ axe principal le plus pertinent est le vecteur propre $\boldsymbol{w}_\ell$ associé à la $\ell^{\text{ième}}$ plus grande valeur propre λ_ℓ de $\mathfrak{C}$.

Ayant les données, $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N$, le CF résout le problème de la décomposition en éléments propres. La complexité de calcul de la matrice de covariance est en $\mathcal{O}(Np^2)$, et elle nécessite $\mathcal{O}(p^2)$ unités de stockage en mémoire. La complexité de calcul du problème de décomposition en éléments propres est de $\mathcal{O}(p^3)$. A ces coûts, il faut aussi inclure le coût de la communication entre les nœuds et le CF, qui est en $\mathcal{O}(Np)$ sur une distance $\mathcal{O}(1)$. Dans ce qui suit, nous notons $\boldsymbol{w}_*$ l'axe principal le plus pertinent obtenu par l'algorithme classique de l'ACP, et $\boldsymbol{w}_{*k}$ le $k^{\text{ième}}$ axe principal le plus pertinent associé à la $k^{\text{ième}}$ plus grande valeur propre λ_k de $\mathfrak{C}$. Différents algorithmes ont été proposés afin d'estimer $\boldsymbol{w}_*$ avec une faible complexité de calcul et de stockage.

Méthode de la puissance itérée

Rappelons que si $\boldsymbol{w}_*$ est le vecteur propre associé à la valeur propre λ de $\mathfrak{C}$, alors $\mathfrak{C}\boldsymbol{w}_* = \lambda \boldsymbol{w}_*$, et en général, $\mathfrak{C}^k \boldsymbol{w}_* = \lambda^k \boldsymbol{w}_*$ pour tout $k \in \mathbb{N}^+$. Cette observation est la base de la méthode de la puissance itérée, réexaminée dans le cadre des réseaux de capteurs sans fil dans (Le Borgne et al. 2008). En effet, l'ensemble $\{\boldsymbol{w}_{*1}, \cdots, \boldsymbol{w}_{*p}\}$ des vecteurs propres de $\mathfrak{C}$ constitue une base de $\mathbb{R}^p$. Soit $\boldsymbol{v}_0$ une initialisation pour approximer le premier vecteur propre de $\mathfrak{C}$, avec $\|\boldsymbol{v}_0\| = 1$. Alors $\boldsymbol{v}_0$ s'écrit sous la forme d'une combinaison linéaire des vecteurs propres de $\mathfrak{C}$ avec des coefficients $c_1, \cdots, c_p \in \mathbb{R}$:

$$\boldsymbol{v}_0 = c_1 \boldsymbol{w}_{*1} + \cdots + c_p \boldsymbol{w}_{*p}.$$

Nous supposons pour la suite que $c_1 \neq 0$. En multipliant (à gauche) les deux côtés de l'égalité par $\mathfrak{C}$, nous obtenons :

$$\mathfrak{C}\boldsymbol{v}_0 = c_1\lambda_1 \boldsymbol{w}_{*1} + c_2\lambda_2 \boldsymbol{w}_{*2} + \cdots + c_p\lambda_p \boldsymbol{w}_{*p},$$

ou encore en les multipliant k fois par $\mathfrak{C}$,

$$\begin{aligned}\mathfrak{C}^k\boldsymbol{v}_0 &= c_1\lambda_1^k \boldsymbol{w}_{*1} + c_2\lambda_2^k \boldsymbol{w}_{*2} + \cdots + c_p\lambda_p^k \boldsymbol{w}_{*p} \\ &= \lambda_1^k\Big(c_1 \boldsymbol{w}_{*1} + c_2\Big(\frac{\lambda_2}{\lambda_1}\Big)^k \boldsymbol{w}_{*2} + \cdots + c_p\Big(\frac{\lambda_p}{\lambda_1}\Big)^k \boldsymbol{w}_{*p}\Big).\end{aligned}$$

Puisque les valeurs propres sont supposées réelles, distinctes, et par ordre décroissant, il en résulte que pour tout $i = 2, \cdots, p$,

$$\lim_{k\to\infty} \left(\frac{\lambda_i}{\lambda_1}\right)^k = 0.$$

Ceci implique que plus k augmente, plus $\mathfrak{C}^k \boldsymbol{v}_0$ s'approche de $c_1\lambda_1^k \boldsymbol{w}_{*1}$. Ainsi, pour les grandes valeurs de k, nous avons :

$$\boldsymbol{w}_{*1} \approx \frac{\mathfrak{C}^k \boldsymbol{v}_0}{\|\mathfrak{C}^k \boldsymbol{v}_0\|}.$$

La méthode de la puissance itérée consiste alors à choisir aléatoirement d'abord un vecteur $\boldsymbol{v}_0$, puis à chaque itération t, la règle de mise à jour suivante est appliquée :

$$\boldsymbol{v}_t = \frac{\mathfrak{C}\boldsymbol{v}_{t-1}}{\|\mathfrak{C}\boldsymbol{v}_{t-1}\|}.$$

Les critères de convergence peuvent être définis soit avec une variation minimale δ de $\boldsymbol{v}_t$, soit avec un plus grand nombre d'itérations $t_{\max}$. La méthode de la puissance itérée est simple et élégante. Cependant, cette méthode ne retourne que l'estimation du vecteur propre correspondant à la valeur propre de plus grande amplitude (premier axe principal seulement). En outre, la convergence est garantie uniquement si les valeurs propres sont distinctes, et en particulier les deux valeurs propres (de plus grande valeur absolue) doivent avoir grandeurs distinctes. Le taux de convergence dépend principalement du rapport de ces grandeurs. En effet, si les deux plus grandes valeurs propres sont proches, alors la convergence sera lente.

Algorithme de Korada *et al.*

La méthode de la puissance itérée utilise la matrice de covariance $\mathfrak{C}$ pour calculer itérativement l'axe principal. Une idée simple afin de diminuer la complexité de calcul (le nombre de multiplications et d'additions) est d'utiliser des matrices creuses. On dit qu'une matrice $\boldsymbol{S}$ de taille $(N \times N)$ est une matrice creuse de $\mathfrak{C}$ si elle est obtenue en annulant quelques éléments de $\mathfrak{C}$ et en redimensionnant les éléments non nuls. Korada et al. (2011) a proposé un algorithme en combinant le principe de la matrice creuse avec un algorithme de type gossip. Soit d le paramètre désignant le nombre d'éléments non nuls par ligne de la matrice creuse. A chaque itération t, une matrice creuse de $\mathfrak{C}$, désignée par $\boldsymbol{S}_t$, est obtenue en annulant chaque élément de $\mathfrak{C}$ avec une probabilité $1 - d/N$. Puis, les éléments non nuls sont redimensionnés par un facteur N/d. En initialisant avec un vecteur aléatoire $\boldsymbol{v}_0$, le vecteur $\boldsymbol{v}_t$ est obtenu selon la mise à jour :

$$\boldsymbol{v}_t = \boldsymbol{S}_t \boldsymbol{v}_{t-1}.$$

Une normalisation de $\boldsymbol{v}_t$ est nécessaire pour un vecteur de norme unité, c'est-à-dire $\frac{\boldsymbol{v}_t}{\|\boldsymbol{v}_t\|}$. L'estimation du premier axe principal à l'itération t est :

$$\boldsymbol{w}_t = c(t) \sum_{s=1}^{t} \frac{\boldsymbol{v}_s}{\|\boldsymbol{v}_s\|},$$

avec $c(t)$ un paramètre qui assure $\|\boldsymbol{w}_t\| = 1$. A noter que les normes sont calculées par un algorithme de type gossip. Voir Section 4.2 pour plus de détail sur les algorithmes de type gossip.

ACP collective

Le principal verrou de la complexité de calcul de l'ACP provient principalement de la grande taille des données à traiter, puisque la complexité de calcul de la décomposition en éléments propres est en $\mathcal{O}(p^3)$. Pour contourner ce problème, une idée est de diviser l'ensemble des dimensions (caractéristiques) des données en plusieurs sous-ensembles. Ce principe est utilisé dans l'ACP collective, proposée par Kargupta et al. (2000). Cet algorithme opère essentiellement en deux étapes, la matrice $\boldsymbol{Y}$ étant divisée en d sous-matrices :

$$\boldsymbol{Y} = [\boldsymbol{Y}_1 \;\; \boldsymbol{Y}_2 \;\; \cdots \;\; \boldsymbol{Y}_d],$$

où $\boldsymbol{Y}_i$ est une sous-matrice de taille $(N \times p_i)$ de $\boldsymbol{Y}$ et $\sum_{i=1}^{d} p_i = p$. L'ACP est effectuée sur les données de chaque sous-matrice $\boldsymbol{Y}_i$. Soient $\boldsymbol{W}^{(i)}$ la matrice de taille $(p_i \times r_i)$ des r_i premiers axes principaux, et $\boldsymbol{W}^{tot}$ la matrice diagonale par bloc avec les matrices $\boldsymbol{W}^{(1)}, \cdots, \boldsymbol{W}^{(d)}$ sur la diagonale. Les composantes principales correspondant aux données de $\boldsymbol{Y}_i$, notées $\boldsymbol{Z} = \boldsymbol{Y}_i\boldsymbol{W}^{(i)}$, sont recueillies dans une matrice $\boldsymbol{Z} = [\boldsymbol{Z}_1 \;\; \boldsymbol{Z}_2 \;\; \cdots \;\; \boldsymbol{Z}_d]$ de taille $(N \times r)$, où $r = \sum_{i=1}^{d} r_i$. Une ACP sur $\boldsymbol{Z}$ est effectuée afin d'extraire ses axes principaux $\boldsymbol{v}_i$. Enfin, les axes principaux de $\boldsymbol{Y}$ sont :

$$\boldsymbol{w}_i = \boldsymbol{W}^{tot}\,\boldsymbol{v}_i.$$

Approche distribuée basée sur l'ACP

L'ACP collective réduit la complexité de calcul de l'ACP au prix d'utiliser un centre de fusion, ce qui rend cet algorithme moins attractif pour les réseaux à grande échelle. Récemment, Ahmadi Livani et Abadi (2011) proposent de surmonter ce problème en divisant les nœuds du réseau en des clusters/groupes. L'approche de l'ACP distribuée (PCADID pour *PCA-based Distributed Intrusion Detection*) opère à travers un régime "diviser-pour-régner" afin de réduire la complexité de calcul du problème de décomposition en éléments propres. Elle est décrite comme suit. Les N nœuds sont divisés en d groupes de N_i nœuds, pour $i = 1, 2, \cdots, d$, où $\sum_{i=1}^{d} N_i = N$. Chaque groupe i traite p_i caractéristiques où $\sum_{i=1}^{d} p_i = p$. Dans ce cadre, l'algorithme de PCADID opère en plusieurs étapes, comme suit. D'abord, pour chaque groupe i, les données associées au groupe sont normalisées dans $[0\ , 1]$, puis la matrice centrée par colonne $\boldsymbol{Y}_i^c$ est calculée. Les axes principaux des données de la matrice $\boldsymbol{Y}_i^c$ sont extraits en utilisant une décomposition en valeurs singulières ou de façon équivalente la décomposition en éléments propres de la matrice de covariance correspondante. Chaque groupe transmet ces axes principaux aux groupes voisins, selon un processus de routage. Enfin, les axes principaux de $\boldsymbol{Y}$ sont obtenus par l'union des axes principaux de tous les groupes.

Late/Early gossip de l'ACP

Les algorithmes décrits ci-dessus traitent le problème du calcul des vecteurs propres de la matrice $\mathfrak{C}$. La matrice de covariance de l'ensemble (ou des sous-ensembles) des données doit être déterminée. Or le calcul de cette matrice et son stockage ont une grande complexité. Pour résoudre ce problème, il existe des méthodes pour estimer la matrice de covariance, telles les méthodes Late/Early gossip de l'ACP qui ont été proposées plus récemment par Fellus et al. (2014). L'algorithme Late gossip correspond à déterminer la matrice de covariance $\mathfrak{C}$ d'une manière itérative, en utilisant le gossip (principe détaillé dans le Chapitre 4, Section 4.2). A cette fin, à chaque nœud i est défini trois variables avec des valeurs initiales comme suit :

$$\begin{aligned} \boldsymbol{a}_i(0) &= \boldsymbol{Y}^\top \mathbb{1}_N, \\ \boldsymbol{B}_i(0) &= \boldsymbol{y}_i^\top \boldsymbol{y}_i, \\ m_i(0) &= p. \end{aligned}$$

Le vecteur $\mathbb{1}_N$ est le vecteur de taille $(N \times 1)$ dont les éléments sont égaux à un. A chaque itération t, un nœud émetteur k et un nœud récepteur l, choisis aléatoirement, appliquent la moyenne de chacune de ces trois variables. En utilisant ces résultats, nous obtenons l'estimateur de la matrice de covariance pour chaque nœud i :

$$\mathfrak{C}_i(t) = \frac{\boldsymbol{B}_i(t)}{m_i(t)} - \frac{\boldsymbol{a}_i(t)\boldsymbol{a}_i(t)^\top}{m_i(t)^2}.$$

Il s'avère que chaque $\mathfrak{C}_i(t)$ tend vers la matrice de covariance $\mathfrak{C}$. Après convergence, l'axe principal est déterminé localement à un nœud quelconque i par une décomposition en éléments propres de $\mathfrak{C}_i$.

L'algorithme Early gossip intègre une réduction de dimensionnalité dans les règles de mise à jour. En fait, la décomposition en éléments propres de la matrice $\boldsymbol{B}_i(0) = \boldsymbol{U}_i(0)\boldsymbol{L}_i(0)\boldsymbol{U}_i(0)^\top$ peut être obtenue en diagonalisant $\boldsymbol{y}_i\boldsymbol{y}_i^\top = \boldsymbol{V}_i(0)\boldsymbol{\Lambda}_i(0)\boldsymbol{V}_i(0)^\top$ qui a une plus petite taille, donc une plus petite complexité de calcul et de stockage. En effet, les deux équations :

$$\begin{aligned} \boldsymbol{B}_i(0)^2 &= \boldsymbol{y}_i^\top \boldsymbol{y}_i \boldsymbol{y}_i^\top \boldsymbol{y}_i \\ &= \boldsymbol{y}_i^\top \boldsymbol{V}_i(0)\boldsymbol{\Lambda}_i(0)\boldsymbol{V}_i(0)^\top \boldsymbol{y}_i \\ &= (\boldsymbol{y}_i^\top \boldsymbol{V}_i(0)\boldsymbol{\Lambda}_i(0)^{-\frac{1}{2}})\boldsymbol{\Lambda}_i(0)^2(\boldsymbol{y}_i^\top \boldsymbol{V}_i(0)\boldsymbol{\Lambda}_i(0)^{-\frac{1}{2}})^\top, \end{aligned}$$

et

$$(\boldsymbol{y}_i^\top \boldsymbol{V}_i(0)\boldsymbol{\Lambda}_i(0)^{-\frac{1}{2}})^\top(\boldsymbol{y}_i^\top \boldsymbol{V}_i(0)\boldsymbol{\Lambda}_i(0)^{-\frac{1}{2}}) = \boldsymbol{I}$$

impliquent :

$$\boldsymbol{U}_i(0) = \boldsymbol{y}_i^\top \boldsymbol{V}_i(0)\boldsymbol{\Lambda}_i(0)^{-\frac{1}{2}} \qquad et \qquad \boldsymbol{L}_i(0) = \boldsymbol{\Lambda}_i(0).$$

D'autre part, Fellus et al. (2014) démontrent que $\boldsymbol{U}_k(t)$ et $\boldsymbol{L}_k(t)$ à l'itération t, peuvent être obtenus par la décomposition économique QR, d'une

manière itérative :

$$(\boldsymbol{Q}(\tau+1), \boldsymbol{R}(\tau+1)) = \mathrm{QR}\Big(\boldsymbol{U}_k(t-1)\boldsymbol{L}_k(t-1)\boldsymbol{U}_k^\top(t-1)\boldsymbol{Q}(\tau) + \boldsymbol{U}_l(t-1)\boldsymbol{L}_l(t-1)\boldsymbol{U}_l^\top(t-1)\boldsymbol{Q}(\tau)\Big).$$

Ainsi $\boldsymbol{U}_k(t) = \boldsymbol{Q}(\infty)$ et $\boldsymbol{L}_k(t) = \mathrm{diag}(\boldsymbol{R}(\infty))$. Il s'avère que chaque matrice $\boldsymbol{U}_i$ converge vers la matrice des vecteurs propres associés aux plus grandes valeurs propres de la matrice $\mathfrak{C}$.

Plan de cette partie

Nous proposons une nouvelle classe d'algorithmes pour l'estimation des axes principaux de l'ACP dans les réseaux de capteurs sans fil. Nos stratégies d'estimation à la volée permettent d'éviter le calcul de la matrice de covariance des échantillons et sa décomposition en éléments propres. Ceci permet de réduire considérablement la complexité de calcul, la capacité de stockage en mémoire et l'énergie de communication.

Cette partie est formée de deux chapitres. Dans le premier chapitre, nous estimons l'axe principal dans un réseau, sans calcul de la matrice de covariance, en opérant itérativement et de manière distribuée la minimisation d'une fonction coût appropriée. Nous proposons des stratégies non coopératives et coopératives et présentons plusieurs fonctions coût possibles. Nous étudions la convergence des stratégies proposées. Enfin, nous illustrons la pertinence de notre travail sur l'application dans un réseau de capteurs sans fil tout en comparant avec les méthodes de l'état de l'art.

Dans le second chapitre, nous étendons notre étude pour l'estimation de plusieurs axes principaux. De plus, nous étudions en particulier deux problèmes omniprésents dans les réseaux. Le premier est la désynchronisation entre les nœuds dans un réseau décentralisé. Nous proposons une solution qui permet d'estimer les axes principaux dans un réseau asynchrone. Le second problème est la présence de bruit de mesure et de communication. La solution proposée avec un mécanisme de lissage permet d'atténuer l'effet du bruit. Les méthodes proposées sont enfin appliquées sur un réseau de capteurs sans fil pour la surveillance de la diffusion de gaz et le traitement distribué des images.

Analyse en composantes principales dans les RCSF

3

Sommaire

Dans ce chapitre, nous étudions le problème de l'estimation de l'axe principal le plus pertinent des données issues d'un réseau de capteur sans fil, tout en évitant le calcul de la matrice de covariance. A cette fin, nous proposons un cadre qui couple des critères pour estimer l'axe principal avec plusieurs stratégies pour le traitement en réseau. Pour cela, d'une part, nous étudions plusieurs critères, comme celui mis en œuvre par Oja (1982) et qui a été réexaminé récemment par Honeine (2012) pour l'ACP non linéaire avec des méthodes à noyaux. Nous examinons plusieurs fonctions coût, y compris un critère issu de la théorie de l'information présenté dans (Plumbley 1995), et les fonctions coût quotient de Rayleigh, LUO et OJAn présentées dans (Cirrincione et al. 2002, Luo et al. 1997). D'autre part, nous explorons ces critères pour le traitement distribué dans les réseaux. A cette fin, plusieurs stratégies en réseau sont proposées, y compris les stratégies non coopératives et coopératives. Dans les stratégies coopératives, les capteurs coopèrent ensemble par diffusion de l'information en vue d'estimer les axes principaux. Nous proposons deux stratégies coopératives, les stratégies combiner-puis-adapter et adapter-puis-combiner, qui

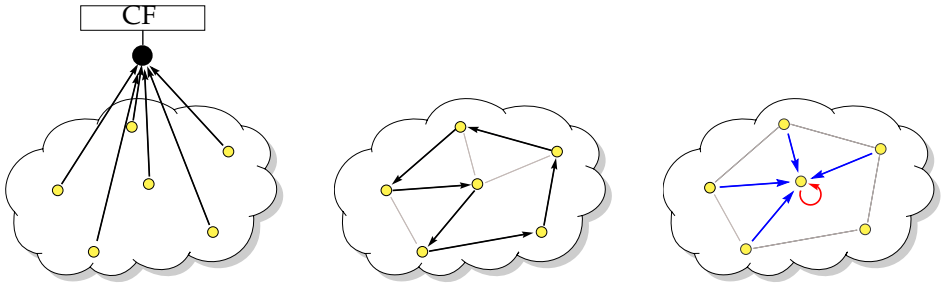

FIGURE 3.1 – *Illustration du traitement de l'information dans un réseau : Alors que le réseau centralisé nécessite un centre de fusion, les réseaux non coopératifs et coopératifs fournissent le traitement en réseau. Le réseau non coopératif traite l'information en utilisant un chemin de routage donné. Le réseau de coopération gère une diffusion de l'information, en utilisant soit une stratégie combiner-puis-adapter ou une stratégie adapter-puis-combiner, ainsi qu'une stratégie de consensus.*

ont été récemment étudiées dans la littérature du filtrage adaptatif linéaire par Sayed et al. (2013). Nous fournissons également des connexions à l'apprentissage par consensus. A notre connaissance, la présente étude est une première qui étudie ces stratégies d'adaptation pour l'apprentissage non supervisé, comme indiqué ici avec l'ACP. Pour fournir une présentation détaillée dans le présent document, nous étudions d'abord en détail l'estimation d'un unique axe principal pour une fonction coût donnée, avant de généraliser l'approche à différentes fonctions coût.

3.1 L'ACP EN RÉSEAUX

Dans cette section, nous décrivons les différentes stratégies en fonction de la topologie du réseau, comme illustré dans la FIGURE 3.1. La stratégie classique de l'ACP, présentée dans l'introduction de cette partie, s'effectue au sein d'un réseau centralisé. Nous détaillons nos nouvelles stratégies en réseaux décentralisés pour extraire l'axe principal. A la section 4.1 du chapitre suivant, nous généralisons ces techniques pour le cas de plusieurs axes principaux.

3.1.1 Stratégie non coopérative

Au lieu de maximiser directement la variance globale projetée et de calculer explicitement la matrice de covariance, nous considérons un système en réseau où l'axe principal est estimé d'une manière adaptative, à la volée. A cette fin, nous considérons une estimation "instantanée" de l'axe principal à chaque nœud. Selon un processus de routage prédéfini, chaque nœud k reçoit une estimation $\boldsymbol{w}_{t-1}$ d'un autre nœud, et l'ajuste en utilisant son propre vecteur de données $\boldsymbol{y}_k$ afin d'avoir une nouvelle estimation, désignée par $\boldsymbol{w}_t$. Dans ce but, nous minimisons l'erreur quadratique "instantanée" de reconstruction, à savoir $\boldsymbol{w}_t = \arg\min_{\boldsymbol{w}} J_k(\boldsymbol{w})$ avec

$$J_k(\boldsymbol{w}) = \tfrac{1}{4}\|\boldsymbol{y}_k - z_{\boldsymbol{w},k}\boldsymbol{w}\|^2, \tag{3.1}$$

où $z_{\boldsymbol{w},k} = \boldsymbol{w}^\top \boldsymbol{y}_k$ et le facteur $\frac{1}{4}$ est inclus pour plus de commodité. Le gradient de cette fonction coût par rapport à $\boldsymbol{w}$ est

$$\nabla_{\boldsymbol{w}} J_k(\boldsymbol{w}) = z_{\boldsymbol{w},k}(z_{\boldsymbol{w},k}\boldsymbol{w} - \boldsymbol{y}_k), \tag{3.2}$$

ce qui conduit à la formulation de la descente de gradient

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \eta_t \left(\boldsymbol{y}_k\, z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k}\, \boldsymbol{w}_{t-1}\right), \tag{3.3}$$

où η_t est le taux d'apprentissage. Il s'avère que cette règle de mise à jour est essentiellement le critère de Oja (1982), qui est un cas particulier avec un unique neurone de la règle de Hebb (Dony et Haykin 1995).

Cette règle d'apprentissage converge vers un état d'équilibre, qui est le premier axe principal $\boldsymbol{w}_*$. En effet, quand $\boldsymbol{w}_t$ converge à un état d'équilibre $\boldsymbol{w}$, nous avons de (3.3) que

$$\boldsymbol{y}_k z_{\boldsymbol{w},k} = z^2_{\boldsymbol{w},k}\boldsymbol{w},$$

à savoir

$$\boldsymbol{y}_k \boldsymbol{y}_k^\top \boldsymbol{w} = z^2_{\boldsymbol{w},k}\boldsymbol{w}.$$

En faisant la moyenne de la dernière expression sur l'ensemble des données, nous obtenons le problème bien connu de la décomposition en éléments propres de la matrice de covariance

$$\mathfrak{C}\boldsymbol{w} = \mathbb{E}(z^2_{\boldsymbol{w}})\, \boldsymbol{w},$$

où la valeur propre associée au vecteur propre en question $\boldsymbol{w}$ n'est autre que l'espérance du carré de $z_{\boldsymbol{w},k}$, c'est-à-dire la variance projetée. En conséquence, l'équilibre de $\boldsymbol{w}$ est le long de l'un des vecteurs propres de la matrice de covariance. Il est démontré dans (Oja 1982) que seul l'axe principal (à un signe près) est stable, tandis que les autres équilibres sont des points selles. En effet, il est facile de voir que la variance est maximale lorsque la fonction coût (3.1) est minimale, puisque cette minimisation est équivalente à la maximisation de $z^2_{\boldsymbol{w},k}\left(2 - \|\boldsymbol{w}\|^2\right)$ où le terme entre parenthèses est positif puisque la norme de $\boldsymbol{w}$ tend à l'unité. Par conséquent, la règle de mise à jour (3.3) converge vers le vecteur propre associé à la plus grande valeur propre de la matrice de covariance, sans la nécessité de la calculer.

3.1.2 Stratégies coopératives

Nous considérons maintenant une stratégie coopérative où chaque nœud k a accès aux informations de son voisinage $\mathcal{V}_k$, qui est le sous-ensemble de nœuds directement connectés au nœud k, y compris lui-même, c'est-à-dire $k \in \mathcal{V}_k$. Le problème d'optimisation est présenté comme une minimisation de la fonction coût $\sum_{k=1}^{N} J_k(\boldsymbol{w})$, où $J_k(\boldsymbol{w})$ est la fonction coût au nœud k, par exemple (3.1). Comme chaque nœud k communique uniquement avec ses nœuds voisins, nous introduisons les coefficients non négatifs c_{kl} qui rapportent le nœud k à ses voisins, en remplissant les conditions suivantes :

$$c_{kl} \geq 0, \qquad \sum_{l \in \mathcal{V}_k} c_{kl} = 1, \quad \text{et} \quad c_{kl} = 0 \text{ si } l \notin \mathcal{V}_k. \tag{3.4}$$

Il est à noter que si $l \notin \mathcal{V}_k$, c'est-à-dire $c_{kl} = 0$, alors $k \notin \mathcal{V}_l$ et ainsi c_{lk} est également nul. Nous collectons les coefficients c_{kl} dans une matrice $\boldsymbol{C}$ de taille $(N \times N)$, telle que la $k^{\text{ème}}$ ligne de $\boldsymbol{C}$ soit formée de $\{c_{kl}, l = 1, 2, \cdots, N\}$. La sommation dans les conditions (3.4) s'exprime en forme matricielle comme suit :

$$\boldsymbol{C}\mathbb{1}_N = \mathbb{1}_N$$

où $\mathbb{1}_N$ est le vecteur de taille $(N \times 1)$ dont les éléments sont égaux à un. On dit que $\boldsymbol{C}$ est une matrice stochastique droite. Utilisant ces coefficients, chaque nœud l a une fonction coût locale qui consiste en une combinaison pondérée des coûts individuels des voisins du nœud l (y compris l lui-même) :

$$J_l^{\text{loc}}(\boldsymbol{w}) = \sum_{k \in \mathcal{V}_l} c_{kl}\, J_k(\boldsymbol{w}) = \sum_{k=1}^{N} c_{kl}\, J_k(\boldsymbol{w}). \tag{3.5}$$

Il s'avère que la somme cumulée de ces fonctions coût locales est égale à la fonction coût globale, puisque nous avons

$$\begin{aligned}
\sum_{k=1}^{N} J_k(\boldsymbol{w}) &= \sum_{k=1}^{N} \Big(\sum_{l=1}^{N} c_{kl} \Big) J_k(\boldsymbol{w}) \\
&= \sum_{l=1}^{N} \sum_{k=1}^{N} c_{kl} J_k(\boldsymbol{w}) \\
&= \sum_{l=1}^{N} J_l^{\text{loc}}(\boldsymbol{w}).
\end{aligned}$$

Pour le nœud k, la fonction coût globale peut être décomposée comme suit :

$$\sum_{l=1}^{N} J_l(\boldsymbol{w}) = J_k^{\text{loc}}(\boldsymbol{w}) + \sum_{\substack{l=1 \\ l \neq k}}^{N} J_l^{\text{loc}}(\boldsymbol{w}). \tag{3.6}$$

Alors que le premier terme du côté droit est connu au niveau du nœud k, le second devrait être estimé en utilisant les informations des voisins du nœud en question ; ainsi, la somme ci-dessus est-elle contrainte à son voisinage. En outre, nous relaxons cette restriction en imposant une contrainte sur la norme entre le vecteur estimé $\boldsymbol{w}$ et l'axe principal optimal (global) $\boldsymbol{w}_*$ dans le voisinage du nœud k, avec la régularisation suivante :

$$\sum_{\substack{l=1 \\ l \neq k}}^{N} J_l^{\text{loc}}(\boldsymbol{w}) \approx \sum_{l \in \mathcal{V}_k \setminus \{k\}} b_{lk} \, \|\boldsymbol{w} - \boldsymbol{w}_*\|^2,$$

où les paramètres b_{lk} contrôlent le compromis entre la précision et la finesse de la solution. Cette régularisation est aussi motivée par l'utilisation du développement de Taylor de $J_l^{\text{loc}}(\boldsymbol{w})$, tel que décrit par Chen et Sayed (2012). Rappelons que nous utilisons $\boldsymbol{w}_*$ dans cette formulation, sachant que nous n'en avons pas accès. Nous montrons dans ce qui suit comment surmonter ce problème.

En injectant l'approximation ci-dessus dans (3.6), la minimisation de la fonction coût globale de k est équivalente à minimiser

$$\begin{aligned} J_k^{\text{glob}}(\boldsymbol{w}) &= J_k^{\text{loc}}(\boldsymbol{w}) + \sum_{l \in \mathcal{V}_k \setminus \{k\}} b_{lk} \|\boldsymbol{w} - \boldsymbol{w}_*\|^2 \\ &= \sum_{l \in \mathcal{V}_k} c_{lk} J_l(\boldsymbol{w}) + \sum_{l \in \mathcal{V}_k \setminus \{k\}} b_{lk} \|\boldsymbol{w} - \boldsymbol{w}_*\|^2, \end{aligned} \tag{3.7}$$

où la dernière égalité découle de (3.5). Afin de minimiser cette fonction coût, le nœud k applique la descente de gradient sur $J_k^{\text{glob}}(\boldsymbol{w})$ avec :

$$\boldsymbol{w}_{k,t} = \boldsymbol{w}_{k,t-1} - \eta_{k,t} \nabla_{\boldsymbol{w}} J_k^{\text{glob}}(\boldsymbol{w}_{k,t-1}). \tag{3.8}$$

Ici, $\eta_{k,t}$ est le taux d'apprentissage pour le nœud k à l'itération t. En remplaçant $J_k^{\text{glob}}(\boldsymbol{w})$ par son expression dans (3.7), nous obtenons :

$$\boldsymbol{w}_{k,t} = \boldsymbol{w}_{k,t-1} - \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \nabla_{\boldsymbol{w}} J_l(\boldsymbol{w}_{k,t-1}) + \eta_{k,t} \sum_{l \in \mathcal{V}_k \setminus \{k\}} b_{lk} (\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}).$$

Dans cette expression, $\nabla_{\boldsymbol{w}} J_l(\boldsymbol{w}_{k,t-1})$ est le gradient de la fonction coût, présenté dans (3.2). Dans ce cas, nous obtenons

$$\begin{aligned} \boldsymbol{w}_{k,t} = \boldsymbol{w}_{k,t-1} &+ \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{w}_{k,t-1} \right) \\ &+ \eta_{k,t} \sum_{l \in \mathcal{V}_k \setminus \{k\}} b_{lk} (\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}). \end{aligned} \tag{3.9}$$

Sans perte de généralité, nous considérons la règle de mise à jour (3.9) dans ce qui suit. Cette règle de mise à jour pour $\boldsymbol{w}_{k,t}$ consiste à ajouter deux termes de correction à la précédente estimation $\boldsymbol{w}_{k,t-1}$: le premier terme opère dans le sens de la variance maximale et le second associe la régularisation du voisinage. En décomposant ce calcul en deux étapes successives, nous obtenons deux stratégies possibles, en fonction de l'ordre d'ajout des termes de correction, et qui diffèrent essentiellement dans l'approximation de l'inconnu $\boldsymbol{w}_*$ dans (3.9), comme montré dans ce qui suit.

Stratégie adapter-puis-combiner (APC)

Nous exprimons la règle de mise à jour (3.9) comme suit :

$$\begin{aligned} \boldsymbol{\phi}_{k,t} &= \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{w}_{k,t-1} \right), \\ \boldsymbol{w}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l \in \mathcal{V}_k \setminus \{k\}} b_{lk} (\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}). \end{aligned}$$

La première étape utilise les vecteurs gradients locaux du voisinage du nœud k pour mettre à jour $\boldsymbol{w}_{k,t-1}$ afin d'avoir une estimation intermédiaire $\boldsymbol{\phi}_{k,t}$. La seconde étape met à jour cette dernière $\boldsymbol{\phi}_{k,t}$ pour avoir $\boldsymbol{w}_{k,t}$. Cette seconde étape n'est pas réalisable car les nœuds n'ont pas accès à la solution optimale $\boldsymbol{w}_*$. Cependant, chaque nœud l a sa propre approximation de $\boldsymbol{w}_*$, qui est son estimation intermédiaire locale $\boldsymbol{\phi}_{l,t}$. Par conséquent, nous remplaçons $\boldsymbol{w}_*$ par $\boldsymbol{\phi}_{l,t}$. D'autre part, la valeur intermédiaire $\boldsymbol{\phi}_{k,t}$ au

nœud k est obtenue en incluant des informations des voisins (comme indiqué par la première étape), ce qui la rend une meilleure estimation de $\boldsymbol{w}_*$ que $\boldsymbol{w}_{k,t-1}$. Par conséquent, nous remplaçons $\boldsymbol{w}_{k,t-1}$ par $\boldsymbol{\phi}_{k,t}$ dans la seconde étape. Ainsi, la seconde étape est-elle réécrite sous la forme :

$$\begin{aligned}\boldsymbol{w}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l\in\mathcal{V}_k\setminus\{k\}} b_{lk}\,(\boldsymbol{\phi}_{l,t} - \boldsymbol{\phi}_{k,t}) \\ &= \Big(1 - \eta_{k,t} \sum_{l\in\mathcal{V}_k\setminus\{k\}} b_{lk}\Big)\,\boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l\in\mathcal{V}_k\setminus\{k\}} b_{lk}\,\phi_{l,t},\end{aligned}$$

ou, d'une façon équivalente, $\boldsymbol{w}_{k,t} = \sum_{l\in\mathcal{V}_k} a_{kl}\,\boldsymbol{\phi}_{l,t}$ où nous avons utilisé les coefficients de pondération non négatifs suivants :

$$a_{kl} = \begin{cases} 1 - \eta_{k,t}\sum_{i\in\mathcal{V}_k\setminus\{k\}} b_{ik}, & \text{si } l = k\,; \\ \eta_{k,t}\,b_{lk}, & \text{si } l \in \mathcal{V}_k\setminus\{k\}\,; \\ 0, & \text{autrement.}\end{cases}$$

Ces coefficients remplissent les conditions de convexité :

$$a_{kl} \geq 0, \qquad \sum_{l\in\mathcal{V}_k} a_{kl} = 1. \tag{3.10}$$

Soit $\boldsymbol{A}$ la matrice de taille $(N \times N)$ qui collecte les coefficients a_{kl} telle que la $k^{\text{ème}}$ colonne de $\boldsymbol{A}$ est formée des $a_{kl}, l = 1, 2, \cdots, N$. La sommation dans les conditions (3.10) s'exprime sous forme matricielle comme suit :

$$\boldsymbol{A}^\top \mathbb{1}_N = \mathbb{1}_N$$

On dit que $\boldsymbol{A}$ est une matrice stochastique gauche. Enfin, la règle de mise à jour pour la stratégie adapter-puis-combiner (APC) est :

$$\begin{aligned}\boldsymbol{\phi}_{k,t} &= \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk}\Big(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l}\boldsymbol{w}_{k,t-1}\Big), \\ \boldsymbol{w}_{k,t} &= \sum_{l\in\mathcal{V}_k} a_{kl}\,\boldsymbol{\phi}_{l,t}.\end{aligned}$$

Dans le cas particulier où aucun échange d'information ne s'effectue entre les nœuds, sauf pour l'étape d'agrégation, c'est-à-dire $c_{kk} = 1$ et $c_{kl} = 0$ pour tout $l \neq k$, nous obtenons :

$$\begin{aligned}\boldsymbol{\phi}_{k,t} &= \boldsymbol{w}_{k,t-1} + \eta_{k,t}\,\big(\boldsymbol{y}_k\, z_{\boldsymbol{w}_{k,t-1},k} - z^2_{\boldsymbol{w}_{k,t-1},k}\,\boldsymbol{w}_{k,t-1}\big), \\ \boldsymbol{w}_{k,t} &= \sum_{l\in\mathcal{V}_k} a_{kl}\,\boldsymbol{\phi}_{l,t}.\end{aligned}$$

Notons que cette configuration particulière permet de minimiser la quantité d'information échangée entre les nœuds, en excluant la transmission des estimations z et des mesures $\boldsymbol{y}$ dans le réseau, et en limitant les échanges à $\boldsymbol{w}$ (ou d'une manière équivalente $\boldsymbol{\phi}$).

Stratégie combiner-puis-adapter (CPA)

Dans cette stratégie, nous exprimons la règle de mise à jour (3.9) comme suit :

$$\begin{aligned}\boldsymbol{\phi}_{k,t} &= \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l\in\mathcal{V}_k\setminus\{k\}} b_{lk}(\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}), \\ \boldsymbol{w}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk}\Big(\boldsymbol{y}_l\, z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l}\,\boldsymbol{w}_{k,t-1}\Big).\end{aligned}$$

Pour les mêmes raisons indiquées dans la stratégie adapter-puis-combiner, nous remplaçons $\boldsymbol{w}_*$ par $\boldsymbol{w}_{l,t-1}$ dans la première étape, et $\boldsymbol{w}_{l,t-1}$ par $\boldsymbol{\phi}_{l,t}$ dans la seconde étape. En présentant les même coefficients a_{kl}, nous obtenons la règle de mise à jour de la stratégie combiner-puis-adapter (CPA) :

$$\begin{aligned}\boldsymbol{\phi}_{k,t} &= \sum_{l\in\mathcal{V}_k} a_{kl}\,\boldsymbol{w}_{l,t-1},\\ \boldsymbol{w}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t}\sum_{l\in\mathcal{V}_k} c_{lk}\Big(\boldsymbol{y}_l\, z_{\boldsymbol{\phi}_{k,t-1},l} - z^2_{\boldsymbol{\phi}_{k,t-1},l}\,\boldsymbol{\phi}_{k,t}\Big).\end{aligned}$$

Dans le cas d'aucun échange d'information à l'exception de l'étape d'agrégation, nous obtenons :

$$\begin{aligned}\boldsymbol{\phi}_{k,t} &= \sum_{l\in\mathcal{V}_k} a_{kl}\,\boldsymbol{w}_{l,t-1},\\ \boldsymbol{w}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t}\left(\boldsymbol{y}_k\, z_{\boldsymbol{\phi}_{k,t-1},k} - z^2_{\boldsymbol{\phi}_{k,t-1},k}\,\boldsymbol{\phi}_{k,t}\right).\end{aligned}$$

Stratégie non coopérative

Nous notons que les stratégies adaptatives APC et CPA se réduisent à la stratégie non coopérative quand le nœud k ne prend en considération que ses propres informations. Dans ce cas, les coefficients a_{kl} et c_{kl} sont choisis tels que :

$$a_{kl} = c_{kl} = \delta_{kl}$$

où δ_{kl} est le symbole de Kronecker défini selon :

$$\delta_{kl} = \begin{cases} 1, & \text{si } l = k\,;\\ 0, & \text{sinon.}\end{cases}$$

En forme matricielle, ce cas se traduit par :

$$\boldsymbol{A} = \boldsymbol{C} = \boldsymbol{I}_N$$

3.1.3 Stratégie de consensus

Le consensus (Degroot 1974) est une classe de stratégies distribuées qui révèle la coordination des réseaux de capteurs/agents autonomes. Afin d'estimer un paramètre commun $\boldsymbol{w}_*$, le consensus permet à chaque nœud k d'aligner son estimation $\boldsymbol{w}_{k,t}$ avec celles de ses voisins, comme suit

$$\boldsymbol{w}_{k,t} = \sum_{l\in\mathcal{V}_k} a_{kl}\,\boldsymbol{w}_{l,t-1},$$

où a_{kl} sont des poids non négatifs que le nœud k affecte aux estimations de ses voisins. Afin de répartir le calcul entre les nœuds, nous réexaminons les travaux de Nedic et Ozdaglar (2009) et de Yuan et al. (2013) afin de combiner la stratégie de consensus et l'approche de la descente de gradient susmentionnée. Dans un tel algorithme, chaque nœud k soustrait le gradient de la moyenne des estimations de ses voisins. Par conséquent, le nœud k aligne son estimation $\boldsymbol{w}_{k,t}$ avec celles de ses voisins pour minimiser sa propre fonction coût $J_k(\boldsymbol{w})$ en utilisant la règle de mise à jour suivante :

$$\boldsymbol{w}_{k,t} = \sum_{l\in\mathcal{V}_k} a_{kl}\boldsymbol{w}_{k,t-1} - \eta_{k,t}\,\nabla_{\boldsymbol{w}} J_k(\boldsymbol{w}_{k,t-1}).$$

Dans ce contexte, l'estimation de l'axe principal est donnée par la fonction coût proposée dans (3.1). En conséquence, l'ACP en réseau avec la stratégie de consensus prend la forme suivante :

$$\boldsymbol{w}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl} \boldsymbol{w}_{k,t-1} + \eta_{k,t} \Big(\boldsymbol{y}_k \, z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k} \boldsymbol{w}_{k,t-1} \Big).$$

La stratégie de consensus et les stratégies coopératives adapter-puis-combiner et combiner-puis-adapter ont essentiellement la même complexité de calcul. Cependant, les stratégies coopératives surpassent les stratégies de type consensus, comme montré en filtrage adaptatif dans (Tu et Sayed 2012). Cette propriété est toujours valable dans notre cas avec l'ACP. En effet, les stratégies coopératives adapter-puis-combiner et combiner-puis-adapter peuvent être exprimées respectivement par :

$$\boldsymbol{w}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl} \Big(\boldsymbol{w}_{k,t-1} + \eta_{k,t} \left(\boldsymbol{y}_k \, z_{\boldsymbol{w},k} - z^2_{\boldsymbol{w},k} \boldsymbol{w}_{k,t-1} \right) \Big),$$

et

$$\boldsymbol{w}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl} \boldsymbol{w}_{k,t-1} + \eta_{k,t} \Big(\boldsymbol{y}_k \, z_{\boldsymbol{w},k} - z^2_{\boldsymbol{w},k} \sum_{l \in \mathcal{V}_k} a_{kl} \boldsymbol{w}_{k,t-1} \Big).$$

En considérant par exemple la stratégie combiner-puis-adapter, nous notons que les coefficients de combinaison a_{kl} apparaissent dans le terme de correction, alors que le consensus n'utilise que $\boldsymbol{w}_{k,t-1}$ pour corriger l'erreur (c'est-à-dire sans impliquer le voisinage), ce qui est moins efficace. Ces résultats théoriques seront validés par les résultats expérimentaux dans la section 3.5.

3.2 Autres fonctions coût pour l'ACP en réseau

Dans cette partie, nous étendons l'étude de la section 3.1 pour inclure différentes fonctions coût. Ces fonctions coût émergent de divers principes, comme décrit ci-après.

3.2.1 La théorie de l'information (TI)

En s'inspirant du cadre de la théorique de l'information étudiée dans (Plumbley 1995, Miao et Hua 1998), nous considérons la maximisation de l'entropie, définie par $\ln(z^2_{\boldsymbol{w},k})$, sous une contrainte sur la norme de $\boldsymbol{w}$. A cette fin, nous minimisons la fonction coût suivante associée au nœud k :

$$J_k(\boldsymbol{w}) = \tfrac{1}{2} \boldsymbol{w}^\top \boldsymbol{w} - \tfrac{1}{2} \ln(z^2_{\boldsymbol{w},k}), \tag{3.11}$$

dont le gradient par rapport à $\boldsymbol{w}$ est

$$\nabla_{\boldsymbol{w}} J_k(\boldsymbol{w}) = \boldsymbol{w} - \frac{\boldsymbol{y}_k}{z_{\boldsymbol{w},k}}. \tag{3.12}$$

En appliquant la technique de descente de gradient à cette fonction coût, nous obtenons la règle de mise à jour non coopérative qui suit :

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \eta_t \Big(\frac{\boldsymbol{y}_k}{z_{\boldsymbol{w}_{t-1},k}} - \boldsymbol{w}_{t-1} \Big).$$

Les stratégies de coopération sont également étudiées par la diffusion de l'information. La règle de mise à jour adapter-puis-combiner devient dans ce cas :

$$\boldsymbol{\phi}_{k,t} = \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \Big(\frac{\boldsymbol{y}_l}{z_{\boldsymbol{w}_{k,t-1},l}} - \boldsymbol{w}_{k,t-1} \Big),$$
$$\boldsymbol{w}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\phi}_{l,t}.$$

La règle de mise à jour combiner-puis-adapter devient :

$$\boldsymbol{\phi}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{w}_{l,t-1},$$
$$\boldsymbol{w}_{k,t} = \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \Big(\frac{\boldsymbol{y}_l}{z_{\boldsymbol{\phi}_{k,t-1},l}} - \boldsymbol{\phi}_{k,t-1} \Big).$$

Bien que la fonction coût (3.1) étudiée dans la section 3.1 et la fonction coût (3.11) inspirée de la théorie de l'information proviennent de deux principes très différents, les règles de mise à jour correspondantes sont intimement liées. En effet, la seconde peut être obtenue de la première en utilisant le taux d'apprentissage normalisé $\eta_t / z^2_{\boldsymbol{w}_{t-1},k}$.

3.2.2 Quotient de Rayleigh (QR)

Le problème de décomposition en éléments propres est fortement lié au quotient de Rayleigh. Inspiré du cadre du quotient de Rayleigh étudié par Luo et al. (1997) et par Cirrincione et al. (2002), nous minimisons la fonction coût suivante associée au nœud k :

$$J_k(\boldsymbol{w}) = -\frac{z^2_{\boldsymbol{w},k}}{2\, \boldsymbol{w}^\top \boldsymbol{w}}, \tag{3.13}$$

dont le gradient par rapport à $\boldsymbol{w}$ est

$$\nabla_{\boldsymbol{w}} J_k(\boldsymbol{w}) = \frac{1}{\boldsymbol{w}^\top \boldsymbol{w}} \Big(\boldsymbol{w} \frac{z^2_{\boldsymbol{w},k}}{\boldsymbol{w}^\top \boldsymbol{w}} - \boldsymbol{y}_k z_{\boldsymbol{w},k} \Big). \tag{3.14}$$

La technique de descente de gradient, appliquée à cette fonction coût, conduit à la règle de mise à jour non coopérative qui suit :

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \eta_t \frac{1}{\boldsymbol{w}_{t-1}^\top \boldsymbol{w}_{t-1}} \Big(\boldsymbol{y}_k z_{\boldsymbol{w}_{t-1},k} - \boldsymbol{w}_{t-1} \frac{z^2_{\boldsymbol{w}_{t-1},k}}{\boldsymbol{w}_{t-1}^\top \boldsymbol{w}_{t-1}} \Big).$$

En incluant un processus de diffusion, nous proposons deux stratégies de coopération. Dans le cas de la règle de mise à jour adapter-puis-combiner, nous obtenons :

$$\boldsymbol{\phi}_{k,t} = \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \frac{1}{\boldsymbol{w}_{k,t-1}^\top \boldsymbol{w}_{k,t-1}} \Big(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - \boldsymbol{w}_{k,t-1} \frac{z^2_{\boldsymbol{w}_{k,t-1},l}}{\boldsymbol{w}_{k,t-1}^\top \boldsymbol{w}_{k,t-1}} \Big),$$
$$\boldsymbol{w}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\phi}_{l,t}.$$

Pour la règle de mise à jour combiner-puis-adapter, nous avons :

$$\boldsymbol{\phi}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{w}_{l,t-1},$$

$$\boldsymbol{w}_{k,t} = \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \frac{1}{\boldsymbol{\phi}_{k,t-1}^{\top} \boldsymbol{\phi}_{k,t-1}} \Big(y_l z_{\boldsymbol{\phi}_{k,t-1},l} - \boldsymbol{\phi}_{k,t-1} \frac{z^2_{\boldsymbol{\phi}_{k,t-1},l}}{\boldsymbol{\phi}_{k,t-1}^{\top} \boldsymbol{\phi}_{k,t-1}} \Big).$$

3.2.3 Algorithmes OJAn et LUO

Les algorithmes OJAn et LUO, présentés par Luo et al. (1997), sont liés au quotient de Rayleigh (Taleb et Cirrincione 1999, Cirrincione et al. 2002). L'algorithme OJAn est une version normalisée de l'algorithme de Oja. Il peut être obtenu à partir du quotient de Rayleigh en utilisant le taux d'apprentissage $\eta_t(\boldsymbol{w}_{t-1}^{\top} \boldsymbol{w}_{t-1})$. En revenant à la formulation de descente de gradient, nous obtenons la règle de mise à jour non coopérative suivante :

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \eta_t \Big(y_k z_{\boldsymbol{w}_{t-1},k} - \boldsymbol{w}_{t-1} \frac{z^2_{\boldsymbol{w}_{t-1},k}}{\boldsymbol{w}_{t-1}^{\top} \boldsymbol{w}_{t-1}} \Big).$$

Dans ce cas, la règle de mise à jour adapter-puis-combiner devient :

$$\boldsymbol{\phi}_{k,t} = \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \Big(y_l z_{\boldsymbol{w}_{k,t-1},l} - \boldsymbol{w}_{k,t-1} \frac{z^2_{\boldsymbol{w}_{k,t-1},l}}{\boldsymbol{w}_{k,t-1}^{\top} \boldsymbol{w}_{k,t-1}} \Big),$$

$$\boldsymbol{w}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\phi}_{l,t},$$

et la mise à jour combiner-puis-adapter devient :

$$\boldsymbol{\phi}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{w}_{l,t-1},$$

$$\boldsymbol{w}_{k,t} = \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \Big(y_l z_{\boldsymbol{\phi}_{k,t-1},l} - \boldsymbol{\phi}_{k,t-1} \frac{z^2_{\boldsymbol{\phi}_{k,t-1},l}}{\boldsymbol{\phi}_{k,t-1}^{\top} \boldsymbol{\phi}_{k,t-1}} \Big).$$

L'algorithme LUO est obtenu à partir du quotient de Rayleigh en utilisant le taux d'apprentissage $\eta_t(\boldsymbol{w}_{t-1}^{\top} \boldsymbol{w}_{t-1})^2$. Cela conduit à la règle de mise à jour non coopérative suivante :

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \eta_t\, (\boldsymbol{w}_{t-1}^{\top} \boldsymbol{w}_{t-1}) \Big(y_k z_{\boldsymbol{w}_{t-1},k} - \boldsymbol{w}_{t-1} \frac{z^2_{\boldsymbol{w}_{t-1},k}}{\boldsymbol{w}_{t-1}^{\top} \boldsymbol{w}_{t-1}} \Big),$$

ainsi que les deux stratégies de coopération, adapter-puis-combiner :

$$\boldsymbol{\phi}_{k,t} = \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} (\boldsymbol{w}_{k,t-1}^{\top} \boldsymbol{w}_{k,t-1}) \Big(y_l z_{\boldsymbol{w}_{k,t-1},l} - \boldsymbol{w}_{k,t-1} \frac{z^2_{\boldsymbol{w}_{k,t-1},l}}{\boldsymbol{w}_{k,t-1}^{\top} \boldsymbol{w}_{k,t-1}} \Big),$$

$$\boldsymbol{w}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\phi}_{l,t}.$$

et combiner-puis-adapter :

$$\boldsymbol{\phi}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{w}_{l,t-1},$$

$$\boldsymbol{w}_{k,t} = \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} (\boldsymbol{\phi}_{k,t-1}^{\top} \boldsymbol{\phi}_{k,t-1}) \Big(y_l z_{\boldsymbol{\phi}_{k,t-1},l} - \boldsymbol{\phi}_{k,t-1} \frac{z^2_{\boldsymbol{\phi}_{k,t-1},l}}{\boldsymbol{\phi}_{k,t-1}^{\top} \boldsymbol{\phi}_{k,t-1}} \Big).$$

3.3 Performances des stratégies coopératives

Dans ce paragraphe, nous étudions les performances des stratégies coopératives en termes de convergence et d'erreur récursive.

3.3.1 Modèle général coopératif

Nous introduisons un modèle général coopératif afin que les stratégies APC et CPA seront des cas spéciaux de ce modèle. Nous considérons pour cela le modèle suivant :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{1,kl}\, \boldsymbol{w}_{l,t-1}, \\
\boldsymbol{\psi}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} \nabla_{\boldsymbol{w}} J_l(\boldsymbol{\phi}_{k,t}), \\
\boldsymbol{w}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{2,kl}\, \boldsymbol{\psi}_{l,t},
\end{aligned}$$

où les scalaires $a_{1,kl}$, c_{lk} et $a_{2,kl}$ désignent des coefficients non négatifs. Soient $\boldsymbol{A}_1$, $\boldsymbol{C}$ et $\boldsymbol{A}_2$ les matrices qui regroupent ces coefficients respectivement. Alors ces matrices satisfont les conditions suivantes :

$$\boldsymbol{A}_1^\top \mathbb{1}_N = \mathbb{1}_N, \qquad \boldsymbol{C}\mathbb{1}_N = \mathbb{1}_N, \qquad \boldsymbol{A}_2^\top \mathbb{1}_N = \mathbb{1}_N.$$

En d'autres termes, $\boldsymbol{A}_1$ et $\boldsymbol{A}_2$ sont des matrices stochastiques gauches et $\boldsymbol{C}$ est une matrice stochastique droite.

Les stratégies de coopération susmentionnées se déduisent selon le choix des matrices $\boldsymbol{A}_1$, $\boldsymbol{C}$ et $\boldsymbol{A}_2$. Par exemple, si $\boldsymbol{A}_1 = \boldsymbol{I}_N$ et $\boldsymbol{A}_2 = \boldsymbol{A}$, nous obtenons la stratégie APC. Si $\boldsymbol{A}_1 = \boldsymbol{A}$ et $\boldsymbol{A}_2 = \boldsymbol{I}_N$, nous aboutissons à la stratégie CPA. Le choix $\boldsymbol{C} = \boldsymbol{I}_N$ correspond au cas d'aucun échange d'information à l'exception de l'étape d'agrégation. De plus, si $\boldsymbol{A}_1 = \boldsymbol{A}_2 = \boldsymbol{C} = \boldsymbol{I}_N$, nous avons la stratégie non coopérative. La Table 3.1 montre ces différents cas.

3.3.2 Erreur récursive

Notre objectif est d'examiner la convergence des estimations $\boldsymbol{w}_{k,t}$ obtenues à partir des stratégies coopératives, où la convergence est étudiée par

Table 3.1 – *Différents choix de $\boldsymbol{A}_1$, $\boldsymbol{C}$ et $\boldsymbol{A}_2$ correspondant aux différentes stratégies coopératives.*

$\boldsymbol{A}_1$	$\boldsymbol{A}_2$	$\boldsymbol{C}$	Stratégie
$\boldsymbol{I}_N$	$\boldsymbol{A}$	$\boldsymbol{C}$	APC
$\boldsymbol{I}_N$	$\boldsymbol{A}$	$\boldsymbol{I}_N$	APC sans échange d'informations
$\boldsymbol{A}$	$\boldsymbol{I}_N$	$\boldsymbol{C}$	CPA
$\boldsymbol{A}$	$\boldsymbol{I}_N$	$\boldsymbol{I}_N$	CPA sans échange d'informations
$\boldsymbol{I}_N$	$\boldsymbol{I}_N$	$\boldsymbol{I}_N$	non coopérative

rapport à la solution optimale, c'est à dire l'axe principal $\boldsymbol{w}_*$. Pour cela, nous introduisons les vecteurs d'erreur suivants :

$$\widetilde{\boldsymbol{\phi}}_{k,t} = \boldsymbol{w}_* - \boldsymbol{\phi}_{k,t}$$
$$\widetilde{\boldsymbol{\psi}}_{k,t} = \boldsymbol{w}_* - \boldsymbol{\psi}_{k,t}$$
$$\widetilde{\boldsymbol{w}}_{k,t} = \boldsymbol{w}_* - \boldsymbol{w}_{k,t}$$

Chacun de ces vecteurs d'erreur mesure le résidu relatif à l'axe principal optimal $\boldsymbol{w}_*$. Avant d'étudier la convergence, nous rappelons que :

$$z_{\boldsymbol{w}_*,k} = \boldsymbol{y}_k^\top \boldsymbol{w}_*. \tag{3.15}$$

En multipliant les deux côtés de l'égalité par $\boldsymbol{y}_k$ et en prenant l'espérance, nous obtenons :

$$\mathbb{E}(\boldsymbol{y}_k z_{\boldsymbol{w}_*,k}) = \mathbb{E}(\boldsymbol{y}_k \boldsymbol{y}_k^\top)\boldsymbol{w}_*, \tag{3.16}$$

c'est-à-dire

$$\boldsymbol{m}_{y\boldsymbol{w}_*,k} = \mathfrak{C}\boldsymbol{w}_*, \tag{3.17}$$

avec $\boldsymbol{m}_{y\boldsymbol{w}_*,k} = \mathbb{E}(\boldsymbol{y}_k z_{\boldsymbol{w}_*,k})$. Nous en déduisons que $\boldsymbol{w}_*$ est la solution d'un système d'équations linéaires, selon :

$$\boldsymbol{w}_* = \mathfrak{C}^{-1}\boldsymbol{m}_{y\boldsymbol{w}_*,k}. \tag{3.18}$$

Il est utile de réinterpréter cette observation comme étant la solution à un problème de minimisation d'erreur quadratique moyenne, selon :

$$\min_{\boldsymbol{w}} \mathbb{E}\|\boldsymbol{y}_k - z_{\boldsymbol{w},k}\boldsymbol{w}\|^2.$$

Soit donc la fonction coût :

$$J_k^{\text{esp}}(\boldsymbol{w}) = \tfrac{1}{4}\mathbb{E}\|\boldsymbol{y}_k - z_{\boldsymbol{w},k}\boldsymbol{w}\|^2. \tag{3.19}$$

En développant cette expression, nous obtenons :

$$J_k^{\text{esp}}(\boldsymbol{w}) = \tfrac{1}{4}\Big[\mathbb{E}\|\boldsymbol{y}_k\|^2 - \boldsymbol{m}_{y\boldsymbol{w},k}^\top \boldsymbol{w} - \boldsymbol{w}^\top \boldsymbol{m}_{y\boldsymbol{w},k} + \boldsymbol{w}^\top \mathbb{E}(z_{\boldsymbol{w},k}^2)\boldsymbol{w}\Big].$$

En dérivant $J_k^{\text{esp}}(\boldsymbol{w})$ par rapport à $\boldsymbol{w}$, nous obtenons son vecteur gradient :

$$\nabla_{\boldsymbol{w}} J_k^{\text{esp}}(\boldsymbol{w}) = \mathbb{E}(z_{\boldsymbol{w},k}^2)\boldsymbol{w} - \boldsymbol{m}_{y\boldsymbol{w},k}.$$

En remplaçant cette fonction coût dans le modèle général coopératif, nous obtenons :

$$\boldsymbol{\phi}_{k,t} = \sum_{l\in\mathcal{V}_k} a_{1,kl}\, \boldsymbol{w}_{l,t-1}, \tag{3.20}$$

$$\boldsymbol{\psi}_{k,t} = \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk}(\boldsymbol{m}_{y\boldsymbol{w},k} - \mathbb{E}(z_{\boldsymbol{w},k}^2)\boldsymbol{\phi}_{k,t}), \tag{3.21}$$

$$\boldsymbol{w}_{k,t} = \sum_{l\in\mathcal{V}_k} a_{2,kl}\, \boldsymbol{\psi}_{l,t}, \tag{3.22}$$

En utilisant les équations (3.15) et (3.17), et le fait que $\boldsymbol{w}_*$ est un vecteur propre de la matrice de covariance $\mathfrak{C}$, nous avons

$$\boldsymbol{m}_{y\boldsymbol{w}_*,k} = \mathfrak{C}\boldsymbol{w}_* = \lambda\boldsymbol{w}_*,$$

et

$$\begin{aligned}\mathbb{E}(z^2_{\boldsymbol{w}_*,k}) &= \mathbb{E}(\boldsymbol{w}_*^\top \boldsymbol{y}_k \boldsymbol{y}_k^\top \boldsymbol{w}_*) \\ &= \boldsymbol{w}_*^\top \boldsymbol{\mathfrak{C}} \boldsymbol{w}_* \\ &= \lambda \|\boldsymbol{w}_*\|^2 \\ &= \lambda,\end{aligned}$$

puisque $\boldsymbol{w}_*$ est un vecteur de norme unité. En retranchant $\boldsymbol{w}_*$ des deux côtés des équations (3.20)-(3.22), nous obtenons :

$$\widetilde{\boldsymbol{\phi}}_{k,t} = \sum_{l\in\mathcal{V}_k} a_{1,kl}\, \widetilde{\boldsymbol{w}}_{l,t-1}, \tag{3.23}$$

$$\widetilde{\boldsymbol{\psi}}_{k,t} = \Big(\boldsymbol{I}_p - \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk}\lambda\Big)\widetilde{\boldsymbol{\phi}}_{k,t}, \tag{3.24}$$

$$\widetilde{\boldsymbol{w}}_{k,t} = \sum_{l\in\mathcal{V}_k} a_{2,kl}\, \widetilde{\boldsymbol{\psi}}_{l,t}, \tag{3.25}$$

Nous pouvons décrire ces relations en recueillant les informations à travers le réseau dans des blocs de vecteurs et de matrices. Nous recueillons les vecteurs d'erreur de tous les nœuds dans les vecteurs de N blocs, dont les éléments individuels sont de taille $(p \times 1)$ chacun :

$$\widetilde{\boldsymbol{\psi}_t} = \begin{bmatrix} \widetilde{\boldsymbol{\psi}}_{1,t} \\ \widetilde{\boldsymbol{\psi}}_{2,t} \\ \vdots \\ \widetilde{\boldsymbol{\psi}}_{N,t} \end{bmatrix}, \qquad \widetilde{\boldsymbol{\phi}_t} = \begin{bmatrix} \widetilde{\boldsymbol{\phi}}_{1,t} \\ \widetilde{\boldsymbol{\phi}}_{2,t} \\ \vdots \\ \widetilde{\boldsymbol{\phi}}_{N,t} \end{bmatrix}, \qquad \widetilde{\boldsymbol{w}_t} = \begin{bmatrix} \widetilde{\boldsymbol{w}}_{1,t} \\ \widetilde{\boldsymbol{w}}_{2,t} \\ \vdots \\ \widetilde{\boldsymbol{w}}_{N,t} \end{bmatrix}.$$

Ces vecteurs blocs représentent l'erreur du réseau à l'itération t. Nous introduisons les matrices diagonales de taille $(N \times N)$ chacune :

$$\begin{aligned}\boldsymbol{\mathcal{M}} &= \mathrm{diag}\{\eta_{1,t}, \eta_{2,t}, \cdots, \eta_{N,t}\} \\ \boldsymbol{\mathcal{L}} &= \mathrm{diag}\{\sum_{l\in\mathcal{V}_1} c_{l1}\lambda, \sum_{l\in\mathcal{V}_2} c_{l2}\lambda, \cdots, \sum_{l\in\mathcal{V}_N} c_{lN}\lambda\}\end{aligned}$$

Dans le cas où $\boldsymbol{C} = \boldsymbol{I}_N$, nous avons $\boldsymbol{\mathcal{L}} = \lambda \boldsymbol{I}_N$. Nous introduisons de plus le produit de Kronecker :

$$\boldsymbol{\mathcal{A}}_1 = \boldsymbol{A}_1 \otimes \boldsymbol{I}_p, \qquad \boldsymbol{\mathcal{A}}_2 = \boldsymbol{A}_2 \otimes \boldsymbol{I}_p$$

La matrice $\boldsymbol{\mathcal{A}}_1$ est une matrice bloc de taille $(N \times N)$, dont l'élément bloc (l,k) est égale à $a_{1,lk}\boldsymbol{I}_p$. De même pour $\boldsymbol{\mathcal{A}}_2$ avec $a_{2,lk}\boldsymbol{I}_p$. En utilisant ces définitions, nous écrivons les expressions (3.23)-(3.25) comme suit :

$$\widetilde{\boldsymbol{\phi}}_t = \boldsymbol{\mathcal{A}}_1^\top\, \widetilde{\boldsymbol{w}}_{t-1}, \tag{3.26}$$

$$\widetilde{\boldsymbol{\psi}}_t = \Big(\boldsymbol{I}_N - \boldsymbol{\mathcal{M}}\boldsymbol{\mathcal{L}}\Big)\widetilde{\boldsymbol{\phi}}_t, \tag{3.27}$$

$$\widetilde{\boldsymbol{w}}_t = \boldsymbol{\mathcal{A}}_2^\top\, \widetilde{\boldsymbol{\psi}}_t. \tag{3.28}$$

Nous pouvons dire alors que l'erreur $\widetilde{\boldsymbol{w}}_t$ du réseau évolue selon la dynamique suivante :

$$\widetilde{\boldsymbol{w}}_t = \boldsymbol{\mathcal{A}}_2^\top (\boldsymbol{I}_N - \boldsymbol{\mathcal{M}}\boldsymbol{\mathcal{L}})\boldsymbol{\mathcal{A}}_1^\top \widetilde{\boldsymbol{w}}_{t-1}. \tag{3.29}$$

Si chaque nœud minimise sa propre fonction coût $J_k^{\text{esp}}(\boldsymbol{w})$ en utilisant la stratégie non coopérative, alors le vecteur erreur sur les N nœuds évolue selon la dynamique suivante :

$$\widetilde{\boldsymbol{w}}_t = (\boldsymbol{I}_N - \lambda \boldsymbol{\mathcal{M}})\widetilde{\boldsymbol{w}}_{t-1},$$

où les matrices $\boldsymbol{\mathcal{A}}_1$ et $\boldsymbol{\mathcal{A}}_2$ n'apparaissent plus et $\boldsymbol{\mathcal{L}}$ est remplacée par $\lambda \boldsymbol{I}_N$. Cette récurrence est un cas spécial de (3.29) avec $\boldsymbol{A}_1 = \boldsymbol{A}_2 = \boldsymbol{C} = \boldsymbol{I}_N$

3.3.3 Convergence

Le vecteur de l'erreur récursive (3.29) comprend des vecteurs et des matrices blocs. Afin d'examiner les propriétés de stabilité et de convergence de cette erreur, il est judicieux de s'appuyer sur une norme des vecteurs bloc. Dans l'annexe A.1 , nous décrivons la norme de bloc maximale et nous établissons certaines de ses propriétés utiles. Le résultat du Lemme A.5 nous permet d'établir la condition pour la convergence des stratégies coopératives. Il fournit une borne maximale pour le taux d'apprentissage $\eta_{k,t}$ de chaque nœud k.

Théorème 3.1 (Convergence à la solution optimale) *Considérons le problème de minimisation de fonction coût global* (3.7) *avec les fonctions coût individuelles données par* (3.19). *La matrice stochastique droite* $\boldsymbol{C}$ *et les matrices stochastiques gauche* $\boldsymbol{A}_1$ *et* $\boldsymbol{A}_2$ *définissent la topologie du réseau ainsi que l'échange d'information au voisinage. Chaque nœud exécute l'algorithme coopératif* (3.20)-(3.22). *Alors, toutes les estimations* $\boldsymbol{w}_{k,t}$ *convergent vers la solution optimale* $\boldsymbol{w}_*$ *si les taux d'apprentissage* $\eta_{k,t}$ *satisfont :*

$$\eta_{k,t} < \frac{2}{\sum_{l \in \mathcal{V}_k} c_{lk} \lambda} \tag{3.30}$$

Démonstration. Le vecteur erreur $\widetilde{\boldsymbol{w}}_t$ de l'équation (3.29) converge à zéro si, et seulement si, la matrice $\boldsymbol{\mathcal{A}}_2^\top (\boldsymbol{I}_N - \boldsymbol{\mathcal{M}}\boldsymbol{\mathcal{L}})\boldsymbol{\mathcal{A}}_1^\top$ est une matrice stable, c'est-à-dire toutes ses valeurs propres se trouvent strictement à l'intérieur du disque unité (Lancaster et Tismenetsky 1985). D'après le Corollaire A.1 de l'annexe A.1, $\boldsymbol{\mathcal{A}}_2^\top (\boldsymbol{I}_N - \boldsymbol{\mathcal{M}}\boldsymbol{\mathcal{L}})\boldsymbol{\mathcal{A}}_1^\top$ est stable si $(\boldsymbol{I}_N - \boldsymbol{\mathcal{M}}\boldsymbol{\mathcal{L}})$ est stable, puisque $\boldsymbol{\mathcal{A}}_1$ et $\boldsymbol{\mathcal{A}}_2$ sont des matrices stochastiques gauches. Cette stabilité est vérifiée par la condition (3.30) qui implique que toutes les valeurs propres de $(\boldsymbol{I}_N - \boldsymbol{\mathcal{M}}\boldsymbol{\mathcal{L}})$ sont inférieures à 1. □

Nous remarquons que cette stabilité ne dépend pas des matrices de combinaison $\boldsymbol{A}_1$ et $\boldsymbol{A}_2$, par contre elle dépend de la matrice $\boldsymbol{C}$. Dans le cas où il n'y a pas d'échange d'information entre les nœuds à l'exception de l'étape d'agrégation, c'est-à-dire $\boldsymbol{C} = \boldsymbol{I}_N$, cette condition devient :

$$\eta_{k,t} < \frac{2}{\lambda} \tag{3.31}$$

De même dans le cas de la stratégie non coopérative, c'est-à-dire $\boldsymbol{A}_1 = \boldsymbol{A}_2 = \boldsymbol{C} = \boldsymbol{I}_N$, nous obtenons la même condition.

3.4 Analyse de la convergence et étude de paramètres

Dans ce qui suit, nous fournissons une analyse de la convergence et nous étudions le choix des paramètres pour les stratégies proposées dans le présent document pour l'ACP en réseau.

3.4.1 Analyse de la convergence

Les stratégies coopératives adapter-puis-combiner et combiner-puis-adapter ont fondamentalement la même structure, et se comportent essentiellement de la même manière. Elles ne diffusent pas seulement les estimations locales, mais peuvent aussi diffuser les vecteurs gradients locaux. La différence entre les stratégies réside dans la variable choisie comme l'estimateur à mettre à jour. Dans le cas de la stratégie adapter-puis-combiner, l'estimateur est le résultat de l'étape de combinaison, alors que dans le cas de combiner-puis-adapter, l'estimateur est le résultat de l'adaptation.

Dans ce qui suit, nous fournissons une analyse de la convergence des algorithmes associés aux fonctions coût proposées. Les résultats suivants sont tirés ici pour l'ACP en réseau, à la lumière des travaux de Chatterjee (2005) sur les algorithmes adaptatifs. Nous renvoyons le lecteur à Chatterjee (2005) et aux références citées pour plus de détails.

La convergence de l'algorithme TI est indépendante de la plus grande valeur propre λ_1. La convergence de tous les autres algorithmes augmente avec λ_1 et λ_1/λ_2, lors de l'extraction du premier et du deuxième axes principaux. En outre, la convergence de l'algorithme avec la règle fondée sur Oja et celle de l'algorithme OJAn ne peuvent pas être améliorées en augmentant $\|\boldsymbol{w}_{k,0}\|$, tandis que la convergence de l'algorithme LUO diminue pour les petites valeurs de $\|\boldsymbol{w}_{k,0}\|$ et la convergence de l'algorithme QR diminue pour les grandes valeurs de $\|\boldsymbol{w}_{k,0}\|$. La Table 3.2 montre le taux de convergence en termes des expressions des constantes de temps pour le premier et le $i^{\text{ème}}$ axes principaux, pour tous les algorithmes étudiés dans le présent document. Les algorithmes d'extraction de plusieurs axes principaux sont étudiés en détail dans la section 4.1 du chapitre suivant.

Précisons que les algorithmes associés aux critères de Oja et de la TI convergent vers des vecteurs de norme unité, c'est-à-dire $\|\boldsymbol{w}_{k,t}\| \to 1$ lorsque $t \to \infty$. D'autre part, les algorithmes QR, OJAn, et LUO exigent une normalisation selon $\|\boldsymbol{w}_{k,0}\| = 1$ afin de converger vers une solution de norme unité.

3.4.2 Taux d'apprentissage ou pas de l'adaptation

Dans la stratégie non coopérative, le paramètre du taux d'apprentissage (pas de l'adaptation) η_t est choisi pour être suffisamment petit afin d'assurer la convergence. Il doit être inférieur à l'inverse de la plus grande valeur propre de $\mathfrak{C}$ (Chen et Chang 1995). Malheureusement, ces valeurs propres sont généralement inconnues. Pour assurer une convergence appropriée comme démontré par le théorème bien connu de Robbins-Monro en approximation stochastique (Robbins et Siegmund 1985), le pas η_t devrait décroître à chaque itération. Un exemple est $\eta_t = \eta_0/t$, où η_0

est un paramètre constant positif. Une approche alternative est la stratégie "rechercher-puis-converger" étudiée dans (Darken et Moody 1990, Honeine 2012), avec

$$\eta_t = \frac{\eta_0}{1 + t/\tau}.$$

Dans ce cas, le paramètre de retard τ détermine la durée de la phase de recherche initiale, avec $\eta_t \simeq \eta_0$ lorsque $t \ll \tau$, avant une phase de convergence où η_t diminue en se comportant comme η_0/t lorsque $t \gg \tau$.

Pour les stratégies coopératives (APC et CPA), le paramètre du pas $\eta_{k,t}$ est choisi petit pour assurer la convergence. Il peut également diminuer avec les itérations, comme par exemple avec la stratégie "rechercher-puis-converger" selon $\eta_{k,t} = \frac{\eta_{k,0}}{1+t/\tau}$. Nous retrouvons ainsi les mêmes propriétés susmentionnées, dont les deux phases, avec une recherche initiale suivie d'une convergence. Nous remarquons que ce pas converge vers zéro lorsque $t \longrightarrow \infty$. Par conséquent, il éteint l'adaptation. Ce choix n'est donc pas propice à un apprentissage continu, à l'opposé d'un pas constant pour assurer une adaptation continue dans le réseau.

3.4.3 Coefficients pour la combinaison dans les stratégies coopératives

Les stratégies coopératives dépendent du choix des coefficients a_{kl} et c_{lk}. Nous rappelons que les coefficients b_{kl} sont implicitement déterminés une fois les coefficients a_{kl} ont été fixés.

Les coefficients de combinaison a_{kl} utilisés dans les stratégies coopératives déterminent le poids que chaque nœud k attribue aux estimations reçues de ses voisins $l \in \mathcal{V}_k$. Les coefficients de combinaison c_{lk} utilisés dans les stratégies coopératives déterminent le poids que chaque nœud k attribue au gradient envoyé à ses voisins $l \in \mathcal{V}_k$. Ces coefficients doivent satisfaire respectivement les conditions de convexité (3.10) et (3.4). Cela signifie que la somme des coefficients a_{kl}, $l \in \mathcal{V}_k$, mise à l'échelle par le nœud k pour le flux d'information reçu par ses voisins est égale à un, et la somme des coefficients c_{kl}, $l \in \mathcal{V}_k$, mise à l'échelle par le nœud k à la circulation de l'information envoyée à ses voisins est égale à un.

En réexaminant les suggestions données dans la littérature du filtrage adaptatif (Sayed et al. 2013), nous considérons les règles suivantes pour définir les coefficients a_{kl} et c_{lk} :

TABLE 3.2 – *Expressions des constantes de temps pour le premier et le $i^{ème}$ axes principaux, pour chacun des algorithmes étudiés dans le présent document*

Algorithme	Constante de temps pour $\boldsymbol{w}_1$	Constante de temps pour $\boldsymbol{w}_i$
Oja	$1/\lambda_1$	$1/(\lambda_1 - \lambda_i)$
TI	1	1
QR	$\|\boldsymbol{w}_{k,0}\|^2/\lambda_1$	$\|\boldsymbol{w}_{k,0}\|^2/(\lambda_1 - \lambda_i)$
OJAn	$1/\lambda_1$	$1/(\lambda_1 - \lambda_i)$
LUO	$\|\boldsymbol{w}_{k,0}\|^{-2}/\lambda_1$	$\|\boldsymbol{w}_{k,0}\|^{-2}/(\lambda_1 - \lambda_i)$

1. Loi moyenne :
$$\begin{cases} 1/n_k, & \text{si } l \in \mathcal{V}_k\,; \\ 0, & \text{autrement.} \end{cases}$$

2. Loi laplacienne :
$$\begin{cases} 1/n_{\max}, & \text{si } \ l \in \mathcal{V}_k \backslash \{k\}\,; \\ 1-(n_k-1)/n_{\max}, & \text{si } \ k = l\,; \\ 0, & \text{autrement.} \end{cases}$$

3. Loi Métropolis :
$$\begin{cases} 1/\max\{n_k, n_l\}, & \text{si } \ l \in \mathcal{V}_k \backslash \{k\}\,; \\ 1-\sum_{j\in\mathcal{V}_k\backslash\{k\}} a_{kj}, & \text{si } \ k = l\,; \\ 0, & \text{autrement.} \end{cases}$$

Dans ces expressions, n_k désigne le degré du nœud k, c'est-à-dire la taille de son voisinage, et $n_{\max}$ désigne le degré maximum sur le réseau, c'est-à-dire $n_{\max} = \max\limits_{1\leq k\leq N} n_k$. Notons que lorsque la taille du voisinage est la même pour tous les nœuds, c'est-à-dire n_k est constante pour tout $k = 1, 2, \cdots, N$, ces trois règles sont équivalentes.

3.5 Résultats expérimentaux

L'ACP a été largement étudiée en recherche et en ingénierie, dans une large gamme d'applications. Toutes ces applications peuvent être explorées avec le cadre proposé dans le présent document, et peuvent donc bénéficier des avantages de l'ACP en réseau avec des stratégies non coopératives et coopératives. En outre, notre travail offre un socle pour le traitement de données massives (ou *Big Data*) et des systèmes à grande échelle, où le traitement distribué de l'information se généralise. Dans ce document, nous étudions la pertinence de l'approche proposée dans deux applications différentes, illustrant la diversité des applications qui peuvent tirer profit du travail présenté. La première concerne la réduction de la dimensionnalité et la compression de données acquises par les capteurs dans un réseau sans fil ; la suite de ce chapitre est consacrée à cette application. La seconde application, traitée au chapitre suivant, concerne le traitement des données massives et en particulier le traitement des images.

Les performances sont mesurées en termes d'angle entre l'axe principal optimal $\boldsymbol{w}_*$, obtenu à partir de la stratégie centralisée avec une décomposition en éléments propres de la matrice de covariance, et l'estimation $\boldsymbol{w}_{*,k}$ au nœud k, à savoir

$$\Theta = \arccos\left(\frac{\boldsymbol{w}_{*,k}^\top \boldsymbol{w}_*}{\|\boldsymbol{w}_{*,k}\|\|\boldsymbol{w}_*\|}\right). \tag{3.32}$$

Il est à noter qu'afin de fournir une étude comparative équitable, nous utilisons la même estimation initiale aléatoire, c'est-à-dire la même condition initiale, pour tous les algorithmes.

Séries temporelles de mesures dans un RCSF

La première application est la réduction de la dimensionnalité et la compression des données dans les réseaux de capteurs sans fil (RCSF). Les capteurs sont déployés en grand nombre pour couvrir une région donnée, et un grand nombre de mesures est acquis par chaque capteur. Les mesures des capteurs à différents endroits sont analysées afin d'identifier certains événements d'intérêt tels que le repérage et le suivi de la diffusion de phénomènes physiques. La capacité limitée en énergie et la faible porté en communication des capteurs empêchent la transmission d'un grand nombre de mesures recueillies. En outre, la capacité limitée en traitement et en mémoire de chaque capteur nécessite la mise en œuvre d'algorithmes à faible complexité de calcul. Il est donc crucial de réduire la dimensionnalité des mesures à chaque nœud, sans le fardeau de calcul d'une technique conventionnelle d'ACP.

Dans cette section, nous reprenons le problème de suivi de la propagation de gaz en utilisant un RCSF déjà considéré dans la section 2.4. La région sous surveillance $\mathbb{X} = [-0.5, 0.5] \times [-0.5, 0.5]$ est une région à deux dimensions, avec une aire unité. Une source de gaz placée à l'origine $(0, 0)$ est activée à partir de $\theta = 1$s à $\theta = 15$s. La diffusion de gaz à l'intérieur de cette région est régie par l'équation différentielle suivante rappelée ici :

$$\frac{\partial G(\boldsymbol{x}, \theta)}{\partial \theta} - c\, \nabla_{\boldsymbol{x}}^2 G(\boldsymbol{x}, \theta) = Q(\boldsymbol{x}, \theta),$$

où $G(\boldsymbol{x}, \theta)$ est la densité de gaz en fonction de la position $\boldsymbol{x}$ et de l'instant θ, $\nabla_{\boldsymbol{x}}^2$ est l'opérateur de Laplace, c est la conductivité du milieu fixée à 0.1 dans la suite, et $Q(\boldsymbol{x}, \theta)$ correspond à la quantité de gaz injectée à l'endroit $\boldsymbol{x}$ et à l'instant θ.

Nous utilisons $N = 100$ capteurs déployés uniformément dans la région $\mathbb{X}$, chacun collecte une série temporelle de 15 mesures, entre $\theta = 1$s et $\theta = 15$s. Le capteur k, à une position notée $\boldsymbol{x}_k \in \mathbb{X}$, mesure une quantité de gaz désignée par $y_{k,\theta}$ à l'instant θ. Nous visons à réduire la dimension de la série temporelle des mesures. Nous considérons une plage prédéterminée de la communication dans le RCSF : deux nœuds sont considérés connectés quand ils sont distants de moins de δ unités, définissant ainsi le voisinage d'un capteur k selon :

$$\mathcal{V}_k = \{l \ : \ \|\boldsymbol{x}_k - \boldsymbol{x}_l\| < \delta\}.$$

Dans nos expériences, nous avons fixé ce seuil à $\delta = 0.38$.

Quant au choix des paramètres du taux d'apprentissage $\eta_{k,t}$, nous utilisons une approche de validation croisée, en variant le pas selon trois séries de valeurs, la première étant $0.1, 0.01, \cdots, 0.1 \cdot 10^{-12}$, la deuxième $0.25, 0.025, \cdots, 0.25 \cdot 10^{-12}$, et la troisième $0.5, 0.05, \cdots, 0.5 \cdot 10^{-12}$. Puis, nous choisissons la valeur du pas qui mène au meilleur résultat dans la stratégie non coopérative. Nous considérons une valeur de pas $\eta_1 = 0.0025$ pour le premier axe principal avec l'algorithme basé sur Oja. La même valeur du pas est prise pour les deux stratégies de coopération : adapter-puis-combiner et combiner-puis-adapter. Pour l'algorithme TI, nous utilisons $\eta_1 = 0.025$, pour l'algorithme OJAn $\eta_1 = 0.005$, pour l'algorithme LUO $\eta_1 = 0.0025$, avec les stratégies coopératives et non coopératives.

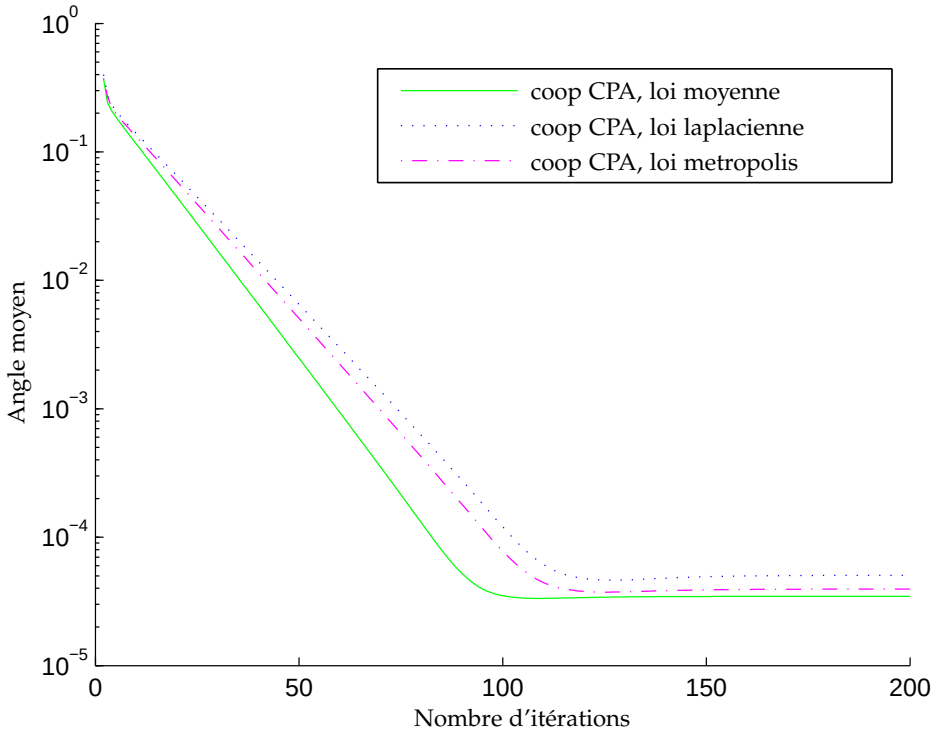

FIGURE 3.2 – *Analyse de convergence du critère de Oja pour les différentes lois des coefficients* a_{kl}.

L'influence des coefficients de combinaison a_{kl} et c_{lk} sur la convergence est illustrée respectivement dans la FIGURE 3.2 et la FIGURE 3.3, avec la stratégie CPA appliquée pour le critère d'Oja. La FIGURE 3.2 utilise le cas simplifié où aucun échange d'information ne s'effectue entre les nœuds, à l'exception de l'étape d'agrégation ; en d'autres termes, nous utilisons $c_{kk} = 1$, et $c_{lk} = 0$ pour chaque $k \neq l$, pour $l, k = 1, 2, \cdots, 100$. La loi moyenne performe légèrement mieux que les autres lois. Lorsque la loi moyenne est utilisée pour a_{kl}, l'influence du choix de c_{lk} est étudiée dans la FIGURE 3.3, en comparant les lois moyenne, laplacienne, et Métropolis, ainsi que le cas de sans échange d'information. Elle montre que toutes les lois ont essentiellement les mêmes performances. Pour ces raisons, la stratégie "sans échange d'information" est utilisée dans ce qui suit parce qu'elle a la plus petite complexité de calcul.

Les FIGURE 3.4-FIGURE 3.8 montrent la convergence de chaque stratégie pour les algorithmes Oja, la théorie de l'information, le quotient de Rayleigh, le OJAn et LUO, en termes de l'angle Θ pour le premier axe principal, en moyenne sur les N nœuds. Les stratégies confrontées sont la stratégie non coopérative, les stratégies coopératives combiner-puis-adapter (CPA) et adapter-puis-combiner (APC), et la stratégie de consensus. En outre, la FIGURE 3.4 montre que toutes les stratégies proposées surpassent l'algorithme de PCADID, et les stratégies coopératives surpassent les algorithmes Late/Early gossip. Nous notons que les simulations ont montré que les résultats de Late et Early gossip sont dépendants du critère du choix des capteurs à chaque itération. D'autre part, les courbes de convergence montrent que la stratégie non coopérative est surpassée

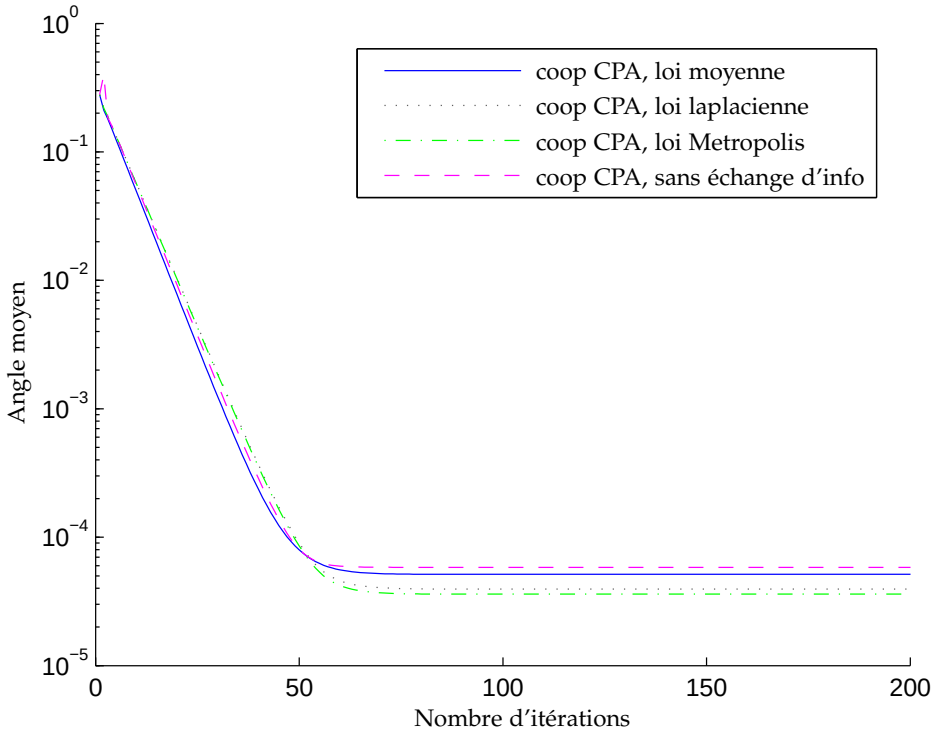

FIGURE 3.3 – *Analyse de convergence du critère de Oja pour les différentes lois des coefficients c_{lk}.*

par toutes les stratégies de coopération. L'analyse des stratégies de coopération montre que la stratégie adapter-puis-combiner performe un peu mieux que la stratégie combiner-puis-adapter. L'efficacité globale des stratégies coopératives est représentée en termes de stabilité. D'autre part, compte tenu de la vitesse de convergence, les figures montrent que l'algorithme LUO a la convergence plus rapide, suivi consécutivement par les algorithmes OJAn, Oja, QR et TI. Les algorithmes Oja et OJAn ont essentiellement le même comportement en termes de vitesse de convergence, alors que QR et LUO ont un comportement conflictuel comme prévu à partir de la TABLE 3.2. Une autre mesure pour la comparaison est le régime permanent de l'angle moyen. L'algorithme TI a le plus grand angle moyen après convergence. Tous les autres algorithmes ont des niveaux comparables. Notons que la stratégie de consensus avec QR, OJAn, et LUO a un angle moyen après convergence inférieur aux autres stratégies, et cette stratégie est plus lente en convergence avec Oja, OJAn et LUO. Pour résumer, nous pouvons dire que les meilleures combinaisons qui permettent un petit angle après convergence, avec une vitesse de convergence acceptable, sont les règles Oja et OJAn avec la stratégie coopérative adapter-puis-combiner.

Perte d'informations

Dans cette section, nous étudions l'impact de l'information manquante dans les RCSF. Ce problème apparaît lorsque, par exemple, un capteur n'a pas acquis toutes les mesures, ou n'a pas reçu correctement les informa-

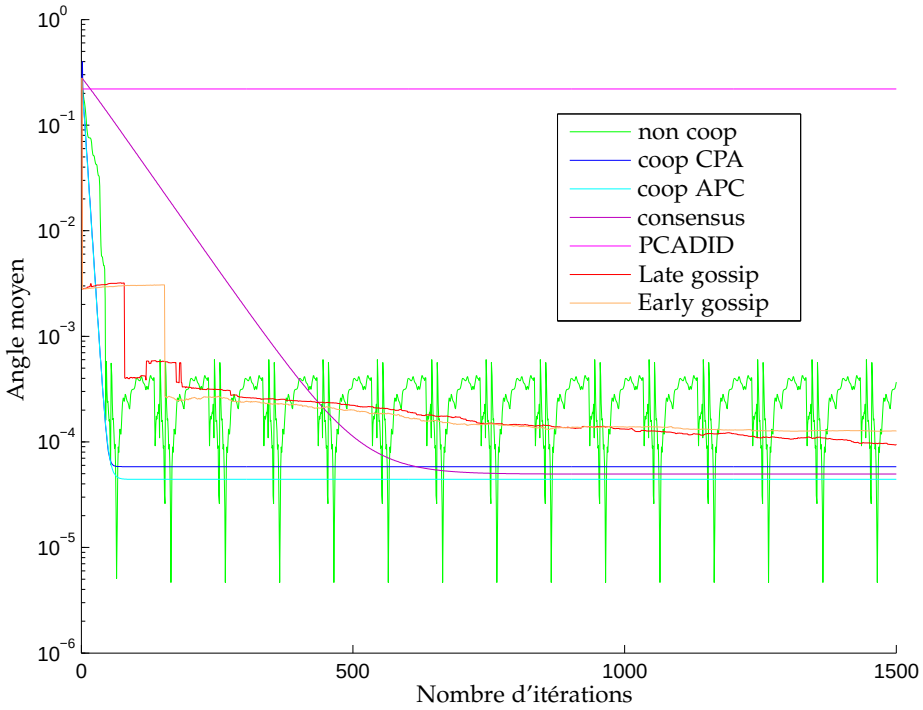

FIGURE 3.4 – *Analyse de convergence du critère d'Oja pour des différentes stratégies.*

tions de ses voisins, ou même en cas de défaut de synchronisation entre les capteurs. Différentes solutions existent pour surmonter la perte de l'information. Sous capacités limitées, nous proposons deux techniques à faible coût calculatoire :

1. Remplacer chaque information perdue par une valeur nulle.
2. Remplacer chaque information perdue par la dernière valeur disponible.

Pour comparer, nous considérons plusieurs paramètres, en faisant varier le nombre de capteurs défectueux et le nombre des mesures perdues correspondantes. La FIGURE 3.9 montre les performances des deux méthodes de récupération proposées sur la convergence des algorithmes de l'ACP en réseau. Bien que l'erreur augmente avec l'augmentation de la perte d'information, nous pouvons voir que le remplacement de chaque information perdue par la dernière valeur disponible est mieux que la récupération par une valeur nulle.

CONCLUSION DU CHAPITRE

Dans ce chapitre, nous avons proposé un cadre de l'ACP en réseau pour estimer l'axe principal en couplant plusieurs stratégies coopératives avec différentes fonctions coût. Pour cette raison, nous avons étudié des stratégies non coopératives et coopératives avec la diffusion de l'information. Une étude théorique sur la convergence de ces méthodes adaptatives a été conduite. Des expériences ont été menées en tenant compte des contraintes imposées dans les réseaux de capteurs. Les résultats ont

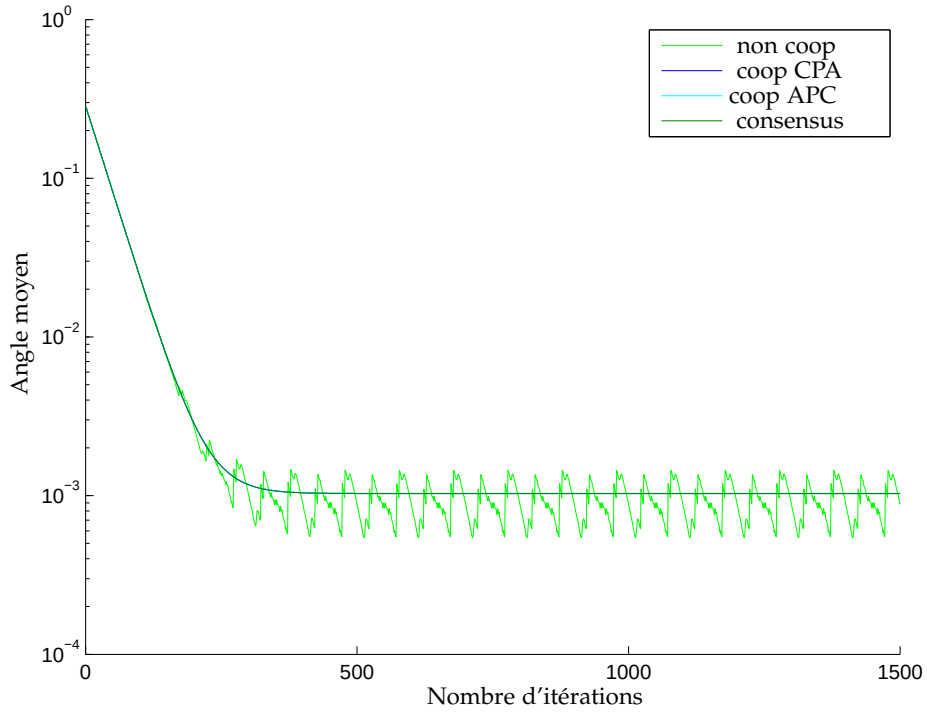

FIGURE 3.5 – *Analyse de convergence du critère de la théorie de l'information pour des différentes stratégies.*

montré la pertinence du cadre proposé sur cette application. Dans le chapitre suivant, nous allons étendre l'approche proposée pour l'estimation de plusieurs axes principaux, en opérant des processus d'orthogonalisation, comme l'orthogonalisation de Gram-Schmidt et l'orthogonalisation par déflation. De plus, nous étudions le cas où nous avons des mesures bruitées ou encore la non synchronisation des capteurs, en proposant comme solution le lissage temporel et le gossip respectivement.

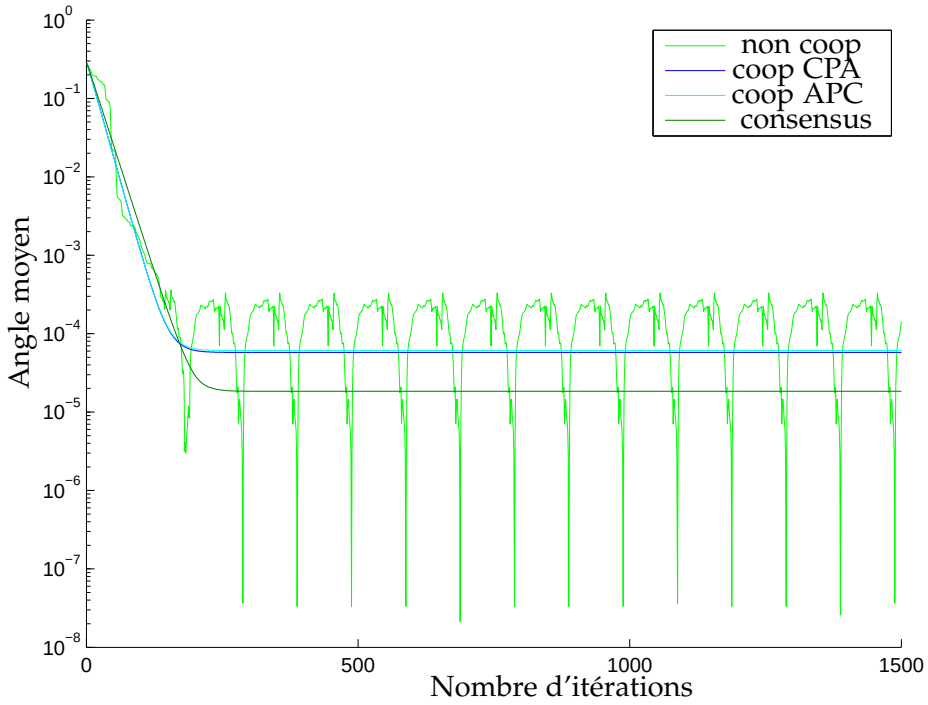

FIGURE 3.6 – *Analyse de convergence du critère du quotient de Rayleigh pour des différentes stratégies.*

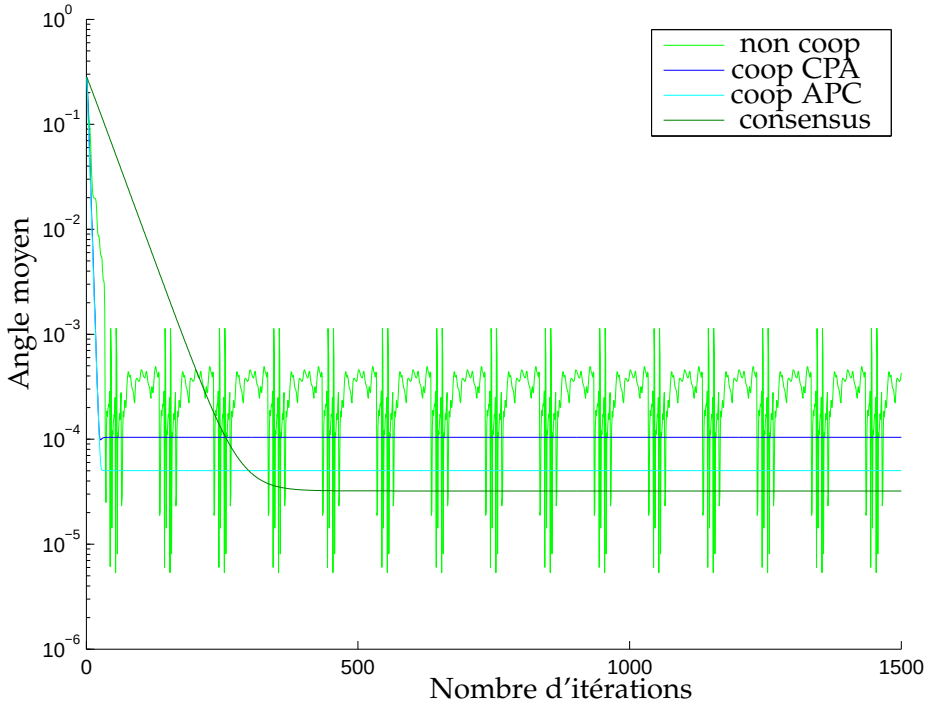

FIGURE 3.7 – *Analyse de convergence du critère de OJAn pour des différentes stratégies.*

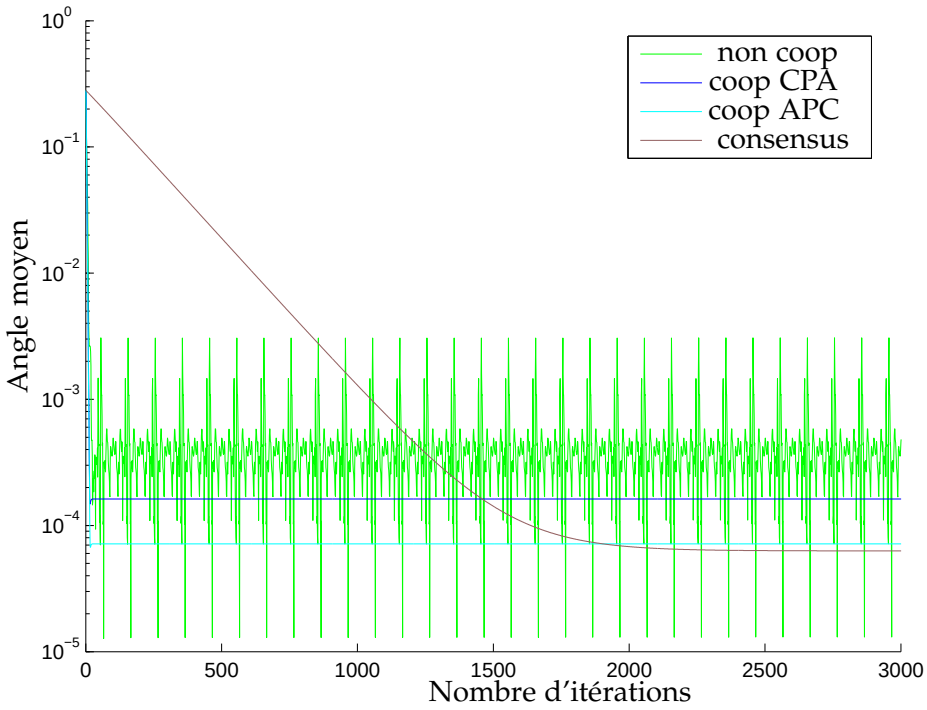

FIGURE 3.8 – *Analyse de convergence du critère de LUO pour des différentes stratégies.*

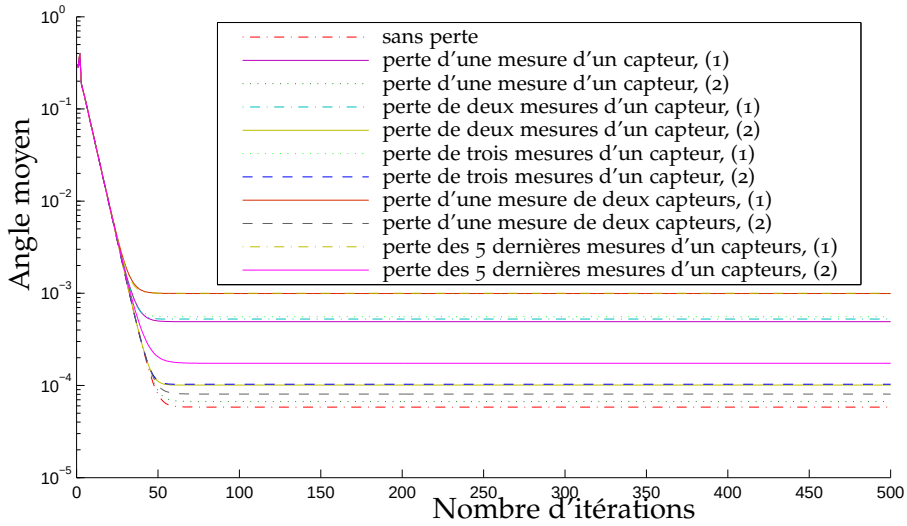

FIGURE 3.9 – *Récupération des informations perdues, en utilisant la stratégie CPA. Dans la légende : (1) : récupérer l'information perdue par une valeur nulle, (2) : récupérer l'information perdue par la dernière valeur disponible.*

Généralisation des stratégies de l'ACP en réseau

4

Sommaire

Dans ce chapitre, nous complétons l'étude de l'ACP en réseau de capteurs du chapitre précédent. Par souci de clarté, nous nous concentrons dans ce chapitre sur la formulation présentée dans la section 3.1 ; une généralisation à d'autres fonctions coût comme celles indiquées dans la section 3.2 est directe. Nous étendons d'abord l'approche proposée pour l'estimation de plusieurs axes principaux, en opérant des processus d'orthogonalisation, comme l'orthogonalisation de Gram-Schmidt et l'orthogonalisation par déflation. Ensuite, nous étudions deux problèmes omniprésents dans les RCSF. Le premier est le problème de désynchronisation entre les nœuds dans un réseau décentralisé. C'est un problème crucial, puisque les stratégies proposées au chapitre 3 nécessitent une synchronisation entre les capteurs. Par conséquent, la synchronisation est nécessaire pour assurer la même échelle de temps pour les horloges locales des capteurs. Ceci exige des oscillateurs pour émettre des signaux à la même fréquence. Pour contourner ce problème, nous combinons la règle basée sur Oja avec la gossip moyenne (Boyd et al. 2006, Asensio-Marco et Beferull-Lozano 2014) où les capteurs coopèrent entre eux pour estimer l'axe principal. La communication entre les capteurs est régie par le concept de gossip pour alléger les contraintes sur le synchronisme de transmission-réception. Le second problème étudié dans ce

chapitre est le problème de mesures bruitées. Afin de réduire l'effet du bruit, nous étudions la coopération temporelle (par lissage) pour les différentes stratégies coopératives susmentionnées. Pour cela, nous réexaminons, dans le cadre de l'ACP en réseau, le processus de lissage étudié dans la littérature de filtrage adaptatif linéaire par Cattivelli et Sayed (2010).

4.1 Extraction de multiples axes principaux

Dans cette section, nous étendons le cadre proposé dans le chapitre précédent à l'estimation de plusieurs axes principaux. A cette fin, nous rappelons le modèle général des stratégies APC et CPA pour l'estimation du premier axe principal :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t} &= \sum_{l\in\mathcal{V}_k} a_{1,kl}\, \boldsymbol{w}_{l,t-1},\\
\boldsymbol{\psi}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk} \nabla_{w} J_l(\boldsymbol{\phi}_{k,t}),\\
\boldsymbol{w}_{k,t} &= \sum_{l\in\mathcal{V}_k} a_{2,kl}\, \boldsymbol{\psi}_{l,t}.
\end{aligned}$$

Nous désignons par

$$\boldsymbol{W}_t = [\boldsymbol{w}_{1,t} \quad \boldsymbol{w}_{2,t} \quad \cdots \quad \boldsymbol{w}_{r,t}]^\top$$

la matrice de taille $(r \times p)$ des premiers r axes principaux estimés à l'itération t, triés dans l'ordre décroissant de leurs valeurs propres. Soit

$$\boldsymbol{z}_{W,k} = [z_{\boldsymbol{w}_1,k} \quad z_{\boldsymbol{w}_2,k} \quad \cdots \quad z_{\boldsymbol{w}_r,k}]^\top,$$

où $z_{\boldsymbol{w}_j,k} = \boldsymbol{w}_{j,t}^\top \boldsymbol{y}_k$. Nous combinons les règles de mise à jour étudiées dans les sections 3.1.1 et 3.1.2 avec plusieurs processus d'orthogonalisation, dont le processus d'orthogonalisation bien connu de Gram-Schmidt et l'orthogonalisation par déflation (Möller 2006). D'autres procédés d'orthogonalisation peuvent également être adaptées à notre cadre de l'ACP en réseau, comme l'orthogonalisation symétrique (Srivastava 2000) qui, cependant, n'est pas appropriée dans une configuration de RCSF en raison de sa haute complexité de calcul.

4.1.1 Processus d'orthogonalisation de Gram-Schmidt

Nous utilisons une approche de type hebbien généralisée en s'inspirant des travaux de Sanger pour l'analyse en composantes principales linéaires (Sanger 1989; 1994). Dans la stratégie non coopérative, nous écrivons la règle de mise à jour du $j^{\text{ème}}$ axe principal comme suit :

$$\boldsymbol{w}_{j,t} = \boldsymbol{w}_{j,t-1} + \eta_t \left(\boldsymbol{y}_k\, z_{\boldsymbol{w}_j,k} - z_{\boldsymbol{w}_j,k} \sum_{l=1}^{j} z_{\boldsymbol{w}_j,l}\, \boldsymbol{w}_{l,t-1} \right). \tag{4.1}$$

Il est clair que l'estimation du $j^{\text{ème}}$ axe principal $\boldsymbol{w}_{j,t}$ à l'itération t implique les estimations précédentes (à l'itération $t-1$) des axes principaux d'ordre

inférieur, c'est-à-dire $\boldsymbol{w}_{l,t-1}$, pour $l = 1, \cdots, j-1$. Cette expression peut s'écrire sous la forme :

$$\boldsymbol{w}_{j,t} = \boldsymbol{w}_{j,t-1} + \eta_t \left(z_{\boldsymbol{w}_j,k} \left(\boldsymbol{y}_k - \sum_{l=1}^{j-1} z_{\boldsymbol{w}_j,l}\, \boldsymbol{w}_{l,t-1} \right) - z^2_{\boldsymbol{w}_j,k}\, \boldsymbol{w}_{j,t-1} \right).$$

C'est le critère de Oja pour $\boldsymbol{w}_{j,t}$ avec une entrée modifiée $\boldsymbol{y}_k - \sum_{l=1}^{j-1} z_{\boldsymbol{w}_j,l}\, \boldsymbol{w}_{l,t-1}$. Cette condition produit le $j^{\text{ème}}$ axe principal. En recueillant toutes ces estimations dans une seule matrice $\boldsymbol{W}_t$, nous obtenons la règle de mise à jour suivante :

$$\boldsymbol{W}_t = \boldsymbol{W}_{t-1} + \eta_t \left(\boldsymbol{z}_{\boldsymbol{W}_{t-1},k}\, \boldsymbol{y}_k^\top - \mathrm{LT}(\boldsymbol{z}_{\boldsymbol{W}_{t-1},k}\, \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top)\, \boldsymbol{W}_{t-1} \right), \tag{4.2}$$

où $\mathrm{LT}(\cdot)$ transforme son argument en une matrice triangulaire inférieure en mettant à zéro les éléments au-dessus de sa diagonale. A noter que le taux d'apprentissage n'a pas besoin d'être le même pour tous les axes principaux.

Dans les stratégies de coopération, nous désignons par $\boldsymbol{\Phi}_{k,t} = [\boldsymbol{\phi}_{1,k,t} \quad \boldsymbol{\phi}_{2,k,t} \cdots \boldsymbol{\phi}_{r,k,t}]^\top$ la matrice de taille $(r \times p)$ des r estimations intermédiaires. En forme matricielle, la règle de mise à jour associée à la stratégie adapter-puis-combiner est la suivante

$$\begin{aligned} \boldsymbol{\Phi}_{k,t} &= \boldsymbol{W}_{k,t-1} + \eta_{k,t} \left(\boldsymbol{z}_{\boldsymbol{W}_{t-1},k} \boldsymbol{y}_k^\top - \mathrm{LT}(\boldsymbol{z}_{\boldsymbol{W}_{t-1},k} \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top)\, \boldsymbol{W}_{k,t-1} \right), \\ \boldsymbol{W}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\Phi}_{l,t}. \end{aligned}$$

La règle de mise à jour de la stratégie combiner-puis-adapter devient

$$\begin{aligned} \boldsymbol{\Phi}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{W}_{l,t-1}, \\ \boldsymbol{W}_{k,kt} &= \boldsymbol{\Phi}_{k,t} + \eta_{k,t} \left(\boldsymbol{z}_{\boldsymbol{W}_{t-1},k} \boldsymbol{y}_k^\top - \mathrm{LT}(\boldsymbol{z}_{\boldsymbol{W}_{t-1},k} \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top)\, \boldsymbol{\Phi}_{k,t} \right). \end{aligned}$$

Le modèle général s'écrit alors :

$$\begin{aligned} \boldsymbol{\Phi}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{1,kl}\, \boldsymbol{W}_{l,t-1}, \\ \boldsymbol{\Psi}_{k,kt} &= \boldsymbol{\Phi}_{k,t} + \eta_{k,t} \left(\boldsymbol{z}_{\boldsymbol{W}_{t-1},k} \boldsymbol{y}_k^\top - \mathrm{LT}(\boldsymbol{z}_{\boldsymbol{W}_{t-1},k} \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top)\, \boldsymbol{\Phi}_{k,t} \right) \\ \boldsymbol{W}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{2,kl}\, \boldsymbol{\Psi}_{l,t}, \end{aligned}$$

avec $\boldsymbol{\Psi}_{k,t} = [\boldsymbol{\psi}_{1,k,t} \quad \boldsymbol{\psi}_{2,k,t} \quad \cdots \quad \boldsymbol{\psi}_{r,k,t}]^\top$ la matrice de taille $(r \times p)$ des r estimations intermédiaires.

4.1.2 Orthogonalisation par déflation

De la même manière qu'avec l'orthogonalisation de Gram-Schmidt, l'orthogonalisation par déflation met à jour toutes les estimations à chaque itération. Cependant, nous procédons en deux étapes consécutives, à chaque itération. Tout d'abord, les estimations intermédiaires $\boldsymbol{w}_{j,\star}$, pour $j = 1, 2, \cdots, r$, sont déterminées indépendamment à l'aide de la règle de mise à jour, selon

$$\boldsymbol{w}_{j,\star} = \boldsymbol{w}_{j,t-1} + \eta_t \left(\boldsymbol{y}_k\, z_{\boldsymbol{w}_j,k} - z^2_{\boldsymbol{w}_j,k}\, \boldsymbol{w}_{j,t-1} \right). \tag{4.3}$$

Soit $\boldsymbol{W}_\star = [\boldsymbol{w}_{1,\star} \quad \boldsymbol{w}_{2,\star} \quad \cdots \quad \boldsymbol{w}_{r,\star}]^\top$ la matrice des estimations intermédiaires. Nous pouvons écrire :

$$\boldsymbol{W}_\star = \boldsymbol{W}_{t-1} + \eta_t \Big(\boldsymbol{y}_k \boldsymbol{z}_{\boldsymbol{W},k}^\top - \boldsymbol{W}_{t-1} \mathrm{diag}(z_{\boldsymbol{W},k}\, \boldsymbol{z}_{\boldsymbol{W},k}^\top) \Big),$$

où $\mathrm{diag}(\cdot)$ transforme son argument en une matrice diagonale en mettant tous les éléments non diagonaux à zéro.

Ensuite, chaque estimation est mise à jour par le nœud k en utilisant les estimations intermédiaires de (4.3) comme suit :

$$\boldsymbol{w}_{j,t} = \boldsymbol{w}_{j,\star} - \sum_{l=1}^{j-1} (\boldsymbol{w}_{l,t-1}^\top \boldsymbol{w}_{j,\star})\, \boldsymbol{w}_{l,t-1}.$$

En arrangeant ces r vecteurs $\boldsymbol{w}_{j,t}$ dans une matrice $\boldsymbol{W}_t$, l'expression de la stratégie non coopérative est écrite sous forme matricielle comme suit :

$$\boldsymbol{W}_t = \boldsymbol{W}_\star - \boldsymbol{W}_{t-1} \Big(\boldsymbol{W}_{t-1}^\top \boldsymbol{W}_\star - \mathrm{LT}(\boldsymbol{W}_{t-1}^\top \boldsymbol{W}_\star) \Big).$$

Une normalisation de $\boldsymbol{w}_{j,t}$ est nécessaire pour la procédure d'orthogonalisation, en opérant la projection obtenue en remplaçant $\boldsymbol{w}_{j,t}$ par $\frac{\boldsymbol{w}_{j,t}}{\|\boldsymbol{w}_{j,t}\|}$.

Nous notons que ce système pourrait être appliqué d'une façon séquentielle : le nœud k extrait les axes principaux en appliquant la règle de mise à jour de la matrice. Pour le $j^{\text{ème}}$ axe principal, avec $j = 1, 2, \cdots, r$, l'estimation intermédiaire définie dans (4.3) est utilisée. Ensuite, le nœud en question met à jour le $j^{\text{ème}}$ axe principal selon la règle de mise à jour suivante :

$$\boldsymbol{w}_{j,t} = \boldsymbol{w}_\star - \sum_{l=1}^{j-1} (\boldsymbol{w}_{l,\infty}^\top \boldsymbol{w}_{j,\star})\, \boldsymbol{w}_{l,\infty}, \tag{4.4}$$

où $\boldsymbol{w}_{l,\infty}$ est le $l^{\text{ème}}$ axe de principal obtenu après convergence de $\boldsymbol{w}_{l,t}$, pour $l = 1, 2, \cdots, j-1$. Cependant, ce processus est mal adapté pour les applications des réseaux en temps réel, car l'extraction du $j^{\text{ème}}$ axe principal ne peut pas commencer avant la convergence de $\boldsymbol{w}_{1,t}, \boldsymbol{w}_{2,t}, \cdots, \boldsymbol{w}_{j-1,t}$.

Pour les stratégies de coopération, l'orthogonalisation par déflation peut être facilement intégrée dans la stratégie adapter-puis-combiner, avec

$$\begin{cases} \boldsymbol{\Phi}_{k,\star} &= \boldsymbol{W}_{k,t-1} + \eta_{k,t} \Big(\boldsymbol{y}_k \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top - \boldsymbol{W}_{k,t-1} \mathrm{diag}(z_{\boldsymbol{W}_{t-1},k}\, \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top) \Big) \\ \boldsymbol{\Phi}_{k,t} &= \boldsymbol{\Phi}_{k,\star} - \boldsymbol{\Phi}_{k,t-1} \Big(\boldsymbol{\Phi}_{k,t-1}^\top \boldsymbol{\Phi}_{k,\star} - \mathrm{LT}(\boldsymbol{\Phi}_{k,t-1}^\top \boldsymbol{\Phi}_{k,\star}) \Big) \\ \boldsymbol{W}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\Phi}_{l,t}, \end{cases}$$

ainsi que la stratégie combiner-puis-adapter, avec

$$\boldsymbol{\Phi}_{k,t-1} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{W}_{l,t-1}$$

$$\begin{cases} \boldsymbol{W}_{k,\star} &= \boldsymbol{\Phi}_{k,t-1} + \eta_{k,t} \Big(\boldsymbol{y}_k \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top - \boldsymbol{\Phi}_{k,t-1} \mathrm{diag}(z_{\boldsymbol{W}_{t-1},k}\, \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top) \Big) \\ \boldsymbol{W}_{k,t} &= \boldsymbol{W}_{k,\star} - \boldsymbol{W}_{k,t-1} \Big(\boldsymbol{W}_{k,t-1}^\top \boldsymbol{W}_{k,\star} - \mathrm{LT}(\boldsymbol{W}_{k,t-1}^\top \boldsymbol{W}_{k,\star}) \Big). \end{cases}$$

Le modèle général s'écrit alors :

$$
\begin{aligned}
\boldsymbol{\Phi}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{1,kl}\, \boldsymbol{W}_{l,t-1}, \\
\begin{cases}
\boldsymbol{\Psi}_{k,\star} &= \boldsymbol{\Phi}_{k,t-1} + \eta_{k,t} \Big(\boldsymbol{y}_k \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top - \boldsymbol{\Phi}_{k,t-1} \mathrm{diag}(\boldsymbol{z}_{\boldsymbol{W}_{t-1},k}\, \boldsymbol{z}_{\boldsymbol{W}_{t-1},k}^\top) \Big) \\
\boldsymbol{\Psi}_{k,t} &= \boldsymbol{\Psi}_{k,\star} - \boldsymbol{\Psi}_{k,t-1} \Big(\boldsymbol{\Psi}_{k,t-1}^\top \boldsymbol{\Psi}_{k,\star} - \mathrm{LT}(\boldsymbol{\Psi}_{k,t-1}^\top \boldsymbol{\Psi}_{k,\star}) \Big).
\end{cases} \\
\boldsymbol{W}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{2,kl}\, \boldsymbol{\Psi}_{l,t},
\end{aligned}
$$

4.2 Gossip

Les réseaux centralisés comme le réseau téléphonique fixe ou Internet permettent une communication fiable et de hautes performances. Les nœuds communiquent entre eux à de très longues distances et à des vitesses très élevées. Contrairement aux réseaux centralisés fixes, les réseaux décentralisés, notamment les réseaux sans fil, nécessitent des algorithmes adaptés afin de surmonter les difficultés en calcul distribué et en échange d'informations (Hatano et Mesbahi 2004). En effet, il n'y a pas de centre de fusion pour faciliter la communication, le calcul et la synchronisation entre les différents nœuds. D'autre part, la topologie du réseau peut ne pas être complètement connue aux nœuds du réseau, et peut même changer parce que les nœuds peuvent rejoindre ou quitter le réseau. Il y a aussi des contraintes sur la puissance de calcul et les ressources d'énergie spécialement pour les réseaux de capteurs sans fil.

La conception d'algorithmes de synchronisation a été étudié dans de nombreux travaux de recherche, notamment les travaux de Sivrikaya et Yener (2004), Lu et Chen (2005). Cependant, ces techniques nécessitent une consommation importante des ressources du réseau qui sont très souvent limitées aussi bien en calcul qu'en communication. Pour surmonter ce problème, des algorithmes distribués et asynchrones ont été étudiés, où chaque nœud échange des informations avec un seul voisin dans un intervalle de temps. Les algorithmes de gossip, également appelés algorithmes épidémiques, sont basée sur une échange d'information asynchrone (Boyd et al. 2006, Asensio-Marco et Beferull-Lozano 2014). Ils permettent une répartition de la charge de calcul puisqu'à chaque instance de temps, un nœud communique avec un seul nœud voisin choisi au hasard. Les algorithmes de gossip sont simples, évolutifs et robustes contre les défaillances des nœuds, la perte de messages, et les perturbations temporaires du réseau.

4.2.1 Gossip moyenne

Nous sommes intéressés, dans cette section, par le cas où les nœuds estiment la moyenne d'un certain état $\boldsymbol{w}_k$ (par exemple une certaine caractéristique extraite à partir des mesures) du nœud k, $k = 1, 2, \cdots, N$. Dans un réseau centralisé, les nœuds envoient leurs états $\boldsymbol{w}_k$, $k = 1, 2, \cdots, N$ à un CF où la moyenne est calculée par la formule $\frac{1}{N} \sum_{k=1}^{N} \boldsymbol{w}_k$. Dans une stratégie non coopérative, les nœuds communiquent selon un procédé de routage, comme suit : chaque nœud k reçoit une estimation à partir du

nœud précédent, lui ajoute $\frac{1}{N}\boldsymbol{w}_k$, et la transmet au nœud suivant. Cette technique nécessite un système de routage bien défini et une transmission/réception synchrone des informations échangées. Les algorithmes de gossip fournissent une élégante solution asynchrone pour surmonter ces problèmes, sans la nécessité d'un CF.

Les algorithmes de gossip utilisent un protocole asynchrone avec une faible complexité de calcul, décrit comme suit (Demers et al. 1987, Kempe et al. 2003). A l'itération t, un nœud k, sélectionné arbitrairement, choisit aléatoirement un de ses voisins, désigné dans la suite par le nœud l. Ce dernier envoie son état au nœud k qui produit un nouvel état selon la règle convex suivante :

$$\boldsymbol{w}_{k,t} = (1-\epsilon)\boldsymbol{w}_{k,t-1} + \epsilon\boldsymbol{w}_{l,t-1},$$

pour un paramètre de mélange $\epsilon \in [0,1]$. Les états des autres nœuds restent inchangés, à savoir $\boldsymbol{w}_{j,t} = \boldsymbol{w}_{j,t-1}$ pour l'ensemble des indices $j \neq k$. C'est le gossip asymétrique, décrit en détail dans (Fagnani et Zampieri 2008a).

Contrairement au gossip asymétrique, le gossip symétrique étudié dans (Boyd et al. 2006) utilise un protocole de communication symétrique entre les nœuds. Dans ce cas, les deux nœuds k et l mettent à jour leurs états en utilisant le paramètre de mélange $\epsilon \in [0,1]$, comme suit :

$$\begin{aligned}\boldsymbol{w}_{k,t} &= (1-\epsilon)\boldsymbol{w}_{k,t-1} + \epsilon\boldsymbol{w}_{l,t-1},\\ \boldsymbol{w}_{l,t} &= \epsilon\boldsymbol{w}_{k,t-1} + (1-\epsilon)\boldsymbol{w}_{l,t-1}.\end{aligned}$$

Les états des autres nœuds restent inchangés. Nous notons que cette mise à jour conserve la somme totale, et donc la moyenne des estimations des nœuds (Fagnani et Zampieri 2008b), puisque nous avons

$$\begin{aligned}\sum_{i=1}^{N}\boldsymbol{w}_{ji,t} =&(1-\epsilon)\boldsymbol{w}_{k,t-1} + \epsilon\boldsymbol{w}_{l,t-1} + \epsilon\boldsymbol{w}_{k,t-1} + (1-\epsilon)\boldsymbol{w}_{l,t-1} + \sum_{\substack{i=1\\ i\neq k,l}}^{N}\boldsymbol{w}_{i,t-1}\\ &= \sum_{i=1}^{N}\boldsymbol{w}_{i,t-1}.\end{aligned}$$

Notons que cette propriété n'est pas vérifiée par l'algorithme de gossip asymétrique. Ceci est la cause principale pour laquelle le gossip symétrique surpasse le gossip asymétrique, mais au prix d'une consommation d'énergie plus élevée en communication et en calcul.

4.2.2 Gossip pour l'ACP

En stratégie coopérative, chaque nœud k a accès aux informations de son voisinage $\mathcal{V}_k$, qui est le sous-ensemble de nœuds actuellement connectés au nœud k. Nous visons à minimiser la fonction coût $\sum_{k=1}^{N} J_k(\boldsymbol{w})$, où $J_k(\boldsymbol{w})$ est la fonction coût au nœud k, par exemple selon (3.1). La fonction coût globale du nœud k peut être décomposée comme suit

$$\sum_{l=1}^{N} J_l(\boldsymbol{w}) = J_k(\boldsymbol{w}) + \sum_{\substack{l=1\\ l\neq k}}^{N} J_l(\boldsymbol{w}). \tag{4.5}$$

Le premier terme du côté droit de l'égalité est connu au niveau du nœud en question, tandis que le second devrait être estimé en utilisant les informations des voisins du nœud k ; ainsi, la somme ci-dessus est-elle limitée à son voisinage. Avec la même motivation de la section 3.1.2, nous avons :

$$\sum_{\substack{l=1 \\ l\neq k}}^{N} J_l(\boldsymbol{w}) \approx \sum_{j\in\mathcal{V}_k} b_{jk} \|\boldsymbol{w}-\boldsymbol{w}_*\|^2,$$

où les paramètres b_{lk} contrôlent le compromis entre la précision et la finesse de la solution. En utilisant le mode gossip entre les nœuds, à chaque itération, le nœud k communique avec un seul de ses voisins, désigné par le nœud l. Par conséquent, $b_{lk} = \nu$ où ν est une constante et $b_{jk} = 0$ pour l'ensemble des indices $j \in \mathcal{V}_k \backslash \{l\}$. Nous avons ainsi :

$$\sum_{\substack{l=1 \\ l\neq k}}^{N} J_l(\boldsymbol{w}) \approx \nu \|\boldsymbol{w}-\boldsymbol{w}_*\|^2.$$

L'utilisation de cette approximation dans (4.5) nous permet de réécrire la fonction coût globale au nœud k comme suit :

$$J_k^{\text{glob}}(\boldsymbol{w}) = J_k(\boldsymbol{w}) + \nu\|\boldsymbol{w}-\boldsymbol{w}_*\|^2. \tag{4.6}$$

Afin de minimiser cette fonction coût, le nœud k applique la descente de gradient sur $J_k^{\text{glob}}(\boldsymbol{w})$ avec :

$$\boldsymbol{w}_{k,t} = \boldsymbol{w}_{k,t-1} - \eta_{k,t}\, \nabla_{\boldsymbol{w}} J_k^{\text{glob}}(\boldsymbol{w}_{k,t-1}). \tag{4.7}$$

Ici, $\eta_{k,t}$ est le taux d'apprentissage du nœud k à l'itération t. En remplaçant $J_k^{\text{glob}}(\boldsymbol{w})$ par son expression dans (4.6), nous obtenons :

$$\boldsymbol{w}_{k,t} = \boldsymbol{w}_{k,t-1} - \eta_{k,t}\nabla_{\boldsymbol{w}} J_k(\boldsymbol{w}_{t-1}) + \eta_{k,t}\nu(\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}).$$

Dans cette expression, $\nabla_{\boldsymbol{w}} J_k(\boldsymbol{w}_{t-1})$ désigne le gradient de la fonction coût (3.1). Dans ce cas, nous obtenons

$$\boldsymbol{w}_{k,t} = \boldsymbol{w}_{k,t-1} + \eta_{k,t}\Big(\boldsymbol{y}_k\, z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k}\, \boldsymbol{w}_{k,t-1}\Big) + \eta_{k,t}\nu(\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}). \tag{4.8}$$

Cette règle de mise à jour pour $\boldsymbol{w}_{k,t}$ consiste à ajouter deux termes de correction à la précédente estimation $\boldsymbol{w}_{k,t-1}$. Nous décomposons ce calcul en deux étapes successives en incluant à chaque fois un terme de correction. Dans ce qui suit, nous décrivons deux approches de mise à jour.

Première approche

Nous exprimons la règle de mise à jour (4.8) comme suit :

$$\begin{cases} \boldsymbol{\phi}_{k,t} &= \boldsymbol{w}_{k,t-1} + \eta_{k,t}\left(\boldsymbol{y}_k z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k}\boldsymbol{w}_{k,t-1}\right), \\ \boldsymbol{w}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t}\,\nu(\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}). \end{cases}$$

Pour les mêmes raisons que celles indiquées dans la section 3.1.2, nous remplaçons $\boldsymbol{w}_{l,t-1}$ par $\boldsymbol{\phi}_{l,t}$ dans la première étape, et $\boldsymbol{w}_*$ par $\boldsymbol{w}_{l,t-1}$ dans la seconde étape. Cette dernière est alors remplacée par :

$$\begin{aligned}\boldsymbol{w}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t} \sum_{l \in \mathcal{V}_k \setminus \{k\}} b_{lk} \left(\boldsymbol{\phi}_{l,t} - \boldsymbol{\phi}_{k,t}\right) \\ &= (1 - \eta_{k,t}\nu)\, \boldsymbol{\phi}_{k,t} + \eta_{k,t}\nu\, \phi_{l,t} \\ &= (1 - \epsilon)\, \boldsymbol{\phi}_{k,t} + \epsilon\, \phi_{l,t}.\end{aligned}$$

Nous obtenons alors la règle de mise à jour de la stratégie adapter-puis-gossip-asymétrique (APGA) :

$$\left\{\begin{array}{ll} \boldsymbol{\phi}_{k,t} & = \boldsymbol{w}_{k,t-1} + \eta_{k,t} \left(\boldsymbol{y}_k z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k} \boldsymbol{w}_{k,t-1}\right), \\ \boldsymbol{w}_{k,t} & = (1-\epsilon)\boldsymbol{\phi}_{k,t} + \epsilon \boldsymbol{\phi}_{l,t}. \end{array}\right.$$

Motivé par le gossip moyenne symétrique où la mise à jour est appliquée simultanément par le nœud k et son voisin l, nous appliquons les équations de APGA sur les deux nœuds. Par conséquent, nous obtenons la règle de mise à jour de la stratégie adapter-puis-gossip-symétrique (APGS) :

$$\left\{\begin{array}{ll} \boldsymbol{\phi}_{k,t} & = \boldsymbol{w}_{k,t-1} + \eta_{k,t} \left(\boldsymbol{y}_k z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k} \boldsymbol{w}_{k,t-1}\right), \\ \boldsymbol{\phi}_{l,t} & = \boldsymbol{w}_{l,t-1} + \eta_{l,t} \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{t-1},l} - z^2_{\boldsymbol{w}_{t-1},l} \boldsymbol{w}_{l,t-1}\right), \\ \boldsymbol{w}_{k,t} & = (1-\epsilon)\boldsymbol{\phi}_{k,t} + \epsilon \boldsymbol{\phi}_{l,t}, \\ \boldsymbol{w}_{l,t} & = \epsilon \boldsymbol{\phi}_{k,t} + (1-\epsilon)\boldsymbol{\phi}_{l,t}. \end{array}\right.$$

Seconde approche

Dans cette stratégie, nous exprimons la règle de mise à jour (4.8) comme suit :

$$\left\{\begin{array}{ll} \boldsymbol{\phi}_{k,t} & = \boldsymbol{w}_{k,t-1} + \eta_{k,t}\, \nu(\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}) \\ \boldsymbol{w}_{k,t} & = \boldsymbol{\phi}_{k,t} + \eta_{k,t} \left(\boldsymbol{y}_k\, z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k}\, \boldsymbol{w}_{k,t-1}\right). \end{array}\right.$$

Pour les mêmes raisons indiquées dans la première approche, nous remplaçons dans la première étape $\boldsymbol{w}_*$ par $\boldsymbol{w}_{l,t-1}$, et dans la seconde étape, nous remplaçons $\boldsymbol{w}_{l,t-1}$ par $\boldsymbol{\phi}_{l,t}$. Nous obtenons la règle de mise à jour de la stratégie gossip-asymétrique-puis-adapter (GAPA) :

$$\left\{\begin{array}{ll} \boldsymbol{\phi}_{k,t} & = (1-\epsilon)\boldsymbol{w}_{k,t-1} + \epsilon \boldsymbol{w}_{l,t-1}, \\ \boldsymbol{w}_{k,t} & = \boldsymbol{\phi}_{k,t} + \eta_{k,t} \left(\boldsymbol{y}_k\, z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k}\, \boldsymbol{\phi}_{k,t}\right). \end{array}\right. \tag{4.9}$$

et la stratégie gossip-symétrique-puis-adapter (GSPA) :

$$\left\{\begin{array}{ll} \boldsymbol{\phi}_{k,t} & = (1-\epsilon)\boldsymbol{w}_{k,t-1} + \epsilon \boldsymbol{w}_{l,t-1}, \\ \boldsymbol{\phi}_{l,t} & = \epsilon \boldsymbol{w}_{k,t-1} + (1-\epsilon)\boldsymbol{w}_{l,t-1}, \\ \boldsymbol{w}_{k,t} & = \boldsymbol{\phi}_{k,t} + \eta_{k,t} \left(\boldsymbol{y}_k\, z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k}\, \boldsymbol{\phi}_{k,t}\right), \\ \boldsymbol{w}_{l,t} & = \boldsymbol{\phi}_{l,t} + \eta_{l,t} \left(\boldsymbol{y}_l\, z_{\boldsymbol{w}_{t-1},l} - z^2_{\boldsymbol{w}_{t-1},l}\, \boldsymbol{\phi}_{l,t}\right). \end{array}\right.$$

Finalement, le modèle général de ces stratégies est :

$$\begin{cases} \boldsymbol{\phi}_{k,t} &= (1-\epsilon_1)\boldsymbol{w}_{k,t-1} + \epsilon_1 \boldsymbol{w}_{l,t-1}, \\ \boldsymbol{\phi}_{l,t} &= \epsilon_2 \boldsymbol{w}_{k,t-1} + (1-\epsilon_2)\boldsymbol{w}_{l,t-1}, \\ \boldsymbol{\psi}_{k,t} &= \boldsymbol{\phi}_{k,t} + \eta_{k,t}\left(\boldsymbol{y}_k\, z_{\boldsymbol{w}_{t-1},k} - z^2_{\boldsymbol{w}_{t-1},k}\,\boldsymbol{\phi}_{k,t}\right), \\ \boldsymbol{\psi}_{l,t} &= \xi\Big[\boldsymbol{\phi}_{l,t} + \eta_{l,t}\left(\boldsymbol{y}_l\, z_{\boldsymbol{w}_{t-1},l} - z^2_{\boldsymbol{w}_{t-1},l}\,\boldsymbol{\phi}_{l,t}\right)\Big] \\ \boldsymbol{w}_{k,t} &= (1-\epsilon_3)\boldsymbol{\psi}_{k,t} + \epsilon_3 \boldsymbol{\psi}_{l,t}, \\ \boldsymbol{w}_{l,t} &= \epsilon_4 \boldsymbol{\psi}_{k,t} + (1-\epsilon_4)\boldsymbol{\psi}_{l,t}, \end{cases}$$

où $\xi \in \{0,1\}$. Pour l'extraction des axes multiples, nous avons selon le processus d'orthogonalisation de Gram-Schmidt :

$$\begin{cases} \boldsymbol{\Phi}_{k,t} &= (1-\epsilon_1)\,\boldsymbol{W}_{k,t-1} + \epsilon_1\,\boldsymbol{W}_{l,t-1}, \\ \boldsymbol{\Phi}_{l,t} &= \epsilon_2\,\boldsymbol{W}_{k,t-1} + (1-\epsilon_2)\,\boldsymbol{W}_{l,t-1}, \\ \boldsymbol{\Psi}_{k,t} &= \boldsymbol{\Phi}_{k,t} + \eta_{k,t}\Big(z_{\boldsymbol{W}_{t-1},k}\boldsymbol{y}_k^\top - \mathrm{LT}(z_{\boldsymbol{W}_{t-1},k} z^\top_{\boldsymbol{W}_{t-1},k})\boldsymbol{\Phi}_{k,t}\Big), \\ \boldsymbol{\Psi}_{l,t} &= \xi\Big[\boldsymbol{\Phi}_{l,t} + \eta_{l,t}\Big(z_{\boldsymbol{W}_{t-1},l}\boldsymbol{y}_l^\top - \mathrm{LT}(z_{\boldsymbol{W}_{t-1},l} z^\top_{\boldsymbol{W}_{t-1},l})\boldsymbol{\Phi}_{l,t}\Big)\Big], \\ \boldsymbol{W}_{k,t} &= (1-\epsilon_3)\,\boldsymbol{\Psi}_{k,t} + \epsilon_3\,\boldsymbol{\Phi}_{l,t}, \\ \boldsymbol{W}_{l,t} &= \epsilon_4\,\boldsymbol{\Psi}_{k,t} + (1-\epsilon_4)\,\boldsymbol{\Phi}_{l,t}. \end{cases}$$

Selon l'orthogonalisation par déflation, nous avons :

$$\begin{cases} \boldsymbol{\Phi}_{k,t} &= (1-\epsilon_1)\,\boldsymbol{W}_{k,t-1} + \epsilon_1\,\boldsymbol{W}_{l,t-1}, \\ \boldsymbol{\Phi}_{l,t} &= \epsilon_2\,\boldsymbol{W}_{k,t-1} + (1-\epsilon_2)\,\boldsymbol{W}_{l,t-1}, \\ \boldsymbol{\Psi}_{k,\star} &= \boldsymbol{\Phi}_{k,t-1} + \eta_{k,t}\Big(z_k z^\top_{\boldsymbol{W}_{t-1},k} - \boldsymbol{\Phi}_{k,t-1}\mathrm{diag}(z_{\boldsymbol{W}_{t-1},k}\, z^\top_{\boldsymbol{W}_{t-1},k})\Big), \\ \boldsymbol{\Psi}_{k,t} &= \boldsymbol{\Psi}_{k,\star} - \boldsymbol{\Psi}_{k,t-1}\Big(\boldsymbol{\Psi}^\top_{k,t-1}\,\boldsymbol{\Psi}_{k,\star} - \mathrm{LT}(\boldsymbol{\Psi}^\top_{k,t-1}\,\boldsymbol{\Psi}_{k,\star})\Big), \\ \boldsymbol{\Psi}_{l,\star} &= \xi\Big[\boldsymbol{\Phi}_{l,t-1} + \eta_{l,t}\Big(z_l z^\top_{\boldsymbol{W}_{t-1},l} - \boldsymbol{\Phi}_{l,t-1}\mathrm{diag}(z_{\boldsymbol{W}_{t-1},l}\, z^\top_{\boldsymbol{W}_{t-1},l})\Big)\Big], \\ \boldsymbol{\Psi}_{l,t} &= \xi\Big[\boldsymbol{\Psi}_{l,\star} - \boldsymbol{\Psi}_{l,t-1}\Big(\boldsymbol{\Psi}^\top_{l,t-1}\,\boldsymbol{\Psi}_{l,\star} - \mathrm{LT}(\boldsymbol{\Psi}^\top_{l,t-1}\,\boldsymbol{\Psi}_{l,\star})\Big)\Big], \\ \boldsymbol{W}_{k,t} &= (1-\epsilon_3)\,\boldsymbol{\Psi}_{k,t} + \epsilon_3\,\boldsymbol{\Phi}_{l,t}, \\ \boldsymbol{W}_{l,t} &= \epsilon_4\,\boldsymbol{\Psi}_{k,t} + (1-\epsilon_4)\,\boldsymbol{\Phi}_{l,t}. \end{cases}$$

La Table 4.1 montre comment les stratégies se déduisent du modèle général.

Convergence

Toutes les règles de type gossip proposées ci-dessus convergent vers le premier axe principal $\boldsymbol{w}_*$. Cela peut être vérifié en examinant, par

Table 4.1 – *Différents choix de ϵ_1, ϵ_2, ϵ_3, ϵ_4 et ξ correspondant aux différentes stratégies*

ϵ_1	ϵ_2	ϵ_3	ϵ_4	ξ	Stratégie
0	0	ϵ	0	0	APGA
0	0	ϵ	ϵ	1	APGS
ϵ	0	0	0	0	GAPA
ϵ	ϵ	0	0	1	GSPA

exemple, la stratégie GAPA donnée dans (4.9). Lorsque les estimations $\boldsymbol{w}_{k,t}$ pour $k = 1, 2, \cdots, N$, convergent vers un état $\boldsymbol{w}$, dans la première équation de (4.9), nous avons pour $k = 1, 2, \cdots, N$:

$$\boldsymbol{\phi}_{k,t} = (1 - \epsilon)\boldsymbol{w} + \epsilon \boldsymbol{w} = \boldsymbol{w}.$$

Quant à la seconde équation, il s'agit de

$$\boldsymbol{y}_k z_{\boldsymbol{w},k} = z^2_{\boldsymbol{w},k} \boldsymbol{w},$$

à savoir

$$\boldsymbol{y}_k \boldsymbol{y}_k^\top \boldsymbol{w} = z^2_{\boldsymbol{w},k} \boldsymbol{w}.$$

La moyenne sur l'ensemble des données nous conduit au problème bien connu de décomposition en éléments propres de la matrice de covariance, selon $\mathfrak{C}\boldsymbol{w} = \boldsymbol{w}^\top \mathfrak{C}\boldsymbol{w}\, \boldsymbol{w}$, où la valeur propre $\boldsymbol{w}^\top \mathfrak{C}\boldsymbol{w}$ correspond à la valeur à maximiser. Par conséquent, la règle de mise à jour de l'algorithme GAPA converge vers le plus grand vecteur propre de la matrice de covariance, sans la nécessité de la calculer. Une démonstration similaire peut être considérée pour les autres stratégies.

4.3 Mécanisme de lissage

Pour atténuer l'effet du bruit sur les mesures et la communication entre les nœuds du réseau, nous avons recours à un lissage temporel. Dans ce qui suit, nous présentons l'intégration du lissage dans les stratégies adaptatives étudiées dans le présent document.

En plus de la combinaison spatiale entre les nœuds, nous étudions la coopération temporelle pour les stratégies coopératives. Par conséquent, le nœud k a accès non seulement aux données actuelles $\{\boldsymbol{y}_l, z_l, \boldsymbol{w}_{l,t-1}, l \in \mathcal{V}_k\}$, et aux estimations actuelles reçues de ses voisins $\{\boldsymbol{\phi}_{l,t}, l \in \mathcal{V}_k\}$, mais aussi aux P estimations, actuelle et passées, c'est-à-dire $\{\boldsymbol{\phi}_{l,j}, j = t, t-1, \cdots, t-P+1\}$. De cette manière, les estimations précédentes sont lissées. En utilisant la règle de Oja, le coût individuel devient :

$$J_k(\boldsymbol{w}) = \sum_{j=0}^{P-1} q_{kj} \tfrac{1}{4} \|\boldsymbol{y}_k - z_{\boldsymbol{w}_{t-j},k} \boldsymbol{w}\|^2.$$

où chaque q_{kj} est un coefficient non négatif qui représente le poids donné par le nœud k aux données de l'itération $t - j$, avec la règle de convexité suivante :

$$q_{k0} > 0, \qquad \sum_{j-0}^{P=1} q_{kj} = 1, \qquad k = 1, 2, \cdots, N$$

Le coût global est :

$$J_k^{\text{glob}}(\boldsymbol{w}) = \sum_{l \in \mathcal{V}_k} c_{lk} J_l(\boldsymbol{w}) + \sum_{l \in \mathcal{V}_k \setminus \{k\}} b_{lk} \|\boldsymbol{w} - \boldsymbol{w}_*\|^2.$$

La minimisation de cette fonction coût par la descente de gradient sur $J_k^{\text{glob}}(\boldsymbol{w})$ donne :

$$\begin{aligned}
\boldsymbol{w}_{k,t} &= \boldsymbol{w}_{k,t-1} - \eta_{k,t}\,\nabla_{\boldsymbol{w}} J_k^{\text{glob}}(\boldsymbol{w}_{k,t-1}) \\
&= \boldsymbol{w}_{k,t-1} - \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk}\nabla_{\boldsymbol{w}} J_l(\boldsymbol{w}_{k,t-1}) + \eta_{k,t} \sum_{l\in\mathcal{V}_k\setminus\{k\}} b_{lk}(\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}) \\
&= \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk} \sum_{j=0}^{P-1} q_{lj}\big(\boldsymbol{y}_l\, z_{\boldsymbol{w}_{k,t-1-j},l} - z^2_{\boldsymbol{w}_{k,t-1-j},l}\,\boldsymbol{w}_{k,t-1-j}\big) \\
&\quad + \eta_{k,t} \sum_{l\in\mathcal{V}_k\setminus\{k\}} b_{lk}(\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}) \\
&= \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk} q_{l0}\big(\boldsymbol{y}_l\, z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l}\,\boldsymbol{w}_{k,t-1}\big) \\
&\quad + \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk} \sum_{j=1}^{P-1} q_{lj}\big(\boldsymbol{y}_l\, z_{\boldsymbol{w}_{k,t-1-j},l} - z^2_{\boldsymbol{w}_{k,t-1-j},l}\,\boldsymbol{w}_{k,t-1-j}\big) \\
&\quad + \eta_{k,t} \sum_{l\in\mathcal{V}_k\setminus\{k\}} b_{lk}(\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}).
\end{aligned} \tag{4.10}$$

Comme les stratégies précédentes, cette règle de mise à jour de $\boldsymbol{w}_{k,t-1}$ à $\boldsymbol{w}_{k,t}$ consiste à ajouter trois termes de correction : le premier fonctionne dans le sens de la variance maximale, le deuxième associe une régularisation du voisinage et le troisième applique un traitement temporel. La décomposition de ce calcul en trois étapes successives mène à six stratégies possibles, en fonction de l'ordre d'ajout des termes de correction, et qui diffèrent essentiellement dans l'approximation de l'inconnu $\boldsymbol{w}_*$.

4.3.1 Stratégie adaptation-puis-lissage temporel-puis-coopération spatiale (ATS)

A titre d'exemple, nous pouvons écrire l'équation (4.10) comme suit :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t} &= \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk} q_{l0}\Big(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l}\boldsymbol{w}_{k,t-1}\Big), \\
\boldsymbol{\psi}_{k,t} &= \boldsymbol{\phi}_{k,t-1} + \eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk} \sum_{j=1}^{P-1} q_{lj}\big(\boldsymbol{y}_l\, z_{\boldsymbol{w}_{k,t-1-j},l} - z^2_{\boldsymbol{w}_{k,t-1-j},l}\,\boldsymbol{w}_{k,t-1-j}\big), \\
\boldsymbol{w}_{k,t} &= \boldsymbol{\psi}_{k,t} + \eta_{k,t} \sum_{l\in\mathcal{V}_k\setminus\{k\}} b_{lk}(\boldsymbol{w}_* - \boldsymbol{w}_{k,t-1}).
\end{aligned}$$

La première étape met à jour $\boldsymbol{w}_{k,t-1}$ pour avoir la première estimation intermédiaire $\boldsymbol{\phi}_{k,t}$ en utilisant les vecteurs de gradient local du voisinage du nœud k. D'après cette étape, nous avons :

$$\eta_{k,t} \sum_{l\in\mathcal{V}_k} c_{lk} q_{l0}\Big(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1-j},l} - z^2_{\boldsymbol{w}_{k,t-1-j},l}\boldsymbol{w}_{k,t-1-j}\Big) = \boldsymbol{\phi}_{k,t-j} - \boldsymbol{w}_{k,t-1-j}.$$

Nous sommes intéressés au cas où les poids donnés aux estimations passées et actuelles, c'est-à-dire q_{kj}, sont identiques pour tous les nœuds car il n'y a pas de discernement entre les nœuds. En d'autres termes, $q_{kj} = q_{lj}$ pour tout $j = 0, \cdots, P-1$.

La deuxième étape met à jour $\boldsymbol{\phi}_{k,t}$ pour avoir la seconde estimation intermédiaire $\boldsymbol{\psi}_{k,t}$. Par conséquent, la deuxième étape peut être écrite comme suit :

$$\boldsymbol{\psi}_{k,t} = \boldsymbol{\phi}_{k,t} + \sum_{j=1}^{P-1} \frac{q_{kj}}{q_{k0}} \Big(\boldsymbol{\phi}_{k,t-j} - \boldsymbol{w}_{k,t-1-j} \Big).$$

La valeur intermédiaire $\boldsymbol{\phi}_{k,t}$ au nœud k est généralement une meilleure estimation pour $\boldsymbol{w}_*$ que $\boldsymbol{w}_{k,t}$, car elle est obtenue en incorporant des informations des voisins (comme donné par la première étape). Par conséquent, nous remplaçons $\boldsymbol{w}_{k,t-1-j}$ par $\boldsymbol{\phi}_{k,t-1-j}$ dans la deuxième étape :

$$\begin{aligned}
\psi_{k,t} &= \boldsymbol{\phi}_{k,t} + \sum_{j=1}^{P-1} \frac{q_{kj}}{q_{k0}} \boldsymbol{\phi}_{k,t-j} - \sum_{j=1}^{P-1} \frac{q_{kj}}{q_{k0}} \boldsymbol{\phi}_{k,t-1-j} \\
&= \boldsymbol{\phi}_{k,t} + \sum_{j=1}^{P-1} \frac{q_{kj}}{q_{k0}} \boldsymbol{\phi}_{k,t-j} - \sum_{j=0}^{P-2} \frac{q_{k(j+1)}}{q_{k0}} \boldsymbol{\phi}_{k,t-j} \\
&= \Big(1 - \frac{q_{k1}}{q_{k0}} \Big) \phi_{k,t} + \sum_{j=1}^{P-2} \frac{q_{kj} - q_{k(j+1)}}{q_{k0}} \boldsymbol{\phi}_{k,t-j} + \frac{q_{k(P-1)}}{q_{k0}} \boldsymbol{\phi}_{k,t-P+1}.
\end{aligned}$$

Nous introduisons les coefficients non négatifs f_{kj} :

$$f_{kj} = \begin{cases} 1 - \frac{q_{k1}}{q_{k0}}, & \text{si } j = 0\,; \\ \frac{q_{kj} - q_{k(j+1)}}{q_{k0}}, & \text{si } 1 \leq j \leq P-2\,; \\ \frac{q_{k(P-1)}}{q_{k0}}, & \text{si } j = P-1. \end{cases}$$

Par conséquent, nous avons :

$$\boldsymbol{\psi}_{k,t} = \sum_{j=0}^{P-1} f_{kj}\, \boldsymbol{\phi}_{l,t}$$

Soit $\boldsymbol{F}$ la matrice de taille $(N \times P)$ dont la ligne k est formée de f_{kj}, $j = 1, 2, \cdots, P$. Nous avons alors $\boldsymbol{F}\mathbb{1}_P = \mathbb{1}_P$.

La troisième étape met à jour $\boldsymbol{\psi}_{k,t}$ pour avoir $\boldsymbol{w}_{k,t}$. Suivant les mêmes arguments des stratégies APC et CPA, nous pouvons écrire la troisième étape en utilisant les mêmes coefficients a_{kl} :

$$\boldsymbol{w}_{k,t} = \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\psi}_{l,t}.$$

Enfin, nous arrivons à la version suivante de la stratégie coopérative avec lissage :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t} &= \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} q_{l0} \Big(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{w}_{k,t-1} \Big), \\
\boldsymbol{\psi}_{k,t} &= \sum_{j=0}^{P-1} f_{kj} \boldsymbol{\phi}_{k,t-j}, \\
\boldsymbol{w}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\psi}_{l,t},
\end{aligned}$$

où les coefficients non négatifs satisfont :

$$c_{kl} \geq 0, \qquad \sum_{l \in \mathcal{V}_k} c_{kl} = 1, \quad c_{kl} = 0 \text{ si } l \notin \mathcal{V}_k \tag{4.11}$$

$$a_{kl} \geq 0, \qquad \sum_{l \in \mathcal{V}_k} a_{kl} = 1, \quad a_{kl} = 0 \text{ si } l \notin \mathcal{V}_k \tag{4.12}$$

$$f_{kj} \geq 0, \qquad \sum_{j=0}^{P-1} f_{kj} = 1 \tag{4.13}$$

$$0 < q_{l0} \leq 1 \tag{4.14}$$

pour $k = 1, 2, \cdots, N$. Puisque seulement $\{q_{l0}\}$ sont utilisés, nous nous référons simplement à leur disposition par $\{q_l\}$.

La première étape de l'algorithme est une étape d'adaptation (désignée par A dans la suite), la deuxième étape est un filtrage temporel ou étape de lissage (dite étape T), et la troisième est une étape de coopération spatiale (dite étape S). Par conséquent, cette stratégie est la stratégie coopérative dite ATS.

4.3.2 Autres stratégies coopératives avec lissage

Dans la section précédente nous avons décrit la stratégie coopérative ATS. En changeant l'ordre des trois étapes, nous obtenons six différentes combinaisons de stratégies coopératives avec lissage :

1. TSA :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t-1} &= \sum_{j=0}^{P-1} f_{kj} \boldsymbol{w}_{k,t-j-1}, \\
\boldsymbol{\psi}_{k,t-1} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\phi}_{l,t-1}, \\
\boldsymbol{w}_{k,t} &= \boldsymbol{\psi}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} q_l \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{\psi}_{k,t-1} \right).
\end{aligned}$$

2. STA :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t-1} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{w}_{l,t-1}, \\
\boldsymbol{\psi}_{k,t-1} &= \sum_{j=0}^{P-1} f_{kj} \boldsymbol{\phi}_{k,t-j-1}, \\
\boldsymbol{w}_{k,t} &= \boldsymbol{\psi}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} q_l \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{\psi}_{k,t-1} \right).
\end{aligned}$$

3. SAT :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t-1} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{w}_{l,t-1}, \\
\boldsymbol{\psi}_{k,t} &= \boldsymbol{\phi}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} q_l \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{\phi}_{k,t-1} \right), \\
\boldsymbol{w}_{k,t} &= \sum_{j=0}^{P-1} f_{kj} \boldsymbol{\psi}_{k,t-j-1}.
\end{aligned}$$

4. TAS :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t-1} &= \sum_{j=0}^{P-1} f_{kj} \boldsymbol{w}_{k,t-j-1}, \\
\boldsymbol{\psi}_{k,t} &= \boldsymbol{\phi}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} q_l \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{\phi}_{k,t-1} \right), \\
\boldsymbol{w}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\psi}_{l,t}.
\end{aligned}$$

5. ATS :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t} &= \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} q_l \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{w}_{k,t-1} \right), \\
\boldsymbol{\psi}_{k,t} &= \sum_{j=0}^{P-1} f_{kj} \boldsymbol{\phi}_{k,t-j}, \\
\boldsymbol{w}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\psi}_{l,t}.
\end{aligned}$$

6. AST :

$$\begin{aligned}
\boldsymbol{\phi}_{k,t} &= \boldsymbol{w}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} q_l \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{w}_{k,t-1} \right), \\
\boldsymbol{\psi}_{k,t} &= \sum_{l \in \mathcal{V}_k} a_{kl}\, \boldsymbol{\phi}_{l,t}, \\
\boldsymbol{w}_{k,t} &= \sum_{j=0}^{P-1} f_{kj} \boldsymbol{\psi}_{k,t-j}.
\end{aligned}$$

Dans le cas particulier où $P = 1$, les stratégies TAS, ATS et AST sont réduites à la stratégie APC, alors que les stratégies TSA, STA et SAT sont réduites à la stratégie CPA. Nous pouvons diviser les six stratégies coopératives avec lissage en deux groupes en fonction de l'ordre du lissage (T) et de l'adaptation (A) :

- Premier groupe : TSA, STA, TAS.
- Second groupe : SAT, ATS, AST.

Le modèle général de ces stratégies s'écrit :

$$\left\{
\begin{array}{rl}
\boldsymbol{\phi}_{k,t-1} & = \sum_{j=0}^{P-1} f_{1,kj} \boldsymbol{w}_{k,t-j-1}, \\
\boldsymbol{\psi}_{k,t-1} & = \sum_{l \in \mathcal{V}_k} a_{1,kl}\, \boldsymbol{\phi}_{l,t-1}, \\
\boldsymbol{\beta}_{k,t-1} & = \sum_{j=0}^{P-1} f_{2,kj} \boldsymbol{\psi}_{k,t-j-1}, \\
\boldsymbol{\varpi}_{k,t} & = \boldsymbol{\beta}_{k,t-1} + \eta_{k,t} \sum_{l \in \mathcal{V}_k} c_{lk} q_l \left(\boldsymbol{y}_l z_{\boldsymbol{w}_{k,t-1},l} - z^2_{\boldsymbol{w}_{k,t-1},l} \boldsymbol{\beta}_{k,t-1} \right) \\
\boldsymbol{\varphi}_{k,t} & = \sum_{j=0}^{P-1} f_{3,kj} \boldsymbol{\varpi}_{k,t-j}, \\
\boldsymbol{\gamma}_{k,t} & = \sum_{l \in \mathcal{V}_k} a_{2,kl}\, \boldsymbol{\varphi}_{l,t} \\
\boldsymbol{w}_{k,t} & = \sum_{j=0}^{P-1} f_{4,kj} \boldsymbol{\gamma}_{k,t-j}.
\end{array}
\right.$$

La Table 4.2 montre comment les différentes stratégies se déduisent du modèle général.

4.3.3 Coefficients de combinaison dans les stratégies coopératives

Afin d'affecter un poids à l'estimation passée j, chaque nœud k utilise les coefficients de combinaison f_{kj} qui satisfont la condition de convexité (4.13). En d'autres termes, la somme des coefficients de f_{kj}, $j = 0, \cdots, P$, affectés par le nœud k à la circulation de l'information passée est égale à un. Il est intéressant de noter que l'estimation actuelle devrait avoir le plus grand poids, et le poids pour les estimations passées devrait diminuer quand j augmente. Nous considérons les règles suivantes pour définir ces coefficients :

1.
$$f_{kj} = \begin{cases} 1/2^{j+1}, & \text{pour } j = 0, \cdots, P-2\,; \\ 1/2^{P-1}, & \text{pour } j = P-1\,. \end{cases} \tag{4.15}$$

2.
$$f_{kj} = \begin{cases} 3/4, & \text{pour } j = 0\,; \\ 1/2^{j+2}, & \text{pour } j = 1, 2, \cdots, P-2\,; \\ 1/2^{P}, & \text{pour } j = P-1\,. \end{cases} \tag{4.16}$$

3.
$$f_{kj} = 1/P. \tag{4.17}$$

La Figure 4.1 présente une schématisation de ces différentes règles afin de montrer la différence entre les poids donnés aux estimations antérieures.

4.4 Résultats expérimentaux

Dans cette section, nous illustrons les performances des approches proposées, au travers de deux applications : la réduction de dimension de séries temporelles dans les RCSF, comme présenté à la section 3.5 du chapitre précédent, et une nouvelle application sur le traitement des images. Nous mesurons les performances des stratégies par l'angle obtenu pour l'estimation du $i^{\text{ème}}$ axe principal :

$$\Theta_i = \arccos\left(\frac{\boldsymbol{w}_{*,i,k}^{\top}\boldsymbol{w}_{*,i}}{\|\boldsymbol{w}_{*,i,k}\|\,\|\boldsymbol{w}_{*,i}\|}\right). \tag{4.18}$$

Table 4.2 – *Différents choix de $\boldsymbol{F}_1$, $\boldsymbol{A}_1$, $\boldsymbol{F}_2$, $\boldsymbol{F}_3$, $\boldsymbol{A}_2$ et $\boldsymbol{F}_4$ correspondant aux différentes stratégies*

$\boldsymbol{F}_1$	$\boldsymbol{A}_1$	$\boldsymbol{F}_2$	$\boldsymbol{F}_3$	$\boldsymbol{A}_2$	$\boldsymbol{F}_4$	Stratégie
$\boldsymbol{F}$	$\boldsymbol{A}$	$\mathbb{1}_N$	$\mathbb{1}_N$	$\boldsymbol{I}_N$	$\mathbb{1}_N$	TSA
$\mathbb{1}_N$	$\boldsymbol{A}$	$\boldsymbol{F}$	$\mathbb{1}_N$	$\boldsymbol{I}_N$	$\mathbb{1}_N$	STA
$\mathbb{1}_N$	$\boldsymbol{A}$	$\mathbb{1}_N$	$\boldsymbol{F}$	$\boldsymbol{I}_N$	$\mathbb{1}_N$	SAT
$\boldsymbol{F}$	$\boldsymbol{I}_N$	$\mathbb{1}_N$	$\mathbb{1}_N$	$\boldsymbol{A}$	$\mathbb{1}_N$	TAS
$\mathbb{1}_N$	$\boldsymbol{I}_N$	$\mathbb{1}_N$	$\boldsymbol{F}$	$\boldsymbol{A}$	$\mathbb{1}_N$	ATS
$\mathbb{1}_N$	$\boldsymbol{I}_N$	$\mathbb{1}_N$	$\mathbb{1}_N$	$\boldsymbol{A}$	$\boldsymbol{F}$	AST

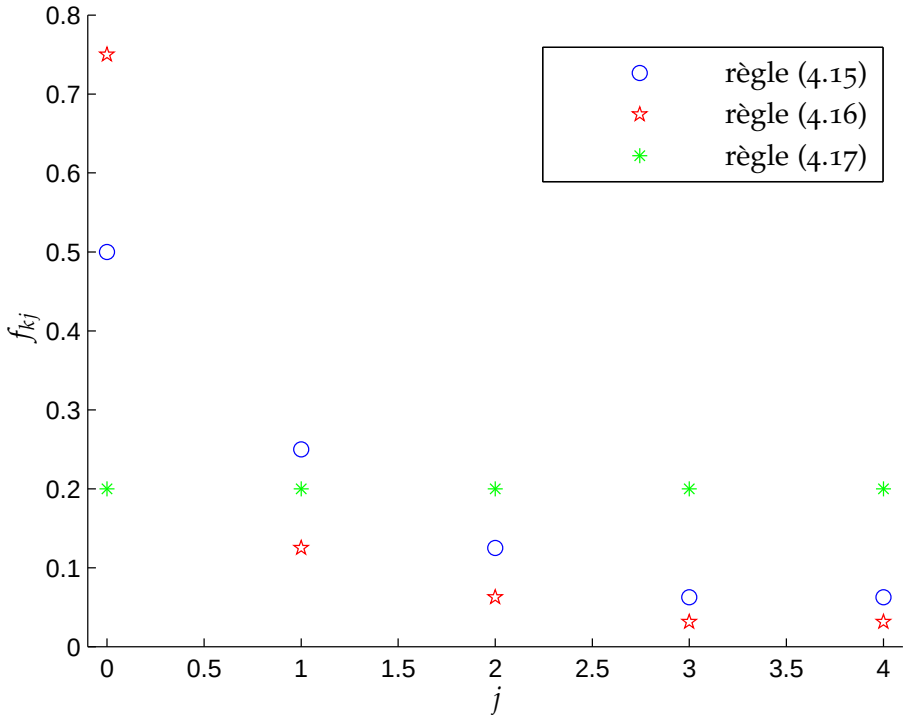

FIGURE 4.1 – *Schématisation des différentes règles de* f_{kj}.

4.4.1 Séries temporelles de mesures dans un RCSF

La même application présentée dans la section 3.5 est utilisée ici. Tout au long de nos expérimentations, nous avons constaté que la valeur de la deuxième valeur propre λ_2 est très faible. La raison de cela est qu'il y a une seule source de diffusion. Par conséquent, le deuxième axe principal est difficile à déterminer dans ce cas, ainsi que les axes principaux de plus faible rang.

Pour les stratégies gossip, nous considérons $\epsilon = 0.5$ pour trouver la moyenne non pondérée de la variable. Quant au choix du paramètre du pas η, nous utilisons une approche de validation croisée, en variant le pas selon trois séries de valeurs, la première étant $0.1, 0.01, \cdots, 0.1 \cdot 10^{-12}$, la deuxième $0.25, 0.025, \cdots, 0.25 \cdot 10^{-12}$, et la troisième $0.5, 0.05, \cdots, 0.5 \cdot 10^{-12}$. Puis, nous choisissons la valeur du pas qui mène au meilleur résultat. Nous considérons une valeur de pas $\eta = 0.0025$ Les résultats sont présentés en termes de l'angle en moyenne sur tous les nœuds. La FIGURE 4.2 et la FIGURE 4.3 illustrent la convergence des stratégies proposées dans ce chapitre. Elles montrent que toutes les stratégies proposées surpassent l'algorithme de PCADID (Ahmadi Livani et Abadi 2011) et l'algorithme de Korada et al. (2011). Ces courbes d'apprentissage montrent les avantages de la mise en œuvre du gossip. Le gossip symétrique surpasse le gossip asymétrique, comme prévu. La stratégie non coopérative présente des fluctuations, principalement en raison du système de routage et elle exige un protocole de communication synchrone. La FIGURE 4.2 et la FIGURE 4.3 montrent la stabilité des stratégies de gossip. Bien que l'utilisa-

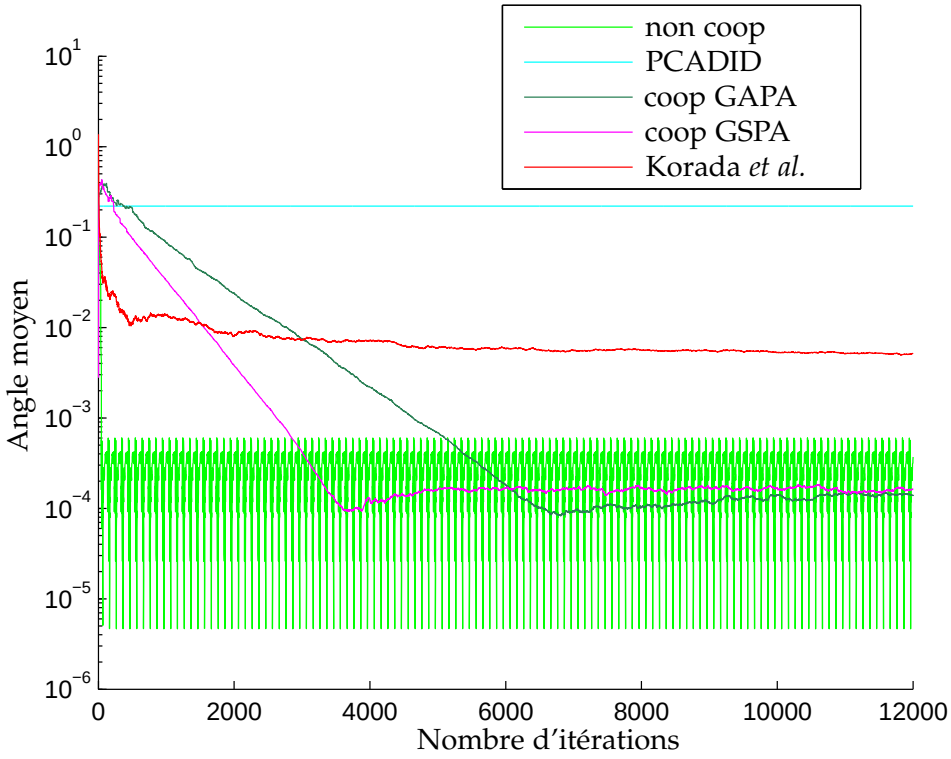

FIGURE 4.2 – *Analyse de convergence des stratégies non coopérative, GAPA, GSPA, PCADID et l'algorithme de Korada* et al. *pour l'application sur les séries temporelles.*

tion de gossip rend la convergence lente, le gossip résout le problème de la désynchronisation entre les nœuds.

Pour les stratégies de lissage, nous considérons également l'approche de validation croisée, avec les mêmes séries de valeurs utilisées pour le choix du pas de la stratégie gossip. Cette approche nous donne la valeur de pas $\eta = 0.0025$. Nous considérons $c_{kk} = 1$ et $c_{lk} = 0$ pour chaque $k \neq l$, avec $l, k = 1, 2, \cdots, 100$. Les résultats des FIGURE 4.4 jusqu'à FIGURE 4.8 sont présentés en termes de l'angle en moyenne sur tous les nœuds. Ces courbes d'apprentissage montrent que la stratégie non coopérative est dépassée par toutes les stratégies coopératives, alors que les stratégies coopératives avec lissage sont très utiles en présence de bruits. Nous distinguons deux types de bruits : le bruit sur les communications entre les nœuds, représenté par un bruit gaussien ajouté à ψ, et le bruit sur les mesures, représenté par un bruit gaussien ajouté à $\boldsymbol{y}$.

La FIGURE 4.4 et la FIGURE 4.5 montrent les performances de la stratégie ATS utilisant la règle (4.15) pour f_{kj}. Nous étudions l'influence de deux paramètres : l'écart type σ du bruit gaussien et le nombre P des dernières estimations. Le cas $P = 1$ correspond aux stratégies APC ou CPA. Nous utilisons les valeurs suivantes pour l'écart type du bruit : $\sigma = 0,01$ qui correspond à 10% de la plus petite composante du premier axe principal pour le bruit de la communication, ainsi que $\sigma = 0,05$. Comme prévu, pour un même niveau de bruit, l'algorithm ATS donne un angle moyen après convergence inférieur lorsque P augmente, mais plus lent car il implique le lissage des estimations précédentes. Nous notons que les courbes

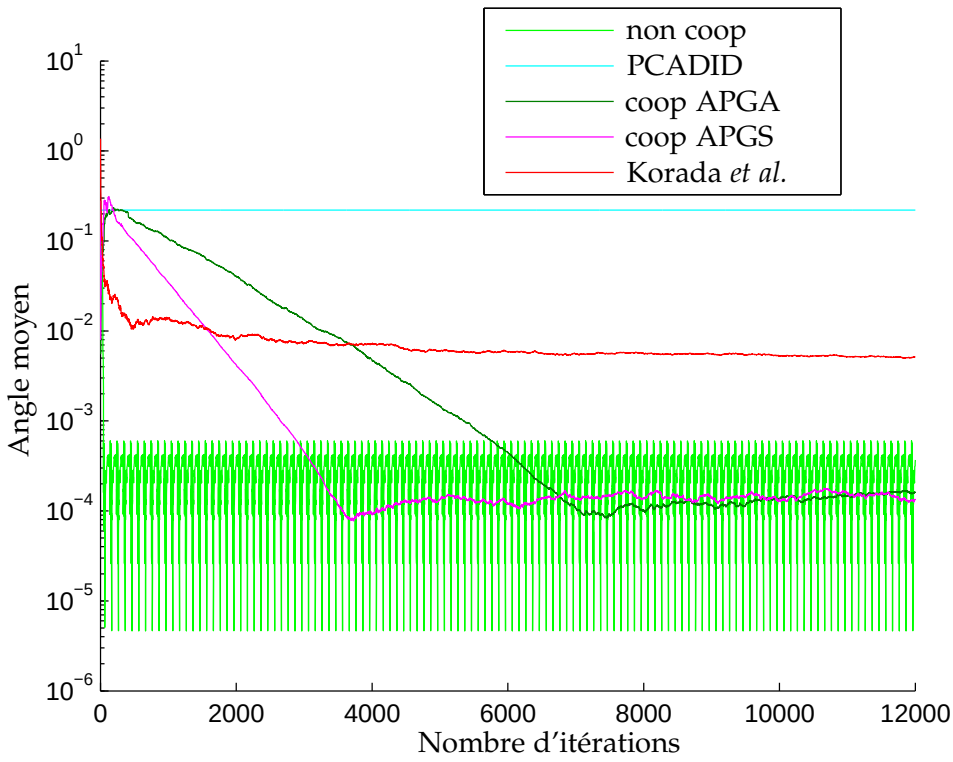

FIGURE 4.3 – *Analyse de convergence des stratégies non coopérative, APGA, APGS, PCADID et l'algorithme de Korada* et al. *pour l'application sur les séries temporelles.*

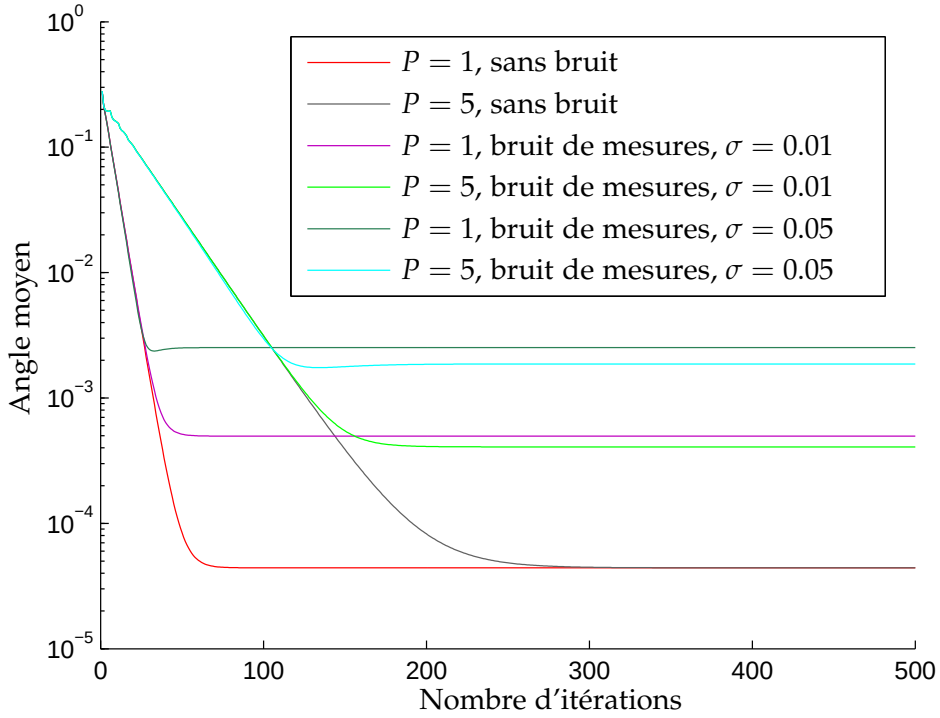

FIGURE 4.4 – *Analyse de convergence de la stratégie de lissage ATS pour l'application sur les séries temporelles, avec un bruit additif sur les mesures.*

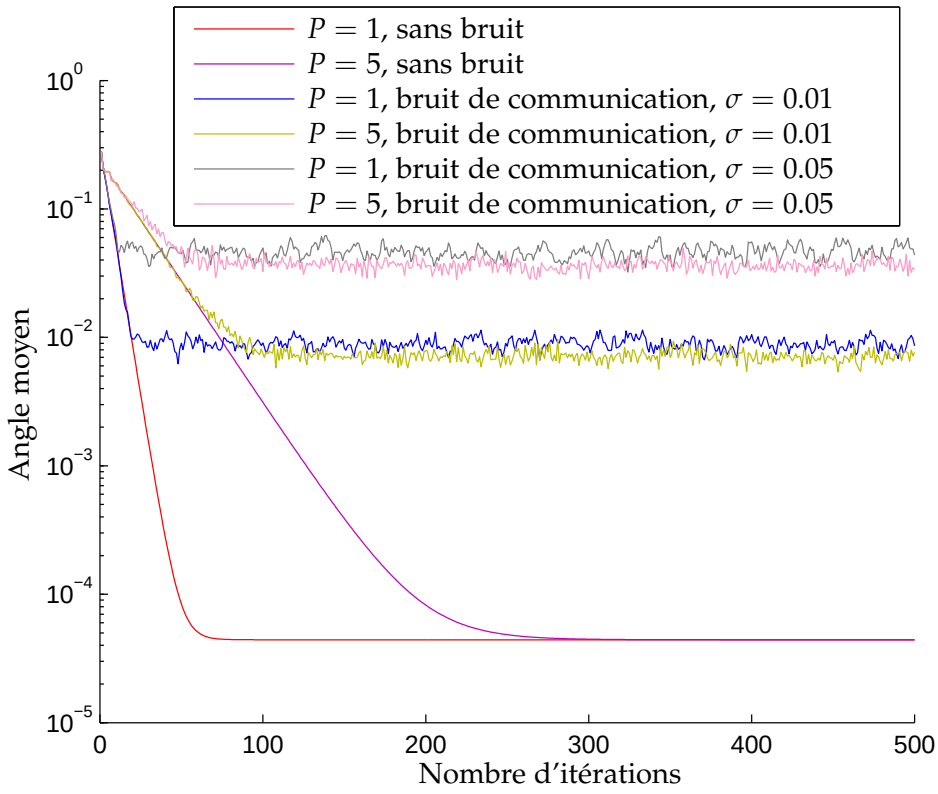

FIGURE 4.5 – *Analyse de convergence de la stratégie de lissage ATS pour l'application sur les séries temporelles avec bruit additif sur les communications.*

restent lisses lorsque les mesures sont bruitées, contrairement au cas de bruit de communication parce que dans le premier, le bruit est ajouté une fois à l'étape de mesure avant le traitement, tandis que dans le second, le bruit est ajouté après chaque itération.

La FIGURE 4.6 montre les performances de la stratégie ATS utilisant les règles (4.15), (4.16) et (4.17) pour f_{kj}. Les deux règles(4.15), (4.16) ont essentiellement les mêmes performances, avec un léger avantage pour la règle (4.15) et elles surpassent la règle (4.17). La FIGURE 4.7 montre les performances du premier groupe (STA, TSA et TAS) et la FIGURE 4.8 montre les performances du deuxième groupe (ATS, AST et SAT). Nous remarquons que, dans chaque groupe, l'ordre de A et S détermine les performances. Il est ainsi préférable que la stratégie effectue l'adaptation (A) avant la coopération spatiale (S).

4.4.2 Application sur le traitement des images

La seconde application concerne l'analyse des données à grande échelle, c'est-à-dire données massives ou *big data*, telles que les documents texte et les images. Le traitement des images est étudié dans de nombreux domaines, dont la reconnaissance de l'écriture manuscrite, la segmentation et le suivi en microscopie de cellules vivantes, la construction et la correction des cartes à partir d'images satellites ou d'images aériennes, la surveillance et l'évaluation de la production agricole, le contrôle du niveau de maturité des fruits sur une ligne d'emballage, pour n'en citer

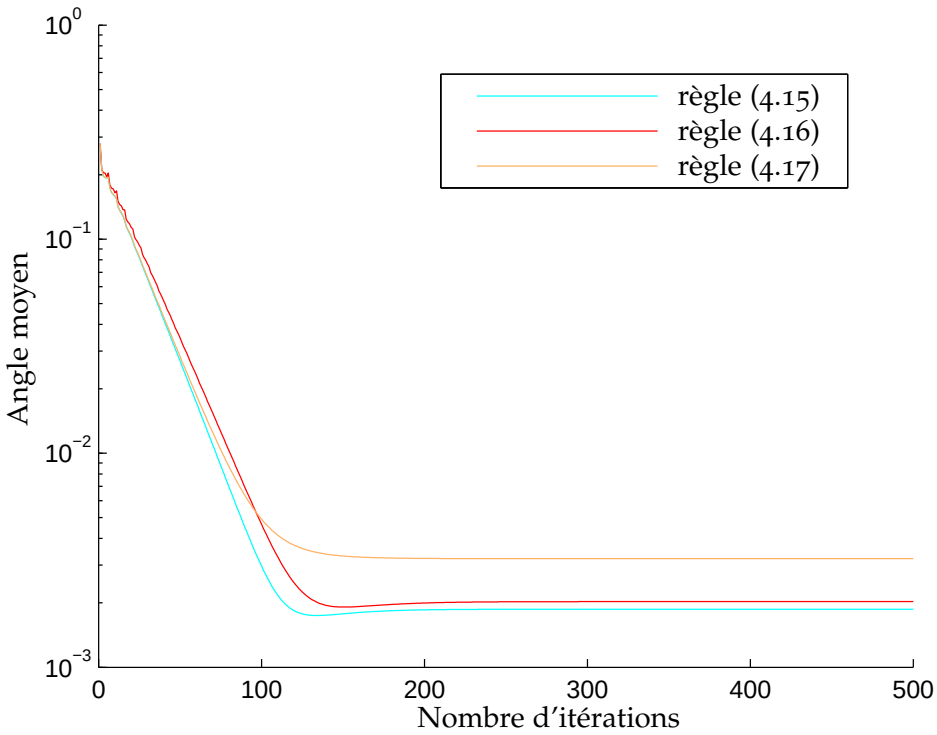

FIGURE 4.6 – *Analyse de convergence de la stratégie de lissage ATS, pour $P = 5$ et avec un bruit de mesures ($\sigma = 0.05$), utilisant les règles* (4.15), (4.16) *et* (4.17) *pour f_{kj}.*

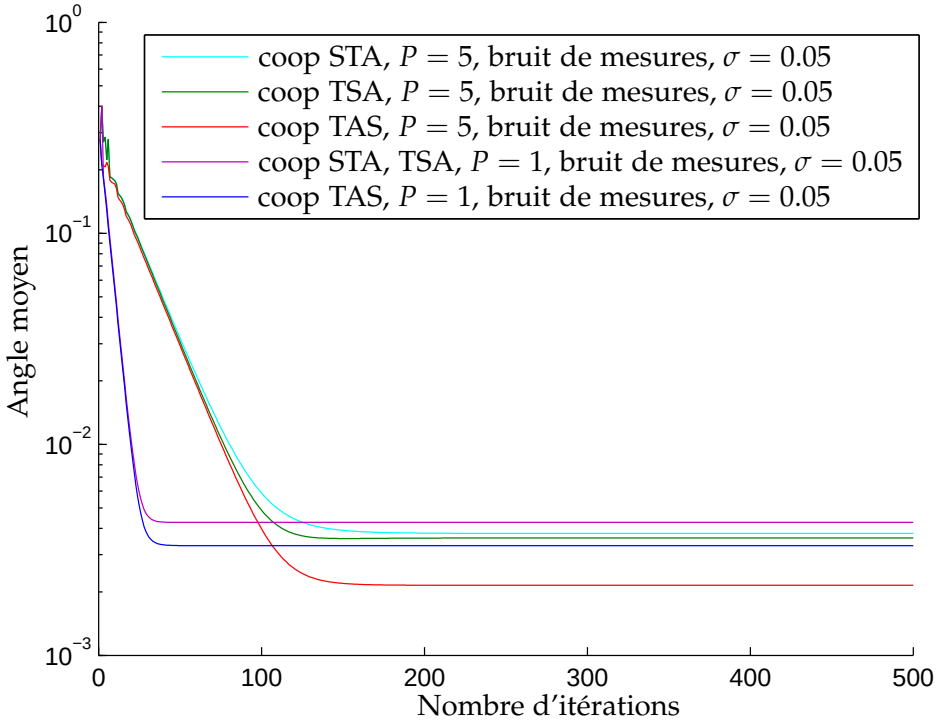

FIGURE 4.7 – *Analyse de convergence des stratégies de lissage STA, TSA et TAS dans l'application sur les séries temporelles avec bruit de mesures.*

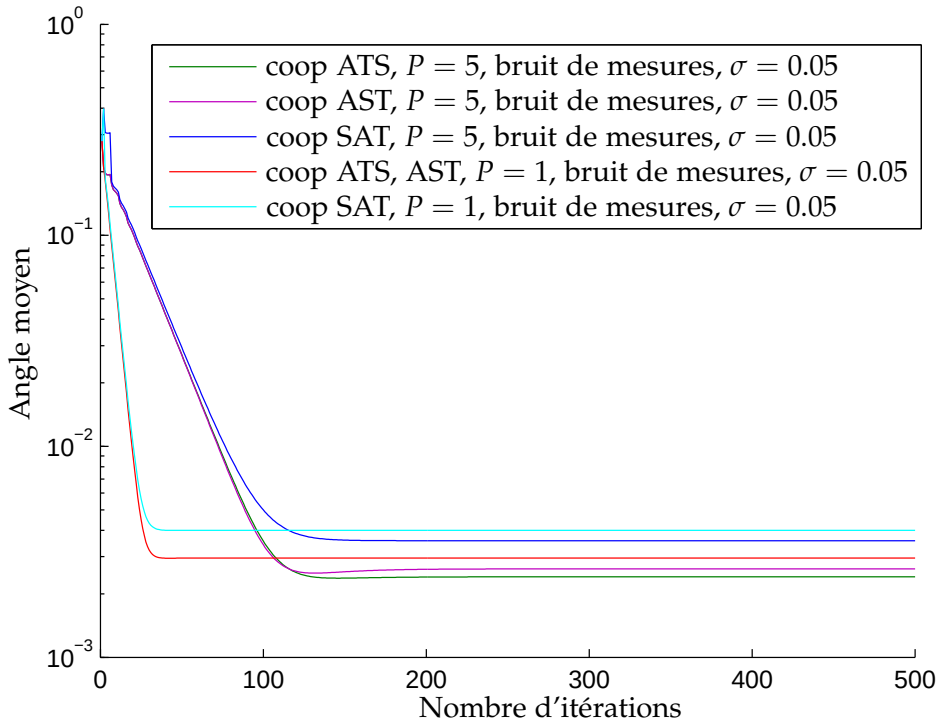

FIGURE 4.8 – *Analyse de convergence des stratégies de lissage ATS, AST, et SAT dans l'application sur les séries temporelles avec bruit de mesures.*

que quelques exemples. Toutes ces applications disposent d'énormes ensembles d'images, souvent de grande taille, afin d'en déduire un modèle pour le débruitage, l'extraction de caractéristiques, la reconnaissance de formes ou la classification/détection. Les techniques de l'ACP ont été appliquées avec succès sur des ensembles de données à petite échelle, tandis que les difficultés de calcul émergent lorsqu'il s'agit de grands ensembles de données tels qu'un grand nombre d'images de grande taille. Nous allons montrer que le cadre proposé surmonte ces obstacles et fournit un moyen naturel de traiter des ensembles de données à grande échelle.

Nous considérons un ensemble d'images manuscrites du chiffre manuscrit "1" de 28-par-28 pixels, traitées en tant que vecteurs de dimension 784. Afin de comparer avec la solution obtenue à partir de la décomposition en éléments propres de la matrice de covariance d'échantillon, le nombre d'images est limité à 90. Cette fois-ci, la connexion entre les nœuds est établi pour $\delta = 2250$ qui représente à peu près 30% de la distance maximale séparante les données. Pour les paramètres de pas, l'approche de validation croisée, avec les trois séries de valeurs, $0.1, 0.01, \cdots, 0.1 \cdot 10^{-12}$, $0.25, 0.025, \cdots, 0.25 \cdot 10^{-12}$ et $0.5, 0.05, \cdots, 0.5 \cdot 10^{-12}$, nous permet de prendre $\eta_1 = 5 \cdot 10^{-8}$ pour le premier axe principal et un autre, $\eta_2 = 1 \cdot 10^{-7}$ pour le deuxième axe principal, afin d'assurer une meilleure convergence. Les mêmes valeurs de pas sont prises pour les stratégies coopératives et non coopératives. Pour l'algorithme TI, nous prenons $\eta_1 = \eta_2 = 0.05$, pour l'algorithme QR $\eta_1 = \eta_2 = 5 \cdot 10^{-6}$, pour l'algorithme OJAn $\eta_1 = \eta_2 = 5 \cdot 10^{-8}$, et pour l'algorithme LUO $\eta_1 = 5 \cdot 10^{-11}$ et

$\eta_2 = 2.5 \cdot 10^{-10}$, dans les stratégies coopératives et non coopératives, ainsi que la stratégie de consensus.

La FIGURE 4.9 montre la convergence des différents processus d'orthogonalisation en utilisant le critère de Oja. Elle montre que l'orthogonalisation de Gram-Schmidt et l'orthogonalisation par déflation ont une efficacité comparable, en particulier avec les stratégies coopératives. L'orthogonalisation de Gram-Schmidt est légèrement plus performante que l'orthogonalisation par déflation. Pour cette raison, nous allons utiliser le premier dans la suite.

Les FIGURE 4.10 jusqu'à FIGURE 4.12 montrent la convergence de chaque stratégie pour les différents critères proposés dans le présent document, avec les angles en moyenne sur les N nœuds. Encore une fois, la stratégie non coopérative est nettement dépassée par toutes les stratégies coopératives. Compte tenu de la vitesse de convergence, les figures montrent que les algorithmes OJAn, Oja, et TI ont la convergence la plus rapide, suivi consécutivement par les algorithmes QR et LUO. En termes de convergence, l'algorithme TI a le plus grand angle moyen après convergence, alors que tous les autres algorithmes ont un angle moyen comparable durant le régime permanent. Comme dans l'application sur les séries temporelles donnée dans la section 3.5, le couplage de l'algorithme Oja (ou OJAn) avec la stratégie coopérative adapter-puis-combiner présente le meilleur compromis entre l'angle moyen après convergence et la vitesse de convergence.

Pour les stratégies gossip, nous considérons les pas $\eta_1 = 5 \cdot 10^{-9}$ pour le premier axe principal et $\eta_2 = 1 \cdot 10^{-8}$ pour le second axe principal. Notons que $\eta_2 \geq \eta_1$ parce que l'estimation du second axe principal $\boldsymbol{w}_{2,t}$ utilise l'estimation du premier axe principal $\boldsymbol{w}_{1,t-1}$. La FIGURE 4.13 et la FIGURE 4.14 montrent les résultats de la mise en œuvre de gossip. Elles montrent que les stratégies proposées sont plus stables que la stratégie non coopérative. Bien que l'utilisation de gossip rend la convergence plus lente, elle résout le problème de la désynchronisation et de la communication entre les nœuds.

Conclusion du chapitre

Dans ce chapitre, nous avons étendu le cadre de l'ACP en réseau pour estimer plusieurs axes principaux en opérant des processus d'orthogonalisation, comme l'orthogonalisation de Gram-Schmidt et l'orthogonalisation par déflation. De plus, nous avons résolu le problème de désynchronisation entre les nœuds d'un réseau en implémentant une approche de type gossip pour l'ACP. Nous avons également étudié l'effet du bruit de communication ou de mesure, en proposant des stratégies coopératives par lissage pour y remédier. Des expériences ont été menées sur deux applications différentes, montrant la pertinence des travaux réalisés dans ce chapitre.

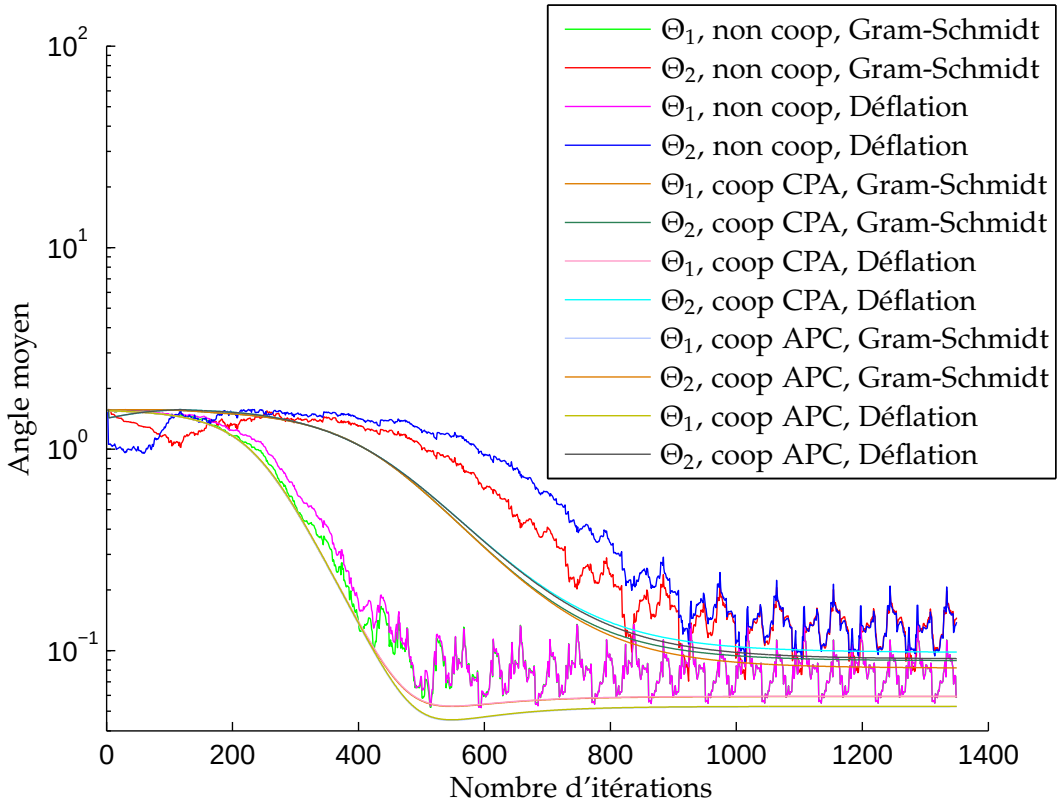

FIGURE 4.9 – *Analyse de la convergence dans l'application de traitement des images, pour différents processus d'orthogonalisation avec le critère de Oja.*

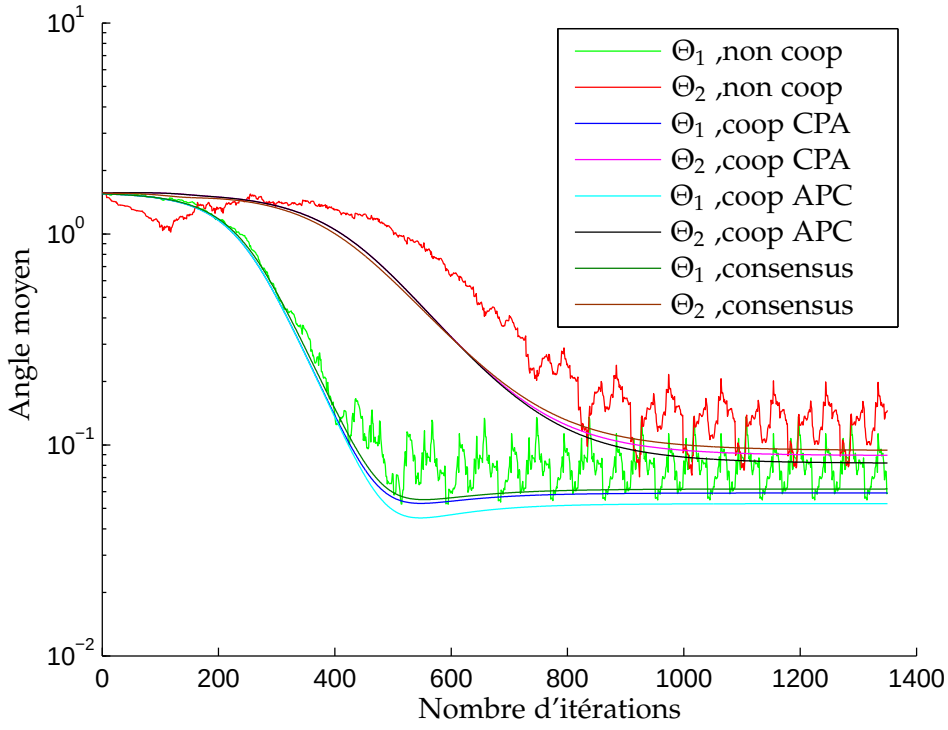

FIGURE 4.10 – *Analyse de la convergence dans l'application de traitement des images, pour différentes stratégies de critère de Oja avec l'orthogonalisation de Gram-Schmidt.*

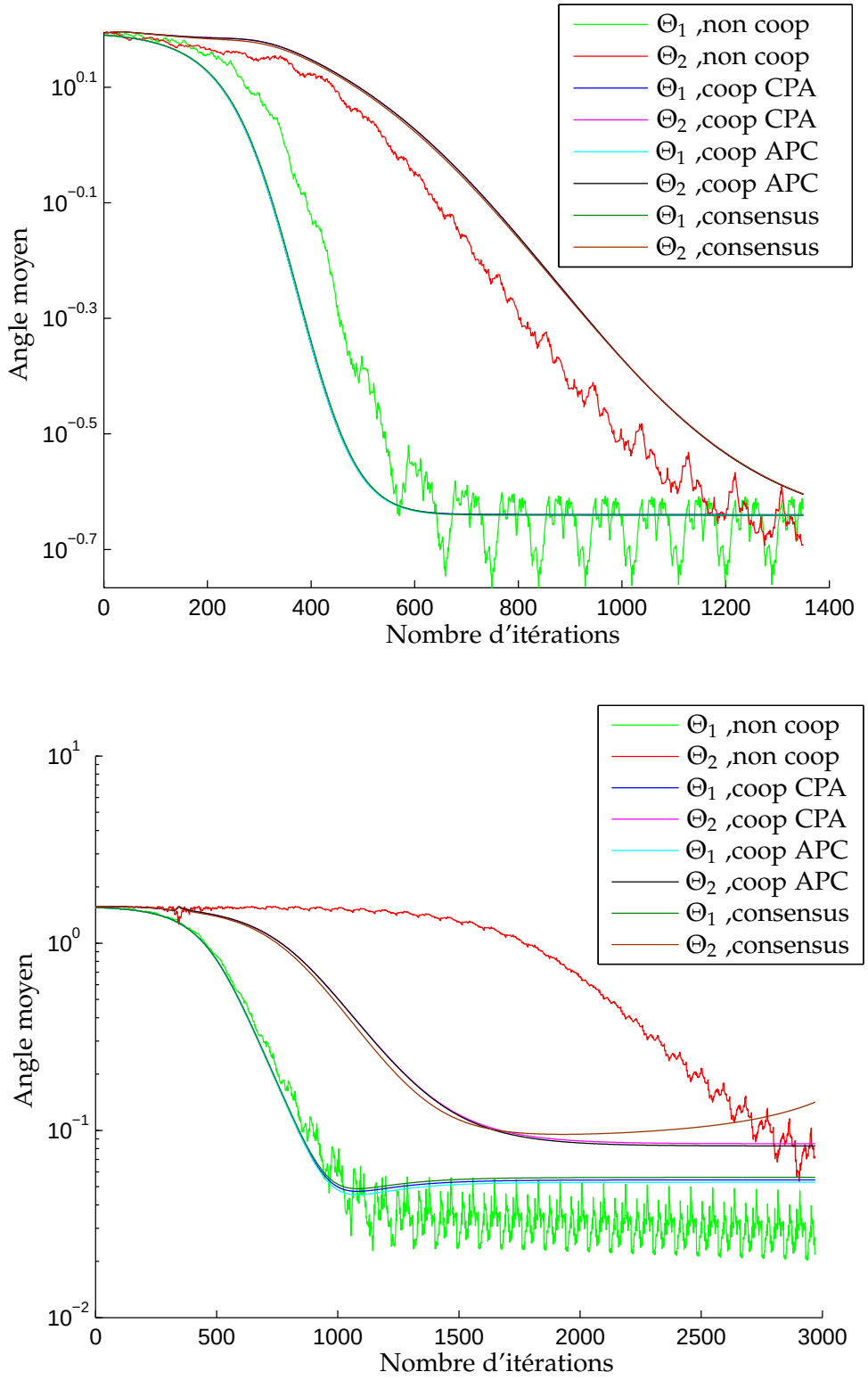

FIGURE 4.11 – *Analyse de la convergence dans l'application de traitement des images, pour différentes stratégies des algorithmes de la théorie de l'information (figure du haut) et du quotient de Rayleigh (figure du bas).*

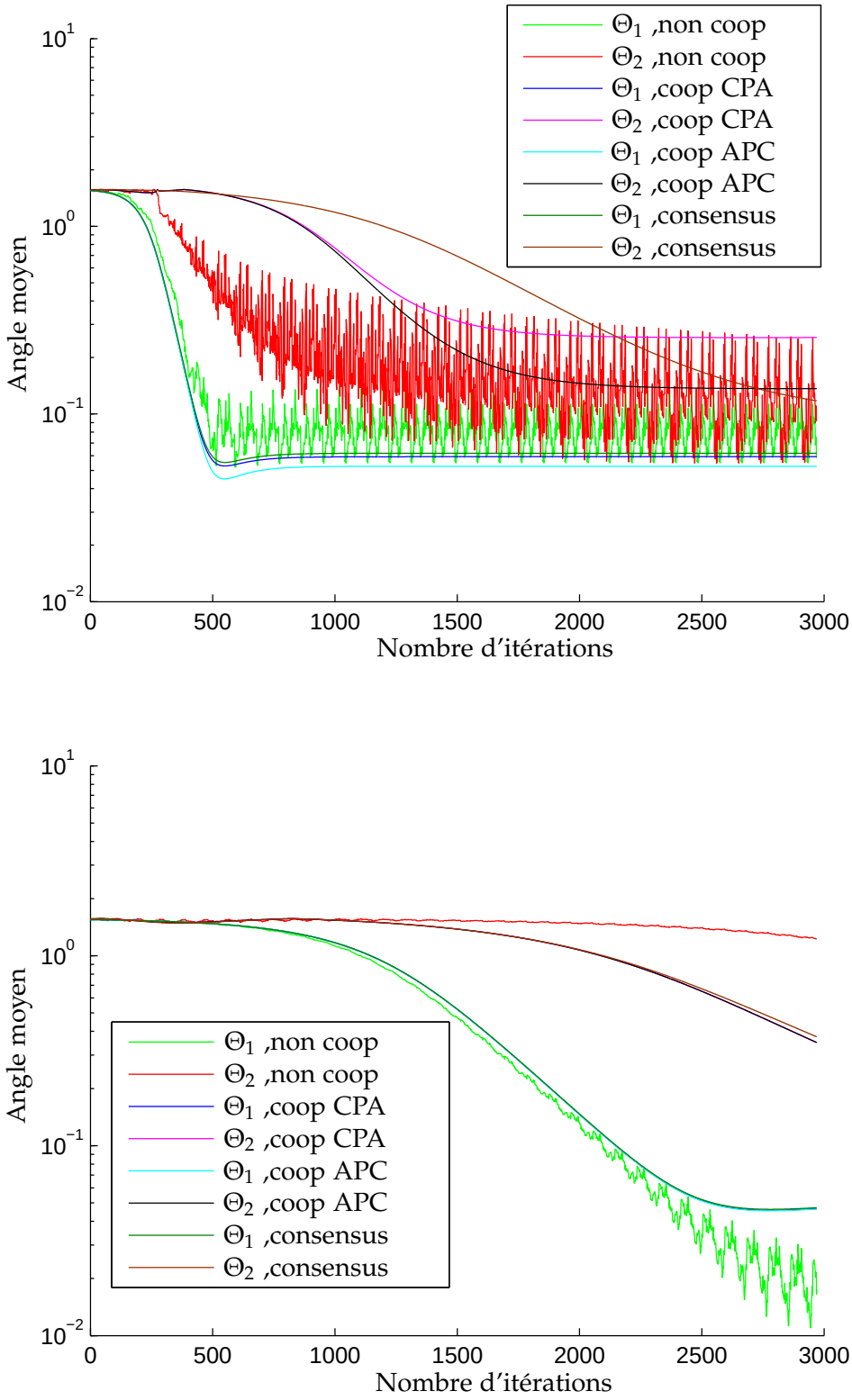

FIGURE 4.12 – *Analyse de la convergence dans l'application de traitement des images, pour différentes stratégies des algorithmes de OJAn (figure du haut) et de LUO (figure du bas).*

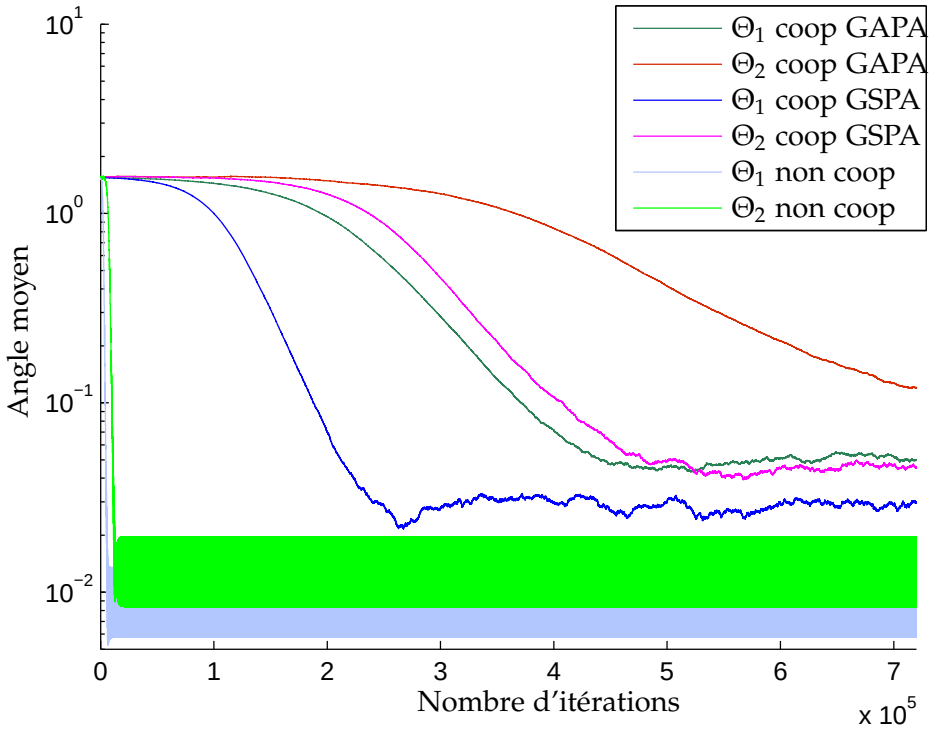

FIGURE 4.13 – *Analyse de convergence des stratégies non coopératives, GAPA et GSPA dans l'application de traitement des images.*

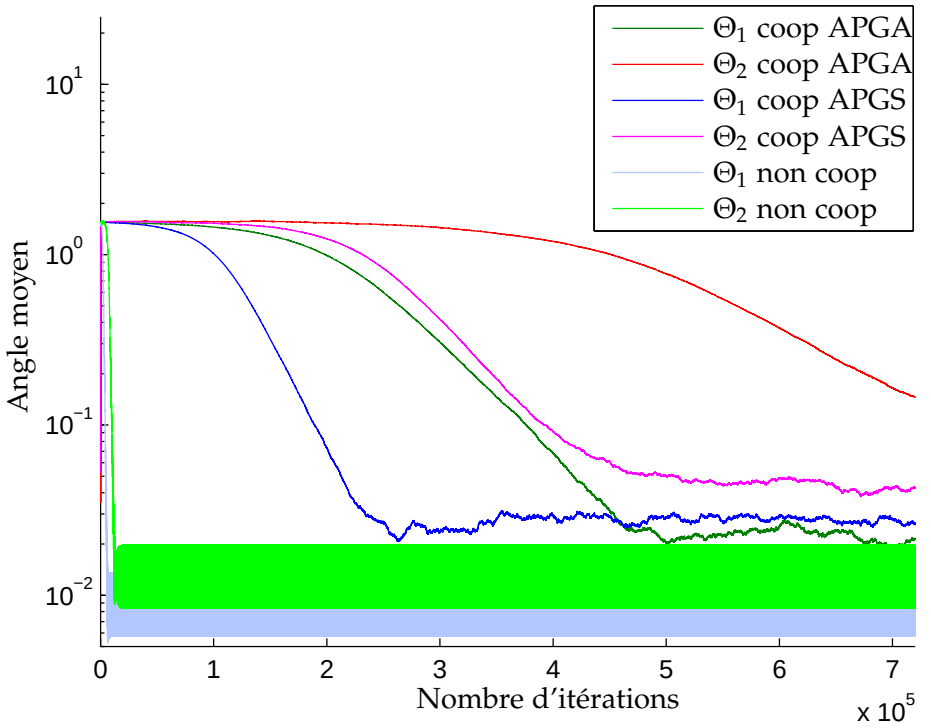

FIGURE 4.14 – *Analyse de convergence des stratégies non coopératives, APGA et APGS dans l'application de traitement des images.*

Conclusion générale

Sommaire

Dans ce mémoire de thèse, nous avons fait face à plusieurs problèmes qui limitent le fonctionnement et le développement des réseaux de capteurs sans fil. Nous avons traité le problème de surveillance d'un phénomène physique. Tout d'abord, nous avons modélisé un champ diffusé. Afin d'améliorer l'estimation, nous avons profité de la mobilité des capteurs. Les capteurs se sont déplacés pour assurer une meilleure couverture de la région étudiée. Ensuite, nous avons traité le problème de la grande masse de données acquises par un RCSF. En effet, la grande dimension des données limite le fonctionnement du réseau de capteurs à cause des contraintes d'énergie et de calcul. Nous avons proposé des méthodes de réduction de la dimensionnalité des données en réexaminant l'analyse en composantes principales dans les cadre des réseaux.

Cette conclusion commence par un résumé des contributions majeures de cette thèse. Nous proposons par la suite quelques perspectives futures de recherche.

Résumé des contributions

Le travail présenté dans les chapitres précédents, est concentré sur deux principales applications dans les réseaux de capteurs sans fil. La première est la modélisation d'un champ diffusé par un réseau de capteurs mobiles, alors que la seconde consiste à réduire la dimensionnalité des données. Les principales contributions de cette thèse peuvent être résumées comme suit.

Dans la première partie de ce document, nous avons introduit au chapitre 1 un cadre général pour bénéficier des plus récents développements des méthodes à noyaux en apprentissage statistique. Nous avons présenté la théorie des noyaux reproduisants tout en décrivant leurs caractéristiques, et en précisant les deux éléments fondamentaux des méthodes à noyaux à savoir l'astuce du noyau et le théorème de représentation. Nous avons également présenté les méthodes à noyaux avec l'ACP à noyaux et la régression par moindres carrés à noyaux. Dans le chapitre 2, nous avons utilisé cette dernière dans le but de modéliser la distribution spatiale d'un champ de diffusion. Ce champ est estimé, soit par un modèle global calculé par un centre de fusion, soit par un modèle local construit par chaque capteur en utilisant les informations collectées à partir de ses voisins. Nous avons examiné à cet effet l'impact de la taille du voisinage

sur l'erreur de validation et nous avons estimé la taille optimale du voisinage qui minimise cette erreur. Afin de diminuer l'erreur d'estimation et ainsi améliorer la modélisation, nous avons profité de la mobilité des capteurs. Nous avons proposé trois scénarios de mobilité qui déplacent un seul capteur à chaque itération vers la région la moins couverte selon un schéma d'optimisation. Le premier scénario construit un modèle local pour chaque capteur. Tous les capteurs sont capables de se déplacer dans le but de réduire l'erreur de validation. Le deuxième scénario a le même principe que le premier mais avec un modèle global pour tous les capteurs. Dans le dernier scénario, une partie des capteurs est mobile, et cherche les informations pour adapter l'ensemble d'apprentissage formé initialement par les capteurs fixes. Les schémas d'optimisation utilisés sont basés sur les dérivées du premier et du second ordre de l'erreur de régression, en utilisant trois schémas qui sont : la méthode du point fixe, l'algorithme du gradient et la méthode de Newton. Enfin, nous avons montré la pertinence des méthodes proposées en termes d'erreur de validation, par rapport au cas statique. En étudiant ces différents scénarios, nous avons montré que la méthode de Newton surpasse les deux premiers en terme d'erreur réduite et de faible distance parcourue, mais au prix d'une plus grande complexité de calcul.

La deuxième partie de cette thèse a traité le problème de réduction de dimensionnalité des données acquises par les capteurs, en utilisant l'analyse en composantes principales. Dans le chapitre 3, nous avons estimé l'axe principal le plus pertinent, sans calcul de la matrice de covariance des données. Nous avons proposé, selon la topologie du réseau, des stratégies non coopératives et coopératives adaptées aux réseaux de capteurs sans fil, en minimisant à la volée une fonction coût appropriée. Dans la stratégie coopérative, les capteurs coopèrent ensemble par diffusion de l'information en vue d'estimer les axes principaux. Nous avons proposé deux stratégies coopératives, les stratégies combiner-puis-adapter et adapter-puis-combiner. Nous avons fourni également des connexions avec l'apprentissage par consensus. Une étude de convergence des méthodes proposées a été détaillée. Les résultats ont montré la pertinence du cadre proposé sur cette application. Nous avons étendu notre étude sur la mise en œuvre de l'ACP dans le chapitre 4 pour l'estimation de plusieurs axes principaux en opérant l'orthogonalisation de Gram-Schmidt et l'orthogonalisation par déflation. Ensuite, nous avons implémenté une approche de type gossip pour l'ACP dans le but de résoudre le problème de désynchronisation entre les nœuds d'un réseau. De plus, nous avons proposé des stratégies coopératives par lissage temporel pour remédier aux effets du bruit de communication et de mesure. Des expériences ont été menées sur deux applications différentes, montrant la pertinence des travaux réalisés.

Perspectives

Cette thèse a traité deux problèmes dans les réseaux de capteurs sans fil. Elle a fourni des solutions originales, d'une part pour améliorer la modélisation du champ de diffusion par la mobilité des capteurs, et d'autre part pour réduire la dimensionnalité des meures des capteurs par l'ACP

en réseaux. Dans la continuité de notre travail de thèse, nous tenons à étudier les aspects suivants afin d'améliorer les méthodes proposées.

- Prendre en considération la consommation d'énergie des capteurs : la consommation d'énergie est l'une des plus grandes contraintes des RCSF. Nous pouvons utiliser un modèle d'énergie pour chaque capteur. Ce modèle est mis à jour à chaque intervalle de temps. Les algorithmes de mobilité alors doivent prendre en considération ce modèle. Un capteur de faible énergie limite ses déplacements et passe son tour à celui qui a une plus grande énergie. De plus, un capteur dont sa source d'énergie est "sur le point de mourir", doit trouver une solution pour communiquer ses informations afin de ne pas les perdre.
- Prendre en considération la présence des obstacles : en général, les capteurs ne sont pas déployés dans un espace vide. Un capteur mobile peut rencontrer des obstacles physiques durant son mouvement. Ces derniers peuvent stopper le déplacement du capteur ou même causer sa destruction. Pour cela, il faut trouver des solutions pour estimer les positions de ces obstacles et les contourner.
- Prendre en considération le changement temporel du champ diffusé : Notre travail dans la première partie traite un champ de diffusion dans l'espace considéré stationnaire, c'est-à-dire invariant par rapport au temps. Nous pouvons étendre cette étude à un champ de diffusion variable en fonction du temps en prenant en considération le changement de la mesure entre différentes itérations.
- Détecter et isoler un capteur défectueux : un capteur malsain peut fournir des mesures imprécises ou erronées, ce qui induit une diffusion des erreurs de proche en proche dans les modèles résultants. Il est alors indispensable de détecter le capteur défaillant et de l'isoler afin d'empêcher la propagation des erreurs. Pour cela, différentes approches peuvent être considérées, comme par exemple en confrontant deux modèles, un avec et un sans les mesures issues du capteur étudié.
- Considérer le problème de séparation des sources : un champ de diffusion peut résulter de plusieurs sources. Une des méthodes connues pour la séparation des sources est l'analyse des composantes indépendantes. Un futur travail consiste à proposer des stratégies non coopératives et coopératives, afin de résoudre ce problème dans les réseaux.

Annexes

A

Sommaire

A.1 Norme de bloc maximale

Soit $\boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N]^\top$ un vecteur de N blocs dont les éléments individuels sont chacun de dimension $(p \times 1)$. La norme de bloc maximale de $\boldsymbol{x}$ est définie par Bertsekas et Tsitsiklis (1997), Takahashi et Yamada (2008) comme suit :

$$\|\boldsymbol{x}\|_{b,\infty} = \max_{1 \le k \le N} \|\boldsymbol{x}\|,$$

où $\|\cdot\|$ désigne la norme Euclidienne de son argument. En conséquence, la norme de bloc maximale d'une matrice bloc $\boldsymbol{\mathcal{A}}$ de dimension $(N \times N)$, dont les éléments individuels sont chacun de taille $(p \times p)$, est définie comme suit :

$$\|\boldsymbol{\mathcal{A}}\|_{b,\infty} = \max_{x \neq 0} \frac{\|dA\|_{b,\infty}}{\|\boldsymbol{x}\|_{b,\infty}},$$

Lemme A.1 (Invariance unitaire) *Soit $\boldsymbol{\mathcal{U}} = diag\{\boldsymbol{U}_1, \boldsymbol{U}_2, \cdots, \boldsymbol{U}_N\}$ la matrice bloc diagonale de dimension $(N \times N)$ avec $\boldsymbol{U}_k$ des matrices unitaires de dimension $(p \times p)$. On a alors les propriétés suivantes :*

1. $\|\boldsymbol{\mathcal{U}}\boldsymbol{x}\|_{b,\infty} = \|\boldsymbol{x}\|_{b,\infty}$
2. $\|\boldsymbol{\mathcal{U}}\boldsymbol{\mathcal{A}}\boldsymbol{\mathcal{U}}^\top\|_{b,\infty} = \|\boldsymbol{\mathcal{A}}\|_{b,\infty}$

pour tout vecteur bloc $\boldsymbol{x}$ et matrice $\boldsymbol{\mathcal{A}}$ de dimensions appropriées.

Lemme A.2 (Bornes) *Soit $\boldsymbol{\mathcal{A}}$ une matrice bloc de dimension $(N \times N)$ avec les matrices blocs $\boldsymbol{\mathcal{A}}_{lk}$ chacune de dimension $(p \times p)$. On a alors les résultats suivants :*

1. *La norme de $\boldsymbol{\mathcal{A}}$ et de sa transposée sont reliées par :*

$$\|\boldsymbol{\mathcal{A}}^\top\|_{b,\infty} \le N\|\boldsymbol{\mathcal{A}}\|_{b,\infty}$$

2. *La norme de $\boldsymbol{\mathcal{A}}$ est bornée par :*

$$max_{1 \le l,k \le N}\|\boldsymbol{A}_{lk}\| \le \|\boldsymbol{\mathcal{A}}\|_{b,\infty} \le N\Big(max_{1 \le l,k \le N}\|\boldsymbol{A}_{lk}\|\Big)$$

3. *Si $\boldsymbol{\mathcal{A}}$ est hermitienne et définie non négative, alors il existe des constantes positives finies c_1 et c_2 telles que :*

$$c_1 Tr(\boldsymbol{\mathcal{A}}) \le \|\boldsymbol{\mathcal{A}}\|_{b,\infty} \le c_2 Tr(\boldsymbol{\mathcal{A}})$$

Lemme A.3 (Matrice stochastique droite ou gauche) *Soient $\boldsymbol{C}$ une matrice stochastique droite de dimension $(N \times N)$, c'est-à-dire $\boldsymbol{C}\mathbb{1} = \mathbb{1}$ et $\boldsymbol{A}$ une matrice stochastique gauche de dimension $(N \times N)$, c'est-à-dire $\boldsymbol{A}^\top\mathbb{1} = \mathbb{1}$. Soient les matrices bloc*

$$\boldsymbol{\mathcal{A}}^\top = \boldsymbol{A}^\top \otimes \boldsymbol{I}_p, \qquad \boldsymbol{\mathcal{C}} = \boldsymbol{C} \otimes \boldsymbol{I}_p,$$

formées de blocs chacun de dimension $(p \times p)$. On a alors :

$$\|\boldsymbol{\mathcal{A}}^\top\|_{b,\infty} = 1, \qquad \|\boldsymbol{\mathcal{C}}\|_{b,\infty} = 1$$

Lemme A.4 (Matrices bloc diagonales hermitiennes) *Soit la matrice bloc diagonale hermitienne de dimension $(N \times N)$, $\boldsymbol{\mathcal{D}} = diag\{\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_N\}$, où chaque $\boldsymbol{D}_k$ est une matrice hermitienne de dimension $(p \times p)$. On a alors :*

$$\rho(\boldsymbol{\mathcal{D}}) = \max_{1 \le k \le N} \rho(\boldsymbol{D}_k) = \|\boldsymbol{\mathcal{D}}\|_{b,\infty}$$

où $\rho(\cdot)$ désigne le rayon spectral (la plus grande valeur propre) de son argument. Autrement dit, le rayon spectral de $\boldsymbol{\mathcal{D}}$ est égale à sa norme de bloc maximale, qui à son tour est égale au plus grand rayon spectral de ses éléments de bloc.

Lemme A.5 (Matrices bloc diagonales transformées par des matrices stochastiques gauches) *Soit la matrice bloc diagonale hermitienne, $\boldsymbol{\mathcal{D}} = diag\{\boldsymbol{D}_1, \boldsymbol{D}_2, \cdots, \boldsymbol{D}_N\}$, où chaque $\boldsymbol{D}_k$ de dimension $(N \times N)$ est une matrice hermitienne de dimension $(p \times p)$. Soient $\boldsymbol{A}_1$ et $\boldsymbol{A}_2$ des matrices stochastiques gauches de dimension $(N \times N)$, c'est-à-dire $\boldsymbol{A}_1^\top \mathbb{1} = \mathbb{1}$ et $\boldsymbol{A}_2^\top \mathbb{1} = \mathbb{1}$. Soient les matrices bloc :*

$$\boldsymbol{\mathcal{A}}_1^\top = \boldsymbol{A}_1^\top \otimes \boldsymbol{I}_p, \qquad \boldsymbol{\mathcal{A}}_2^\top = \boldsymbol{A}_2^\top \otimes \boldsymbol{I}_p.$$

Les matrices $\boldsymbol{\mathcal{A}}_1$ et $\boldsymbol{\mathcal{A}}_2$ sont formées de blocs chacun de dimension $(p \times p)$. On a alors :

$$\rho(\boldsymbol{\mathcal{A}}_2^\top \boldsymbol{\mathcal{D}} \boldsymbol{\mathcal{A}}_1^\top) \leq \rho(\boldsymbol{\mathcal{D}})$$

Corollaire A.1 (Propriétés de stabilité) *On considère la même énoncé du Lemme A.5, on déduit que :*

1. *La matrice $\boldsymbol{\mathcal{A}}_2^\top \boldsymbol{\mathcal{D}} \boldsymbol{\mathcal{A}}_1^\top$ est stable si $\boldsymbol{\mathcal{D}}$ est stable.*
2. *La matrice $\boldsymbol{\mathcal{A}}_2^\top \boldsymbol{\mathcal{D}} \boldsymbol{\mathcal{A}}_1^\top$ est stable pour toutes matrices stochastiques gauches $\boldsymbol{\mathcal{A}}_1$ et $\boldsymbol{\mathcal{A}}_2$ si, et seulement si, $\boldsymbol{\mathcal{D}}$ est stable.*

Bibliographie

H. Abdi et L. J. Williams. Principal component analysis. *Wiley Interdisciplinary Reviews : Computational Statistics*, 2(4) :433–459, 2010. ISSN 1939-0068. (Cité pages 10 et 61.)

M. Ahmadi Livani et M. Abadi. A PCA-based distributed approach for intrusion detection in wireless sensor networks. Dans *Computer Networks and Distributed Systems (CNDS), 2011 International Symposium on*, pages 55–60, Feb 2011. (Cité pages 62, 66 et 108.)

I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, et E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40 :102–114, 2002a. (Cité page 2.)

I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, et E. Cayirci. Wireless sensor networks : a survey. *Computer Networks*, 38 :393–422, 2002b. (Cité page 3.)

J. N. Al-Karaki, R. Ul-Mustafa, et A. E. Kamal. Data aggregation and routing in wireless sensor networks : Optimal and heuristic algorithms. *Computer Networks*, 53(7) :945 – 960, 2009. ISSN 1389-1286. (Cité page 61.)

I. Amundson, X. Koutsoukos, et J. Sallai. Mobile sensor localization and navigation using rf doppler shifts. Dans *Proceedings of the First ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*, MELT '08, pages 97–102, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-189-7. (Cité page 8.)

N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68, 1950. (Cité pages 21, 23, 26 et 27.)

C. Asensio-Marco et B. Beferull-Lozano. Fast average gossiping under asymmetric links in wsns. Dans *Signal Processing Conference (EUSIPCO), Proceedings of the 22nd European*, pages 131–135, Sept 2014. (Cité pages 93 et 97.)

J. E. Barcelo-Llado, A. M. Perez, et G. Seco-Granados. Enhanced correlation estimators for distributed source coding in large wireless sensor networks. *Sensors Journal, IEEE*, 12(9) :2799–2806, Sept 2012. (Cité page 61.)

R Battiti. First- and second-order methods for learning : Between steepest descent and newton's method. *Neural Computation*, 4(2) :141–166, March 1992. ISSN 0899-7667. (Cité pages 38 et 39.)

Y. P. Bergamo et C. G. Lopes. Scalar field estimation using adaptive networks. Dans *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pages 3565–3568, March 2012. (Cité page 16.)

D. P. Bertsekas et J. N. Tsitsiklis. *Parallel and Distributed Computation : Numerical Methods*. Athena Scientific, 1997. ISBN 1886529019. (Cité page 125.)

S. Boyd, A. Ghosh, B. Prabhakar, et D. Shah. Randomized gossip algorithms. *Information Theory, IEEE Transactions on*, 52(6) :2508–2530, June 2006. ISSN 0018-9448. (Cité pages 93, 97 et 98.)

J. R. Bunch et C. P. Nielsen. Updating the singular value decomposition. *Numerische Mathematik*, 31 :111–129, 1978. (Cité page 62.)

O. Burdakov, P. Doherty, K. Holmberg, J. Kvarnström, et P.-M. Olsson. Relay positioning for unmanned aerial vehicle surveillance. *I. J. Robotic Res.*, pages 1069–1087, 2010. (Cité page 7.)

S. Buruhanudeen, M. Othman, M. Othman, et B. M. Ali. Mobility models, broadcasting methods and factors contributing towards the efficiency of the manet routing protocols : Overview. Dans *Telecommunications and Malaysia International Conference on Communications, 2007. ICT-MICC 2007. IEEE International Conference on*, pages 226–230, May 2007. (Cité page 16.)

L. Cadariu et V. Radu. Fixed point methods for the generalized stability of functional equations in a single variable. *Fixed Point Theory and Applications*, 2008. (Cité page 38.)

E. Callaway, P. Gorday, L. Hester, J. A. Gutierrez, M. Naeve, B. Heile, et V. Bahl. Home networking with IEEE 802.15.4 : a developing standard for low-rate wireless personal area networks. *Communications Magazine, IEEE*, 40(8) :70–77, Aug 2002. ISSN 0163-6804. (Cité page 7.)

F. S. Cattivelli et A. H. Sayed. Diffusion strategies for distributed kalman filtering and smoothing. *Automatic Control, IEEE Transactions on*, 55(9) : 2069–2084, Sept 2010. ISSN 0018-9286. (Cité page 94.)

M. Cetin, L. Chen, J. W. Fisher, A. T. Ihler, R. L. Moses, M. J. Wainwright, et A. S. Willsky. Distributed fusion in sensor networks. *Signal Processing Magazine, IEEE*, 23(4) :42–55, July 2006. ISSN 1053-5888. (Cité page 9.)

C. Chatterjee. Adaptive algorithms for first principal eigenvector computation. *Neural Networks*, 18(2) :145–159, 2005. (Cité page 83.)

D. Chen et M. Wang. A home security zigbee network for remote monitoring application. Dans *Wireless, Mobile and Multimedia Networks, 2006 IET International Conference on*, pages 1–4, Nov 2006. (Cité page 7.)

F. Chen, F. Wen, et H. Jia. Algorithm of data compression based on multiple principal component analysis over the WSN. Dans *Wireless Communications Networking and Mobile Computing (WiCOM), 6th International Conference on*, pages 1–4, Sept 2010. (Cité page 62.)

J. Chen et A. H. Sayed. Diffusion adaptation strategies for distributed optimization and learning over networks. *Signal Processing, IEEE Transactions on*, 60(8) :4289–4305, Aug 2012. ISSN 1053-587X. (Cité page 72.)

L. Chen et S. Chang. An adaptive learning algorithm for principal component analysis. *IEEE Transactions on Neural Networks*, 6(5) :1255–1263, 1995. (Cité page 83.)

Q. Chen. *Studies in Autonomous Ground Vehicle Control Systems : Structure and Algorithms*. The Ohio State University, 2007. ISBN 9780549899631. (Cité page 7.)

J. C. Chin, Y. Dong, W. K. Hon, C. Y. T. Ma, et D. K. Y. Yau. Detection of intelligent mobile target in a mobile sensor network. *Networking, IEEE/ACM Transactions on*, 18(1) :41–52, Feb 2010. ISSN 1063-6692. (Cité page 17.)

N. Chitradevi, K. Baskaran, V. Palanisamy, et D. Aswini. Designing an efficient PCA based data model for wireless sensor networks. Dans *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief*, ACWR '11, pages 147–154, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-1011-6. (Cité page 62.)

C. Y. Chong et S. P. Kumar. Sensor networks : evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8) :1247–1256, Aug 2003. ISSN 0018-9219. (Cité page 7.)

J. Chou, D. Petrovic, et K. Ramchandran. A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks. Dans *Proceedings IEEE INFOCOM 2003, The 22nd Annual Joint Conference of the IEEE Computer and Communications Societies, San Franciso, CA, USA, March 30 - April 3*, 2003. (Cité page 61.)

G. Cirrincione, M. Cirrincione, J. Herault, et S. Van Huffel. The mca exin neuron for the minor component analysis. *Neural Networks, IEEE Transactions on*, 13(1) :160–187, Jan 2002. ISSN 1045-9227. (Cité pages 69, 77 et 78.)

S. W. Conner, J. Heidemann, L. Krishnamurthy, X. Wang, et M. Yarvis. Workplace applications of sensor networks. Dans Nirupama Bulusu et Sanjay Jha, éditeurs, *Wireless Sensor Networks : A Systems Perspective*. Artech House Publishers, 2004. (Cité page 1.)

T. F. Cox et M. A. A. Cox. *Multidimensional Scaling, Second Edition*. Chapman and Hall/CRC, 2 édition, 2000. ISBN 1584880945. (Cité page 10.)

F. Cucker et S. Smale. On the mathematical foundations of learning. *Bulletin of the American Mathematical Society*, 39(1), 2002. (Cité page 27.)

A. Czubak et J. Wojtanowski. On applications of wireless sensor networks. Dans Springer Berlin / Heidelberg, éditeur, *Internet - Technical Development and Applications*, volume 64, pages 91–99, 2009. (Cité page 1.)

C. Darken et J. E. Moody. Note on learning rate schedules for stochastic optimization. Dans Richard Lippmann, John E. Moody, et David S. Touretzky, éditeurs, *NIPS*, pages 832–838. Morgan Kaufmann, 1990. ISBN 1-55860-184-8. (Cité page 84.)

S. De Vito et G. Fattoruso. Wireless chemical sensor networks for air quality monitoring. Dans *14th International Meeting on Chemical Sensors-IMCS 2012*, pages 641–644, 2012. (Cité page 5.)

M. H. Degroot. Reaching a Consensus. *Journal of the American Statistical Association*, 69(345) :118–121, 1974. (Cité page 75.)

A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, et D. Terry. Epidemic algorithms for replicated database maintenance. Dans *Proceedings of the Sixth Annual ACM Symposium on Principles of Distributed Computing*, PODC '87, pages 1–12, New York, NY, USA, 1987. ACM. ISBN 0-89791-239-X. (Cité page 98.)

L. Doherty, K. S. J. pister, et L. El Ghaoui. Convex position estimation in wireless sensor networks. Dans *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 3, pages 1655–1663 vol.3, 2001. (Cité page 9.)

L. Doitsidis, S. Weiss, A. Renzaglia, M. Achtelik, E. Kosmatopoulos, R. Siegwart, et D. Scaramuzza. Optimal surveillance coverage for teams of micro aerial vehicles in GPS-denied environments using onboard vision. *Autonomous Robots*, 2012. (Cité page 7.)

R. D. Dony et S. Haykin. Neural network approaches to image compression. *Proceedings of the IEEE*, 83(2) :288–303, Feb 1995. ISSN 0018-9219. (Cité page 71.)

M. Dorigo et L. Gambardella. Ant Colonies for the Traveling Salesman Problem. *BioSystems*, 43 :73–81, 1997. (Cité page 17.)

F. Fagnani et S. Zampieri. Asymmetric randomized gossip algorithms for consensus. Dans *IFAC World Conference*, pages 9052–9056, 2008a. (Cité page 98.)

F. Fagnani et S. Zampieri. Randomized consensus algorithms over large scale networks. *Selected Areas in Communications, IEEE Journal on*, 26(4) : 634–649, May 2008b. ISSN 0733-8716. (Cité page 98.)

J. Fellus, D. Picard, et P. Gosselin. Dimensionality reduction in decentralized networks by gossip aggregation of principal components analyzers. Dans *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pages 171–176, Bruges, Belgique, April 2014. (Cité page 67.)

K. Fisher et C.-I. Chang. Progressive band selection for satellite hyperspectral data compression and transmission. *Journal of Applied Remote Sensing*, 4(1) :041770–041770–17, 2010. (Cité page 10.)

R. Fletcher et M. J. D. Powell. A rapidly convergent descent method for minimization. *The Computer Journal*, 6(2) :163–168, 1963. (Cité page 38.)

S. R. Gandham, M. Dawande, R. Prakash, et S. Venkatesan. Energy efficient schemes for wireless sensor networks with multiple mobile base stations. Dans *Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE*, volume 1, pages 377–381 Vol.1, Dec 2003. (Cité page 8.)

P. Geladi, H. Isaksson, L. Lindqvist, S. Wold, et K. Esbensen. Principal component analysis of multivariate images. *Chemometrics and Intelligent Laboratory Systems*, 5(3) :209 – 220, 1989. ISSN 0169-7439. (Cité page 62.)

G. R. Gerhart et B. A. Abbott. *Robotic and Semi-robotic Ground Vehicle Technology : 15-16 April 1998, Orlando, Florida*. Numéro vol. 3366 dans Proceedings of SPIE. Society of Photo-optical Instrumentation Engineers, 1998. ISBN 9780819428158. (Cité page 7.)

N. Ghadban, P. Honeine, C. Francis, F. Mourad-Chehade, J. Farah, et M. Kallas. Estimation locale d'un champ de diffusion par modèles à noyaux. Dans *Actes de la 14-ème conférence ROADEF de la Société Française de Recherche Opérationnele et Aide à la Décision*, Troyes, France, 13–15 Février 2013a. (Cité page 34.)

N. Ghadban, P. Honeine, C. Francis, F. Mourad-Chehade, J. Farah, et M. Kallas. Mobilité d'un réseau de capteurs sans fil basée sur les méthodes à noyau. Dans *Actes du 24-Ã¨me Colloque GRETSI sur le Traitement du Signal et des Images*, Brest, France, September 2013b. (Cité page 41.)

N. Ghadban, P. Honeine, F. Mourad-Chehade, C. Francis, et J. Farah. Mobility using first and second derivatives for kernel-based regression in wireless sensor networks. Dans *Proc. 21st International Conference on Systems, Signals and Image Processing*, Dubrovnik, Croatia, 12–15 May 2014. (Cité page 41.)

P. E. Gill, G. H. Golub, W. Murray, et M. A. Saunders. Methods for Modifying Matrix Factorizations. *Mathematics of Computation*, 28(126) :505–535, 1974. ISSN 00255718. (Cité page 62.)

M. Gönen et E. Alpaydin. Multiple kernel learning algorithms. *J. Mach. Learn. Res.*, 12 :2211–2268, Juillet 2011. ISSN 1532-4435. (Cité page 21.)

L Grippo, F Lampariello, et S Lucidi. A nonmonotone line search technique for newton's method. *SIAM J. Numer. Anal.*, 23(4) :707–716, Août 1986. ISSN 0036-1429. (Cité page 39.)

P. M. Hall, A. D. Marshall, et R. R. Martin. Merging and splitting eigenspace models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(9) :1042–1049, 2000. (Cité page 62.)

A. Hande, T. Polk, W. Walker, et D. Bhatia. Self-powered wireless sensor networks for remote patient monitoring in hospitals. *Sensors*, 6(9) :1102–1117, 2006. (Cité page 1.)

T. Hastie, R. Tibshirani, et J. Friedman. *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*. Springer, corrected édition, Août 2003. ISBN 0387952845. (Cité page 22.)

Y. Hatano et M. Mesbahi. Agreement over random networks. Dans *Decision and Control, CDC. 43rd IEEE Conference on*, volume 2, pages 2010–2015 Vol.2, Dec 2004. (Cité page 97.)

X. He et P. Niyogi. Locality preserving projections. Dans *In Advances in Neural Information Processing Systems 16*. MIT Press, 2003. (Cité page 10.)

M. Hefeeda et M. Bagheri. Wireless sensor networks for early detection of forest fires. Dans *Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE International Conference on*, pages 1–6, Oct 2007. (Cité page 1.)

A. E. Hoerl et R. W. Kennard. Ridge regression : Biased estimation for nonorthogonal problems. *Technometrics*, 12 :55–67, 1970. (Cité page 29.)

S. C. H. Hoi et R. Jin. Active kernel learning. Dans *International Conference on Machine Learning*, pages 400–407, 2008. (Cité page 22.)

P. Honeine. Online kernel principal component analysis : a reduced-order model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34 (9) :1814–1826, September 2012. (Cité pages 21, 39, 62, 69 et 84.)

P. Honeine, F. Mourad, M. Kallas, H. Snoussi, H. Amoud, et C. Francis. Wireless sensor networks in biomedical : body area networks. Dans *Proc. 7th International Workshop on Systems, Signal Processing and their Applications*, Algeria, 09–11 May 2011. (Cité page 6.)

P. Honeine et C. Richard. A closed-form solution for the pre-image problem in kernel-based machines. *Journal of Signal Processing Systems*, 65 : 289–299, December 2011a. (Cité page 20.)

P. Honeine et C. Richard. Preimage problem in kernel-based machine learning. *IEEE Signal Processing Magazine*, 28(2) :77–88, 2011b. (Cité page 20.)

P. Honeine, C. Richard, J. C. M. Bermudez, H. Snoussi, M. Essoloh, et F. Vincent. Functional estimation in hilbert space for distributed learning in wireless sensor networks. Dans *Proc. 34th IEEE International Conference on Acoustics, Speech and Signal Processing*, Taipei, Taiwan, April 2009. (Cité pages 16 et 34.)

P. Honeine, C. Richard, H. Snoussi, J. C. M. Bermudez, et J. Chen. A decentralized approach for non-linear prediction of time series data in sensor networks. *Journal on Wireless Communications and Networking*, 2010. (Cité pages 16 et 34.)

L. Hou et N. W. Bergmann. Novel industrial wireless sensor networks for machine condition monitoring and fault diagnosis. *Instrumentation and Measurement, IEEE Transactions on*, 61(10) :2787–2798, Oct 2012. ISSN 0018-9456. (Cité page 6.)

L. Huang, M. I. Jordan, A. Joseph, M. Garofalakis, et N. Taft. In-network pca and anomaly detection. Dans *NIPS*, pages 617–624. MIT Press, 2006. (Cité page 62.)

M. Ilyas, I. Mahgoub, et L. Kelly. *Handbook of Sensor Networks : Compact Wireless and Wired Sensing Systems*. CRC Press, Inc., Boca Raton, FL, USA, 2004. ISBN 0849319684. (Cité page 4.)

I. Jawhar, N. Mohamed, et D. P. Agrawal. Linear wireless sensor networks : Classification and applications. *Journal of Network and Computer Applications*, 34(5) :1671 – 1682, 2011. ISSN 1084-8045. Dependable Multimedia Communications : Systems, Services, and Applications. (Cité page 9.)

G. Jayakumar et G. Ganapathi. Reference point group mobility and random waypoint models in performance evaluation of manet routing protocols. *J. Comp. Sys., Netw., and Comm.*, 2008 :13 :1–13 :10, Janvier 2008. ISSN 1687-7381. (Cité page 16.)

S. Jia, Y. Qian, et Z. Ji. Band selection for hyperspectral imagery using affinity propagation. Dans *DICTA*, pages 137–141. IEEE Computer Society, 2008. ISBN 978-0-7695-3456-5. (Cité page 10.)

I. T. Jolliffe. *Principal Component Analysis*. Springer Verlag, 1986. (Cité pages 10 et 61.)

M. Kallas, P. Honeine, C. Richard, H. Amoud, et C. Francis. Nonlinear feature extraction using kernel principal component analysis with nonnegative pre-image. Dans *Proc. 32nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Buenos Aires, Argentina, 31 Aug. - 4 Sept. 2010. (Cité pages 10 et 30.)

H. Kargupta, W. Huang, K. Sivakumar, B. Park, et S. Wang. Collective principal component analysis from distributed, heterogeneous data. Dans *Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery*, pages 452–457, London, UK, UK, 2000. Springer-Verlag. ISBN 3-540-41066-X. (Cité page 66.)

D. Kempe, A. Dobra, et J. Gehrke. Gossip-based computation of aggregate information. Dans *Foundations of Computer Science, 2003. Proceedings. 44th Annual IEEE Symposium on*, pages 482–491, Oct 2003. (Cité page 98.)

G. S. Kimeldorf et G. Wahba. Some results on Tchebycheffian spline functions. *Journal of Mathematical Analysis and Applications*, 33(1) :82–95, 1971. (Cité page 27.)

S. B. Korada, A. Montanari, et S. Oh. Gossip pca. Dans *Proceedings of the ACM SIGMETRICS Joint International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '11, pages 209–220, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0814-4. (Cité pages 65 et 108.)

B. Krishnamachari, D. Estrin, et S. Wicker. The impact of data aggregation in wireless sensor networks. Dans *Distributed Computing Systems Workshops, 2002. Proceedings. 22nd International Conference on*, pages 575–578, 2002. (Cité page 61.)

V. Kurkova et M. Sanguineti. Learning with generalization capability by kernel methods of bounded complexity. *Journal of Complexity*, 21(3) :350 – 367, 2005. ISSN 0885-064X. (Cité page 27.)

H. Labiod, éditeur. *Réseaux mobiles ad hoc et réseaux de capteurs sans fil*. IC2. Hermes science publications, Paris, 2006. ISBN 2-7462-1292-7. (Cité page 8.)

T. P. Lambrou et C. G. Panayiotou. Collaborative event detection using mobile and stationary nodes in sensor networks. Dans *Collaborative Computing : Networking, Applications and Worksharing, 2007. CollaborateCom 2007. International Conference on*, pages 106–115, Nov 2007. (Cité page 18.)

T. P. Lambrou, C. G. Panayiotou, S. Felici, et B. Beferull. Exploiting mobility for efficient coverage in sparse wireless sensor networks. *Wireless Personal Communications*, 54(1) :187–201, 2010. ISSN 0929-6212. (Cité pages 18 et 52.)

P. Lancaster et M. Tismenetsky. *The Theory of Matrices : With Applications*. Computer Science and Scientific Computing Series. Academic press, 1985. ISBN 9780124355606. (Cité page 82.)

Y. Le Borgne, S. Raybaud, et G. Bontempi. Distributed principal component analysis for wireless sensor networks. *Sensors*, 8(8) :4821–4850, 2008. ISSN 1424-8220. (Cité page 64.)

M. H. Lee et Y. H. Choi. Fault detection of wireless sensor networks. *Computer Communications*, 31(14) :3469 – 3475, 2008. ISSN 0140-3664. (Cité page 9.)

S. H. Lee, S. Lee, H. Song, et H. S. Lee. Wireless sensor network design for tactical military applications : Remote large-scale environments. Dans *Proceedings of the 28th IEEE Conference on Military Communications*, MILCOM'09, pages 911–917, Piscataway, NJ, USA, 2009. IEEE Press. ISBN 978-1-4244-5238-5. (Cité page 4.)

J. Li et P. Mohapatra. Analytical modeling and mitigation techniques for the energy hole problem in sensor networks. *Pervasive Mob. Comput.*, 3 (3) :233–254, Juin 2007. ISSN 1574-1192. (Cité page 8.)

LibeliumWorld. Waspmote is released, Novembre 2009. URL `http://www.libelium.com/093283281523/`. (Cité page 2.)

W. R. Loughlin. Principal component analysis for alteration mapping. *Photogrammetric Engineering and Remote Sensing*, 57 :1163–1169, 1991. (Cité page 62.)

B. Lu, D. Gu, et H. Hu. Environmental field estimation of mobile sensor networks using support vector regression. Dans *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 2926–2931, Oct 2010. (Cité page 19.)

J. Lu et G. Chen. A time-varying complex dynamical network model and its controlled synchronization criteria. *Automatic Control, IEEE Transactions on*, 50(6) :841–846, June 2005. ISSN 0018-9286. (Cité page 97.)

F. Luo, R. Unbehauen, et A. Cichocki. A minor component analysis algorithm. *Neural Networks*, 10(2) :291 – 297, 1997. ISSN 0893-6080. (Cité pages 69, 77 et 78.)

J. MacGregor. Multivariate statistical methods for monitoring large data sets from chemical processes. AlChE Meeting, San Francisco, 1989. (Cité page 62.)

D. J. C. MacKay. Information-based objective functions for active data selection. *Neural Comput.*, 4(4) :590–604, Juillet 1992. ISSN 0899-7667. (Cité page 22.)

A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, et J. Anderson. Wireless sensor networks for habitat monitoring. Dans *WSNA'02*, Atlanta, Georgia, USA, September 2002. (Cité page 1.)

I. Maza, F. Caballero, J. Capitán, J. Martínez-de Dios, et A. Ollero. Experimental Results in Multi-UAV Coordination for Disaster Management and Civil Security Applications. *Journal of Intelligent &amp ; Robotic Systems*, pages 1–23, Décembre 2010. ISSN 0921-0296. (Cité page 7.)

Y. Miao et Y. Hua. Fast subspace tracking and neural network learning by a novel information criterion. *IEEE Transactions on Signal Processing*, 46 (7) :1967–1979, 1998. (Cité page 76.)

R. Möller. First-order approximation of gram-schmidt orthonormalization beats deflation in coupled pca learning rules. *Neurocomput.*, 69(13-15) : 1582–1590, Août 2006. ISSN 0925-2312. (Cité page 94.)

F. Mourad, H. Chehade, H. Snoussi, F. Yalaoui, L. Amodeo, et C. Richard. Controlled mobility sensor networks for target tracking using ant colony optimization. *Mobile Computing, IEEE Transactions on*, 11(8) :1261–1273, Aug 2012. ISSN 1536-1233. (Cité page 17.)

A. Nedic et A. Ozdaglar. Cooperative distributed multi-agent optimization. Dans Daniel P. Palomar et Yonina C. Eldar, éditeurs, *Convex Optimization in Signal Processing and Communications*. Cambridge University Press, 2009. (Cité page 75.)

I. Nevat, G. W. Peters, F. Septier, et T. Matsui. Estimation of spatially correlated random fields in heterogeneous wireless sensor networks. *Signal Processing, IEEE Transactions on*, 63(10) :2597–2609, May 2015. ISSN 1053-587X. (Cité page 9.)

X. Nguyen, M. I. Jordan, et B. Sinopoli. A kernel-based learning approach to ad hoc sensor network localization. *ACM Trans. Sen. Netw.*, 1(1) : 134–152, 2005a. (Cité page 16.)

X. Nguyen, M. J. Wainwright, et M. I. Jordan. Nonparametric decentralized detection using kernel methods. *IEEE Trans. Signal Processing*, 53 : 4053–4066, 2005b. (Cité page 16.)

A. A. Nimbalkar et D. Pompili. Reliability in underwater inter-vehicle communications. Dans *Proc. of ACM International Workshop on UnderWater Networks (WUWNet)*, pages 19–26, San Francisco, CA, USA, September 2008. (Cité page 8.)

E. Oja. Simplified neuron model as a principal component analyzer. *Journal of Mathematical Biology*, 15(3) :267–273, November 1982. ISSN 0303-6812. (Cité pages 69 et 71.)

J. B. Ong, Zhanping You, J. Mills-Beale, Ee Lim Tan, B. D. Pereles, et Keat Ghee Ong. A wireless, passive embedded sensor for real-time monitoring of water content in civil engineering materials. *Sensors Journal, IEEE*, 8(12) :2053–2058, Dec 2008. ISSN 1530-437X. (Cité page 7.)

M. D Plumbley. Lyapunov functions for convergence of principal component algorithms. *Neural Networks*, 8 :11–23, 1995. (Cité pages 69 et 76.)

T. Poggio et F. Girosi. A theory of networks for approximation and learning. *Laboratory, Massachusetts Institute of Technology*, 1140, 1989. (Cité page 23.)

P. Prabhakaran et R. Sankar. Impact of realistic mobility models on wireless networks performance. Dans *Wireless and Mobile Computing, Networking and Communications, 2006. (WiMob'2006). IEEE International Conference on*, pages 329–334, June 2006. (Cité page 17.)

D. Puccinelli et M. Haenggi. Wireless sensor networks : applications and challenges of ubiquitous sensing. *Circuits and Systems Magazine, IEEE*, 5 (3) :19 – 31, 2005a. ISSN 1531-636X. (Cité page 1.)

D. Puccinelli et M. Haenggi. Wireless sensor networks : Applications and challenges of ubiquitous sensing. *IEEE Circuits and Systems Magazine*, 5 : 19–31, 2005b. (Cité page 4.)

H. Robbins et S. Monro. A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 22(3) :400–407, 1951. (Cité page 38.)

H. Robbins et D. Siegmund. A convergence theorem for non negative almost supermartingales and some applications. Dans *Herbert Robbins Selected Papers*, pages 111–135. Springer, 1985. (Cité page 83.)

A. Rooshenas, H. R. Rabiee, A. Movaghar, et M. Y. Naderi. Reducing the data transmission in wireless sensor networks using the principal component analysis. Dans *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Sixth International Conference on*, pages 133–138, Dec 2010. (Cité page 62.)

T. D. Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks*, 2 :459–473, 1989. (Cité page 94.)

T. D. Sanger. Two iterative algorithms for computing the singular value decomposition from input/output samples. Dans J. D. Cowan, G. Tesauro, et J. Alspector, éditeurs, *Advances in Neural Information Processing Systems 6*, pages 144–151. Morgan-Kaufmann, 1994. (Cité page 94.)

A. H. Sayed, S. Tu, J. Chen, X. Zhao, et Z. J. Towfic. Diffusion strategies for adaptation and learning over networks : an examination of distributed strategies and network behavior. *Signal Processing Magazine, IEEE*, 30 (3) :155–171, May 2013. ISSN 1053-5888. (Cité pages 70 et 84.)

B. Schölkopf, R. Herbrich, et A. J. Smola. A Generalized Representer Theorem. Dans *COLT '01/EuroCOLT '01 : Proceedings of the 14th Annual Conference on Computational Learning Theory and and 5th European Conference on Computational Learning Theory*, pages 416–426, London, UK, 2001. Springer-Verlag. (Cité pages 21 et 34.)

B. Schölkopf, R. Herbrich, et R. Williamson. A generalized representer theorem. Rapport Technique NC2-TR-2000-81, Royal Holloway College, Univ. of London, UK, 2000. (Cité page 27.)

B. Schölkopf et A. J. Smola. *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001. ISBN 0262194759. (Cité page 21.)

B. Schölkopf, A. J. Smola, et K.-R. Müller. Kernel principal component analysis. *Advances in kernel methods : support vector learning*, pages 327–352, 1999. (Cité pages 10 et 30.)

J. Shawe-Taylor et N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA, 2004. ISBN 0521813972. (Cité pages 21 et 24.)

R. Shorey et C. M. Choon. *Mobile, Wireless and Sensor Networks : Technology, Applications and Future Directions*. Wiley-Interscience, 2005. ISBN 0471718165. (Cité page 8.)

F. Sivrikaya et B. Yener. Time synchronization in sensor networks : a survey. *Network, IEEE*, 18(4) :45–50, July 2004. ISSN 0890-8044. (Cité page 97.)

D. Slepian et J. K. Wolf. Noiseless coding of correlated information sources. *Information Theory, IEEE Transactions on*, 19(4) :471–480, Jul 1973. ISSN 0018-9448. (Cité page 61.)

K. Sohraby, D. Minoli, et T. Znati. *Wireless Sensor Networks : Technology, Protocols, and Applications*. Wiley-Interscience, 2007. ISBN 0471743003. (Cité page 2.)

V. Srivastava. A unified view of the orthogonalization methods. *Journal of Physics A : Mathematical and General*, 33(35) :6219, 2000. (Cité page 94.)

D. C. Steere, A. Baptista, D. Mcnamee, C. Pu, et J. Walpole. Research challenges in environmental observation and forecasting systems. Dans *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom'oo*, pages 292–299, 2000. (Cité page 1.)

N. K. Suryadevara et S. C. Mukhopadhyay. Wireless sensor network based home monitoring system for wellness determination of elderly. *Sensors Journal, IEEE*, 12(6) :1965–1972, June 2012. ISSN 1530-437X. (Cité page 6.)

N. Takahashi et I. Yamada. Parallel algorithms for variational inequalities over the cartesian product of the intersections of the fixed point sets of nonexpansive mappings. *Journal of Approximation Theory*, 153(2) :139 – 160, 2008. ISSN 0021-9045. (Cité page 125.)

A. Taleb et G. Cirrincione. Against the convergence of the minor component analysis neurons. *Neural Networks, IEEE Transactions on*, 10(1) : 207–210, Jan 1999. ISSN 1045-9227. (Cité page 78.)

J. Teng, H. Snoussi, et C. Richard. Decentralized variational filtering for target tracking in binary sensor networks. *IEEE Transactions on Mobile Computing*, 9(10) :1465–1477, 2010. (Cité page 1.)

A. Tiderko, T. Bachran, F. Hoeller, et D. Schulz. Rose - a framework for multicast communication via unreliable networks in multi-robot systems. *Robot. Auton. Syst.*, 56(12) :1017–1026, Décembre 2008. ISSN 0921-8890. (Cité page 8.)

A. N. Tikhonov et V. Y. Arsenin. *Solutions of Ill-Posed Problems*. V. H. Winston & Sons, Washington, D. C. : John Wiley & Sons, New York,, 1977a. (Cité page 23.)

A. N. Tikhonov et V. Y. Arsenin. *Solutions of Ill-posed problems*. W. H. Winston, 1977b. (Cité page 29.)

D. Trinchero, R. Stefanelli, D. Brunazzi, A. Casalegno, M. Durando, et A. Galardini. Integration of smart house sensors into a fully networked (web) environment. Dans *Sensors, 2011 IEEE*, pages 1624–1627, Oct 2011. (Cité page 7.)

S. Tu et A. H. Sayed. Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks. *Signal Processing, IEEE Transactions on*, 60(12) :6217–6234, Dec 2012. ISSN 1053-587X. (Cité page 76.)

V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, 1995. (Cité pages 22 et 23.)

J. P. Vert, K. Tsuda, et B. Scholkopf. A primer on kernel methods. *Kernel Methods in Computational Biology*, pages 35–70, 2004. (Cité page 24.)

M. A. M. Vieira, C. N. Coelho, Jr. da Silva, D. C., et J. M. da Mata. Survey on wireless sensor network devices. Dans *Emerging Technologies and Factory Automation, 2003. Proceedings. ETFA '03. IEEE Conference*, volume 1, pages 537–544 vol.1, Sept 2003. (Cité page 2.)

G. Wang, G. Cao, T. La Porta, et W. Zhang. Sensor relocation in mobile sensor networks. Dans *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 4, pages 2302–2312 vol. 4, March 2005. (Cité page 8.)

S. M. Watt. Pivot-free block matrix inversion. Dans *Proceedings of the Eighth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, SYNASC '06, pages 151–155, Washington, DC, USA, 2006. IEEE Computer Society. (Cité page 35.)

N. Watthanawisuth, A. Tuantranont, et T. Kerdcharoen. Design for the next generation of wireless sensor networks in battlefield based on zigbee. Dans *Defense Science Research Conference and Expo (DSR), 2011*, pages 1–4, Aug 2011. (Cité page 4.)

G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees, et M. Welsh. Fidelity and yield in a volcano monitoring sensor network. Dans *Proceedings of the 7th Symposium on Operating Systems Design and Implementation*, OSDI '06, pages 381–396, Berkeley, CA, USA, 2006. USENIX Association. ISBN 1-931971-47-1. (Cité page 9.)

B. M. Wise, N. Lawrence Ricker, et D. J. Veltkamp. Upset and sensor failure detection in multivariable processes. AlChE Meeting, San Francisco, 1989. (Cité page 62.)

A. D. Wyner et J. Ziv. The rate-distortion function for source coding with side information at the decoder. *Information Theory, IEEE Transactions on*, 22(1) :1–10, Jan 1976. ISSN 0018-9448. (Cité page 61.)

N. Xu. A survey of sensor network applications. *IEEE Communications Magazine*, 40, 2002. (Cité page 5.)

S. Yoon et C. Qiao. Cooperative search and survey using autonomous underwater vehicles (auvs). *IEEE Trans. Parallel Distrib. Syst.*, 22(3) :364–379, 2011. (Cité page 8.)

L. Yu, N. Wang, et X. Meng. Real-time forest fire detection with wireless sensor networks. Dans *Wireless Communications, Networking and Mobile Computing, 2005. Proceedings. 2005 International Conference on*, volume 2, pages 1214–1217, Sept 2005. (Cité page 5.)

F. Yuan, Y. Zhan, et Y. Wang. Data density correlation degree clustering method for data aggregation in WSN. *Sensors Journal, IEEE*, 14(4) :1089–1098, April 2014. ISSN 1530-437X. (Cité page 61.)

K. Yuan, Q. Ling, et W. Yin. On the convergence of decentralized gradient descent. *arXiv preprint arXiv :1310.7063*, 2013. (Cité page 75.)

J. Yuh. Design and control of autonomous underwater robots : A survey. *Auton. Robots*, 8(1) :7–24, 2000. (Cité page 8.)

M. Zakrzewski, S. Junnila, A. Vehkaoja, H. Kailanto, A.-M. Vainio, I. Defee, J. Lekkala, J. Vanhala, et J. Hyttinen. Utilization of wireless sensor network for health monitoring in home environment. Dans *Industrial Embedded Systems, 2009. SIES '09. IEEE International Symposium on*, pages 132–135, July 2009. (Cité page 6.)

Z. Zhu et T. S. Huang. *Multimodal Surveillance : Sensors, Algorithms, and Systems*. Artech House, 2007. ISBN 9781596931848. (Cité page 9.)

L. Zong, J. Houser, et T. R. Damarla. Multimodal unattended ground sensor (mmugs). Dans *Proc. SPIE*, volume 6231, page 623118, apr 2006. (Cité page 9.)

Y. Zou et K. Chakrabarty. Distributed mobility management for target tracking in mobile sensor networks. *Mobile Computing, IEEE Transactions on*, 6(8) :872–887, Aug 2007. ISSN 1536-1233. (Cité page 17.)

Nisrine GHADBAN

Doctorat : Optimisation et Sûreté des Systèmes

Année 2015

Fusion de l'information dans les réseaux de capteurs : application à la surveillance de phénomènes physiques

Cette thèse apporte des solutions clés à deux problèmes omniprésents dans les réseaux de capteurs sans fil, à savoir la précision des mesures acquises dans les régions à faible couverture et la dimensionnalité sans cesse grandissante des données collectées. La première contribution de cette thèse est l'amélioration de la couverture de l'espace à surveiller par le biais de la mobilité des capteurs. Nous avons recours aux méthodes à noyaux en apprentissage statistique pour modéliser un phénomène physique tel que la diffusion d'un gaz. Nous décrivons plusieurs schémas d'optimisation pour améliorer les performances du modèle résultant. Nous proposons plusieurs scenarios de mobilité des capteurs. Ces scénarios définissent d'une part l'ensemble d'apprentissage du modèle et d'autre part le capteur mobile. La seconde contribution de cette thèse se situe dans le contexte de la réduction de la dimensionnalité des données collectées par les capteurs. En se basant sur l'analyse en composantes principales, nous proposons à cet effet des stratégies adaptées au fonctionnement des réseaux de capteurs sans fil. Nous étudions également des problèmes intrinsèques aux réseaux sans fil, dont la désynchronisation entre les nœuds et la présence de bruits de mesures et d'erreurs de communication. Des solutions adéquates avec l'approche Gossip et les mécanismes de lissage sont proposées. L'ensemble des techniques développées dans le cadre de cette thèse est validé sur un réseau de capteurs sans fil qui estime le champ de diffusion d'un gaz.

Mots clés : réseaux de capteurs (technologie) - apprentissage automatique - réduction des données (statistique) - analyse en composantes principales - traitement du signal.

Information Aggregation in Sensor Networks: Application to Monitoring of Physical Activities

This thesis investigates two major problems that are challenging the wireless sensor networks (WSN): the measurements accuracy in the regions with a low density of sensors and the growing volume of data collected by the sensors. The first contribution of this thesis is to enhance the collected measurements accuracy, and hence to strengthen the monitored space coverage by the WSN, by means of the sensors mobility strategy. To this end, we address the estimation problem in a WSN by kernel-based machine learning methods, in order to model some physical phenomenon, such as a gas diffusion. We propose several optimization schemes to increase the relevance of the model. We take advantage of the sensors mobility to introduce several mobility scenarios. Those scenarios define the training set of the model and the sensor that is selected to perform mobility based on several mobility criteria. The second contribution of this thesis addresses the dimensionality reduction of the set of collected data by the WSN. This dimensionality reduction is based on the principal component analysis techniques. For this purpose, we propose several strategies adapted to the restrictions in WSN. We also study two well-known problems in wireless networks: the non-synchronization problem between nodes of the network, and the noise in measures and communication. We propose appropriate solutions with Gossip-like algorithms and smoothing mechanisms. All the techniques developed in this thesis are validated in a WSN dedicated to the monitoring of a physical species leakage such as the diffusion of a gas.

Keywords: sensor networks - machine learning - data reduction - principal components analysis – signal processing.

Thèse réalisée en partenariat entre :

Université Libanaise

utt université de technologie Troyes

Ecole Doctorale "Sciences et Technologies"