Data-driven control of interdependent energy systems

Billy Wladimir Toro Tovar

THÈSE DE DOCTORAT DE

L'École nationale supérieure Mines-Télécom Atlantique Bretagne Pays de la Loire, IMT Atlantique
délivrée conjointement avec
Universidad Nacional de Colombia

ÉCOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies de l'Information et de la Communication
Spécialité : *Automatique*

Par

Billy Wladimir TORO TOVAR

Data-Driven Control of Interdependent Energy Systems

Thèse présentée et soutenue à IMT Atlantique, le 14 decembre 2022
Unité de recherche : Le Laboratoire des Sciences du Numérique de Nantes
Thèse N° : 20022IMTA0337

Rapporteurs avant soutenance :

Laurent BAKO	Maître de Conférences HDR, Ecole Centrale de Lyon, Lyon, France
Juan Manuel REY	Associate Professor, Universidad Industrial de Santander, Bucaramanga, Colombia
Camilo CORTÉS	Professor, Universidad Nacional de Colombia, Bogotá, Colombia

Composition du Jury :

Président :	Camilo CORTÉS	Professor, Universidad Nacional de Colombia, Bogotá, Colombia
Examinateurs :	Pauline KERGUS	Chargée de recherche CNRS, LAPLACE - CODIASE, Toulouse, France
	Laurent BAKO	Maître de Conférences HDR, Ecole Centrale de Lyon, Lyon, France
	Juan Manuel REY	Associate Professor, Universidad Industrial de Santander, Bucaramanga, Col
Dir. de thèse :	Naly RAKOTO	Maitre-Assistant HDR, LS2N, IMT Atlantique, Nantes, France
Co-dir. de thèse :	Eduardo MOJICA-NAVA	Associate Professor, Universidad Nacional de Colombia, Bogotá, Colombia

Acknowledgement

I want to express my sincere gratitude to the people that participated in this process; and that motivated me to follow my work with their advice, support, and help. I want to recognize my mentor and advisor, Eduardo Mojica, who gave me the tools and the motivation to follow this path from the beginning of my formation. Furthermore, I would like to express my gratitude to my advisor Dr. Naly Rakoto, who gave me the opportunity to know a new world while following my project.

I would like to recognize the support of Colciencias 754 and the IMT-Atlantique from France.

Dedicatory

I dedicate this work to my father, who gave me the motivation to be better, and whom I miss so much. To my mother, who is always happy to support me, and to Lina, who gave me the possibility to build a present and future with me. To my friend Duvan Tellez, who influenced significantly my research and supported me to follow. I dedicated this work to my friends from the research group in Bogotá: Jhojan, Miguel, Carlos, and Fabian, and to my friends from France: Aline, Wilmer, Hamid-Reza, Paula, Maria, Simon, Alexander, Ilhem, David, Essan, Karim, and Francois.

Résumé en français

Cette recherche a proposé plusieurs algorithmes pour l'identification et le commande des micro-réseaux basés sur l'opérateur Koopman. Les contributions présentées dans ce manuscrit se concentrent sur le comman de de la tension et de la puissance réactive. Nous avons considéré cinq scénarios de commande basés sur l'opérateur de Koopman : (i) un algorithme centralisé qui régule la tension du micro-réseau sans partager l'information en utilisant MPC. (ii) une commande distribuée non coopérative, avec un terme de consensus dans les restrictions, qui régule la tension en fonction du modèle de Koopman des onduleurs. (iii) un MPC distribué coopératif qui utilise le modèle Koopman du micro-réseau, où les agents partagent leurs entrées de commande pour générer les signaux de commandes. Ici, nous identifions les matrices d'entrée en utilisant des données. (iv) un commande distribuée qui utilise les données pour identifier l'erreur du système afin de concevoir un algorithme ADMM. (v) une commande en ligne pilotée par les données qui régule la tension du micro-réseau et une analyse des valeurs propres du système et des effets des mesures bruitées.

Table of Contents

List of Figures

List of Tables

List of acronyms

AC	Alternate Current
ADMM	Alternate Direction Multipliers Method
DADMM	Distributed Alternate Direction Multipliers Method
DBC	Data-Based Control
DDC	Data-Driven Control
DMD	Dynamic Mode Decomposition
EDMD	Extended Dynamic Mode Decomposition
MBC	Model-Based Control
MG	Microgrid
MPC	Model Predictive Control
PCA	Principal Component Analysis
PPD	Pseudo-Partial Derivative
RL	Reinforcement Learning
RMS	Root Mean Square
SVD	Singular Value Decomposition

Résumé Long

De nos jours, les systèmes électriques comprennent le stockage, la production distribuée et les charges interactives. Les systèmes distribués sont donnés par les générateurs locaux et surtout par les sources renouvelables, c'est-à-dire les générateurs solaires et éoliens. Les charges interactives peuvent inclure des véhicules électriques ayant la capacité de partager l'énergie avec le réseau, et les systèmes de stockage peuvent comprendre des bancs de batteries et des systèmes à volant d'inertie. Tous ces composants doivent être gérés correctement en définissant des points de consigne locaux et en partageant la puissance entre eux. Le micro-réseau (MG) est une solution au problème du contrôle des sources distribuées. Il s'agit d'un système cyber-physique ayant la capacité de fonctionner de manière isolée du réseau électrique tout en conservant les valeurs de référence de ses variables. D'autres systèmes ont également une structure distribuée, comme les systèmes d'approvisionnement en eau, les systèmes de drainage, les véhicules autonomes, les escadrons de drones et les réseaux de capteurs. Certains de ces systèmes présentent un couplage fort dans leurs grandeurs physiques ; par exemple, un changement dans le débit d'eau d'une ligne modifie la pression dans l'ensemble du système. D'autres permettent d'effectuer des changements sans affecter la dynamique des autres agents. Dans le cas des systèmes électriques, ils présentent une structure couplée inhérente où les changements dans la production, les charges et les défaillances affectent le comportement de l'ensemble du système. En outre, la modélisation des systèmes en réseau peut être problématique en raison de l'absence d'un modèle précis, des conditions changeantes des paramètres et de leur comportement non linéaire. De nouvelles techniques axées sur les données ont été développées ces dernières années, [20], exploitant les données disponibles provenant de capteurs. Les mesures d'un système peuvent être utilisées pour modéliser des systèmes non linéaires et pour la conception de commandes. Plusieurs techniques axées sur les données ont été proposées, dont celle basée sur l'opérateur de Koopman [36]. Le principal avantage de l'utilisation de l'opérateur de Koopman est sa versatilité à représenter un système non linéaire comme un système linéaire dans un nouvel espace connu sous le nom d'espace levé. L'extension de ces algorithmes aux systèmes en réseau est encore un problème ouvert et son utilisation dans les systèmes électriques n'a pas encore été

présentée. Le thème principal de cette thèse est l'application et l'analyse d'algorithmes qui utilisent l'opérateur de Koopman pour commander des systèmes distribués en utilisant des données.

Plusieurs problèmes peuvent être représentés en utilisant le cadre multi-agent. La nature distribuée de ces phénomènes rend appropriée l'utilisation de la théorie des graphes. Chaque système dynamique local est représenté comme un nœud, et les communications entre eux sont représentées comme des liens dans un graphe, où l'interaction entre les nœuds comprend plusieurs comportements donnés par le couplage. Le MG peut être défini comme un courant continu (CC) ou alternatif (AC) qui est donné par le courant géré par la connexion principale dans le MG. De plus, lorsque le MG est connecté au réseau principal (réseau d'utilité publique), il fonctionne comme une source de courant contrôlée ; en revanche, si le MG est en mode îloté, il fonctionne comme une source de tension contrôlée. Dans le premier cas, la MG établit uniquement le point de consigne de la puissance, tandis que dans le second cas, la MG établit les points de consigne de la tension, de la fréquence et d'autres variables.

Il est nécessaire de commander la tension et la fréquence ou l'angle de phase pour commander la puissance active et réactive lorsque le MG est isolé. Ici, les lignes de transmission ont un effet pertinent sur le comportement global. La partie résistive et réactive de la ligne de transmission détermine le comportement du flux de puissance active et réactive entre les sources. Les flux de puissance peuvent être commandés séparément dans une ligne de transmission à prédominance réactive (sans perte dans le cas idéal). La puissance active est commandée en modifiant l'angle de phase ou la fréquence, et la puissance réactive est commandée en modifiant l'amplitude de la tension.

L'interaction entre les générateurs de tension implique des comportements non linéaires dus au flux entre les agents. La modélisation et le commande appropriés d'un système distribué, avec un comportement non linéaire et des paramètres difficiles à établir, impliquent un problème de pointe. Les données sont une alternative pour travailler avec ce type de phénomènes. Comme le nombre de capteurs a augmenté, de grandes quantités de données sont disponibles. Ces données peuvent être utilisées pour le commande et l'identification de systèmes non linéaires. Certaines approches ont été proposées avec des algorithmes pour exploiter les données, comme l'apprentissage itératif, la commande sans modèle, l'apprentissage par renforcement, etc. Du point de vue de la conception de la commande, la disponibilité d'un modèle de système est plus souhaitable qu'une approche complète de type boîte noire. Certaines techniques, comme l'apprentissage itératif, exploitent les

données pour régler les contrôleurs PID classiques. Les techniques d'apprentissage par renforcement sont basées sur la programmation dynamique ; elles peuvent utiliser une approche boîte noire complète. Mais il existe également différentes approches qui établissent des conditions de stabilité pour la conception des commandes.

Une alternative à la technique présentée précédemment est celle basée sur les opérateurs linéaires, qui généralisent le concept de carte linéaire. En particulier, l'opérateur de Koopman est un opérateur linéaire dont l'adjoint est l'opérateur de Moore-Penrose qui agit sur les distributions. L'opérateur de Koopman permet de représenter un système non linéaire comme un système linéaire de dimensions infinies dans l'espace des levées ou l'espace des observables. Les observables peuvent être définies comme des mesures du système à partir d'une variable d'état sélectionnée. Déterminer la représentation de Koopman d'un système dans une dimension finie n'est pas une tâche simple. Certaines techniques axées sur les données ont été étudiées, telles que la décomposition en modes dynamiques (DMD) et le cas plus général de la décomposition en modes dynamiques étendus (EDMD), comme réponse à ce problème. La DMD représente un système non linéaire comme un système linéaire de dimension finie en utilisant une minimisation des moindres carrés sur l'ensemble des observables ou en appliquant la décomposition en valeurs singulières (SVD). EDMD définit un ensemble de fonctions non linéaires, appelé dictionnaire, et évalue, sur celui-ci, l'ensemble des observables. L'EDMD résout un problème de minimisation des moindres carrés pour déterminer la matrice de Koopman qui représente le système non linéaire. La manière de déterminer le bon dictionnaire de fonctions est un problème ouvert, avec plusieurs types de fonctions proposées, telles que les fonctions radiales, les fonctions de Fourier, et les polynômes. Dans plusieurs cas, la nature propre du problème peut suggérer les fonctions du dictionnaire. De plus, le nombre de fonctions choisies détermine la dimension de l'approximation de Koopman. Il est également possible de trouver la représentation approximative de Koopman des systèmes avec entrée de commande en supposant que le système est affine et en incluant un signal de commande qui varie dans le temps.

L'acquisition de données, pour l'identification et la conception de la commande de systèmes non linéaires, peut se faire à partir de données provenant de capteurs ou de données issues de simulations détaillées. Elle inclut tous les paramètres possibles pour être plus précis. Les données sont générées pour plusieurs trajectoires du système en effectuant des simulations en faisant varier les conditions initiales. L'entrée de commande varie également pour déterminer la matrice d'entrée. Les données sont échantillonnées

par une période donnée par la réponse inhérente à chaque système. La matrice de sortie est récupérée en comparant l'ensemble des données d'entrée, et l'ensemble des données évaluées sur le dictionnaire.

La représentation linéaire approximée dans l'espace de Koopman est adaptée à la conception de commandes. Premièrement, elle fonctionne comme un prédicteur linéaire dans la commande prédictive par modèle (MPC). Ensuite, l'analyse des valeurs propres de la matrice de Koopman permet d'étudier la stabilité du système et la structure du commande en vérifiant son observabilité et sa contrôlabilité. La conception d'une identification et d'un commande appropriés peut être problématique dans les systèmes en réseau, en particulier s'il existe un fort couplage entre les agents. Lorsqu'il y a un couplage entre les agents, les changements sur les variables d'un agent affectent les variables des autres agents. Une alternative pour commander des systèmes distribués, même avec un couplage, est d'utiliser le MPC distribué. Plusieurs techniques ont été proposées pour résoudre le MPC de manière distribuée : non coopérative, coopérative, ADMM distribuée, etc. Dans la première, les agents n'ont pas besoin de connaître l'action de commande prise par chaque agent et partagent uniquement son état. Dans le second, les agents partagent leurs états et leurs signaux de commande. Dans la dernière, les agents partagent une variable auxiliaire, et le problème est résolu de manière itérative.

Pour faire face à la nature en réseau des systèmes électriques et, en général, des systèmes distribués, l'identification de chaque agent doit tenir compte de l'effet des autres agents. Une approche simple du problème consiste à concevoir un MPC distribué non coopératif. Ainsi, un terme de consensus, qui réduit les différences d'état entre les agents, est inclus dans les restrictions du problème d'optimisation. L'identification de chaque agent se fait de la même manière que pour le modèle décentralisé ; cependant, il est nécessaire de définir les gains pour le MPC, et le gain pour le terme de consensus. Dans le cas du MG, le commande conçu peut réguler la tension par des changements de charge, des changements dans les lignes de transmission, et des changements dans le réseau de communication, qui sont les liens dans le graphique.

Objectifs de la thèse et contributions

Cette thèse étudie le commande des systèmes énergétiques distribués en utilisant des données. La modélisation d'un système distribué est limitée en raison de la difficulté de définir un cadre mathématique précis. Les systèmes énergétiques sont couplés et présen-

tent des variations lors de leur fonctionnement. Dans cette thèse, nous exploitons les données disponibles du système, issues de la simulation de modèles basés sur des principes fondamentaux. Le MG est également un système distribué et nécessite de définir une stratégie de contrôle garantissant son bon fonctionnement. Les algorithmes proposés pour les systèmes non linéaires sont exploités pour commander un MG défini comme un système distribué. Les principales contributions de cette thèse sont résumées comme suit

— La conception centralisée basée sur Koopman utilisant MPC régule la tension du MG tout en maintenant la condition de partage de puissance pour la puissance réactive. Pour l'identification du système, des données sont recueillies auprès de chaque agent pour différentes conditions initiales ; de plus, l'entrée varie pour déterminer la matrice d'entrée. Dans ce cas, chaque agent n'a accès qu'aux mesures de tension locale, et le MPC minimise la différence entre la tension locale, la référence et l'action de commande. Comme chaque agent ne prend des mesures que sur la base de ses mesures, le commande n'atteint qu'un optimum local. Le système complet régule la tension et peut atteindre de bonnes performances ; cependant, certains problèmes peuvent apparaître lorsque le réseau s'étend, ce qui est un sujet de travail futur.

— Pour les systèmes distribués, le MPC doit connaître les actions de commande effectuées par les autres agents. Une approche distribuée peut être utilisée pour éviter le problème d'une conception centralisée avec une communication complète entre les agents. Chaque agent connaît alors l'état et l'action de commande de son ensemble de voisins. Cette approche est connue sous le nom de MPC distribué coopératif ; elle nécessite de connaître la matrice d'entrée de couplage entre les agents. Elle peut être déterminée en faisant varier le signal de commande de chaque agent et en vérifiant comment les autres agents sont affectés. La technique pilotée par les données avec EDMD facilite la vérification de l'effet des signaux de commande locaux sur les autres agents. Une matrice d'entrée globale est construite pour l'ensemble du système. Pour le problème d'optimisation, la fonction objectif minimise l'état et l'entrée de commande, et les restrictions incluent la représentation linéaire du modèle non linéaire dans l'espace soulevé, avec un terme pour le consensus d'état, et un terme pour l'effet des entrées de commande sur les agents qui dépend de la matrice d'entrée globale. Le MPC coopératif a l'avantage d'atteindre une valeur Pareto-optimale par rapport à l'équilibre de Nash atteint par le MPC non-coopératif, et l'optimum local atteint par le contrôleur décentral-

isé. La convergence de chaque algorithme est déterminée, y compris les conditions pour les gains MPC en utilisant le calcul des variations pour le cas non coopératif, et la fonction de Lyapunov pour le cas coopératif. Les conditions de stabilité de la représentation linéaire du système dans l'espace de Koopman sont vérifiées en contrôlant les valeurs propres de la matrice d'état ; les conditions de contrôlabilité et d'observabilité du modèle linéaire sont également vérifiées.

— Une alternative pour résoudre le problème d'optimisation MPC consiste à utiliser un algorithme alternatif basé sur les multiplicateurs de Lagrange, connu sous le nom de méthode ADMM (Alternative Direction Multipliers Method). Cet algorithme divise le processus d'optimisation en deux étapes. Dans la première, l'entrée optimale est calculée ; dans la seconde, le multiplicateur de Lagrange est mis à jour. Il est nécessaire de définir le problème comme un problème de régulation pour définir le MPC au lieu d'un problème de suivi. Ainsi, en éliminant la dépendance de la référence, la commande minimise l'erreur d'état et l'entrée de commande. L'algorithme ADMM est utilisé pour résoudre le problème sous une forme distribuée ; cet algorithme, appelé ADMM distribué (DADMM), utilise la matrice Laplacienne pour calculer la variable auxiliaire partagée avec l'ensemble des voisins de chaque agent. Certaines hypothèses doivent être faites pour garantir la convergence de l'algorithme. Dans ce cas, la dynamique d'erreur de chaque agent est identifiée en utilisant EDMD, en définissant le même dictionnaire que précédemment. Cet algorithme n'a besoin que de l'information partagée par les variables auxiliaires, car il peut être implémenté dans le matériel. L'optimisation à chaque itération, peut être résolue par un algorithme récursif, tel que l'algorithme step descendent. Cet algorithme régule la tension du MG par des changements de charge, de ligne de transmission et de réseau de communication. Dans le dernier cas, lorsqu'un changement dans les graphes déconnecte deux nœuds, la variable auxiliaire entre eux passe à zéro.

— Dans la dernière partie, l'effet du bruit sur l'identification et la commande des systèmes dynamiques est étudié. Premièrement, les mesures bruitées affectent l'identification correcte des matrices espace-état. Ce problème peut être traité en incluant un terme de correction dans l'algorithme EDMD. Il s'agit d'un terme de norme 1 inclus dans la minimisation des moindres carrés pour déterminer la représentation de Koopman du système. Une autre alternative consiste à définir un second dictionnaire de fonctions comme proposé dans la littérature. Ce deux-

ième ensemble de fonctions doit être différent de l'ensemble proposé pour le problème. Ainsi, deux matrices peuvent être calculées en évaluant l'ensemble des mesures dans les deux dictionnaires, puis la représentation de Koopman du système. L'ensemble des valeurs propres de la matrice d'état utilisant l'algorithme EDMD avec des mesures bruitées est comparé à l'ensemble des valeurs propres générées par l'algorithme EDMD à deux dictionnaires. Dans le cas de la MG, nous avons proposé d'utiliser un ensemble de fonctions exponentielles, ainsi que le dictionnaire utilisé dans les premiers chapitres. L'algorithme à deux dictionnaires approxime mieux les valeurs propres au cas idéal avec des mesures sans bruit. Pour la conception de la commande à l'aide de l'approche de Koopman incluant l'effet des perturbations et du bruit, il est possible d'introduire un terme supplémentaire dans le prédicteur linéaire pour inclure l'effet du bruit ou des perturbations bornées. Le prédicteur linéaire, qui fonctionne dans l'espace soulevé, considère le bruit ou les perturbations sous une forme limitée. Cela peut être très utile dans le cas des systèmes électriques pour le bruit ou les perturbations bornées produites dans les mesures de tension. L'algorithme proposé dans cette thèse inclut une perturbation bornée ; il utilise la conception de Koopman et peut rejeter ces perturbations tout en conservant la régulation de tension.

— Enfin, nous avons proposé une commande en ligne pilotée par les données, basée sur une approche de la littérature pour un problème récursif des moindres carrés. Cet algorithme n'a besoin que des deux ensembles pour DMD ou EDMD et de la dernière paire de mesures à venir (la mesure et sa version décalée). Dans la littérature, l'algorithme proposé fonctionne avec DMD, mais, dans notre cas, nous avons utilisé l'algorithme pour résoudre un problème d'optimisation EDMD car nous avons utilisé un petit dictionnaire. Le MPC est conçu comme dans l'approche décentralisée, mais l'équation de contrainte, qui a la matrice d'état du prédicteur linéaire donnée par le modèle de Koopman, est mise à jour à une période prédéfinie. Comme le montrent les simulations, le MG régule la tension lorsqu'il y a des changements dans les charges et les lignes de transmission. L'algorithme commence avec une matrice d'état identique mise à jour avec des données, et la matrice d'entrée est déterminée dans des simulations précédentes et maintenue constante pendant la simulation. Le système doit être observable et contrôlable pour garantir la convergence de cet algorithme. Ainsi, l'ensemble de données utilisé pour déterminer la matrice d'état doit être suffisamment grand pour avoir une matrice d'état de

rang complet. La taille de l'ensemble de données pour remplir ces conditions est un problème ouvert.

— Tous les algorithmes présentés dans ce travail ont permis d'améliorer l'étude et la conception des commandes pilotées par les données. Ils sont adaptés aux systèmes à structure changeante, où il est difficile d'établir un modèle détaillé en raison de la structure du problème ou de sa non-linéarité. De plus, cet algorithme établit une connexion avec les systèmes distribués donnant une meilleure approximation des systèmes de puissance.

Structure du manuscrit et résumé des chapitres

Le manuscrit de thèse est composé de quatre chapitres, qui sont résumés dans les paragraphes suivants

Chapitre 1 - Les micro-réseaux et les techniques basées sur les données : État de l'art

Ce chapitre présente un cadre général du problème pour les systèmes électriques et les MG, à savoir comment l'utilisation de ressources distribuées réduit l'inertie des systèmes électriques. Il présente également le concept de MG, les approches basées sur les données, ainsi que le concept de l'opérateur de Koopman et ses algorithmes basés sur les données. Enfin, ce chapitre présente les concepts généraux des systèmes distribués et de l'optimisation, ainsi qu'une discussion sur le problème non linéaire.

Chapitre 2 - Commande décentralisée du MG basée sur Koopman

Ce chapitre définit le modèle dynamique de la MG et le problème d'optimisation pour réguler la tension. Le cadre de commande secondaire régule la tension ; pour ce faire, nous définissons un MPC dont les contraintes comprennent un prédicteur linéaire qui est trouvé en utilisant les mesures de tension et l'approximation de l'opérateur de Koopman basée sur les données. Cette commande secondaire proposée régule la tension de MG lorsqu'il y a des changements de charge et de ligne de transmission. Enfin, l'algorithme proposé

est comparé à un contrôleur qui utilise un MPC avec un prédicteur non linéaire dans ses contraintes, montrant que l'algorithme basé sur Koopman réduit le coût de calcul.

Chapitre 3 - Approches distribuées pour les commandes basées sur Koopman

Ce chapitre présente trois algorithmes de commande pour les systèmes distribués appliqués à la commande secondaire dans les MG. Tout d'abord, il est présenté un MPC non coopératif qui inclut une équation de consensus comme contrainte avec le prédicteur linéaire déterminé en utilisant l'opérateur Koopman avec EDMD. Deuxièmement, nous présentons un MPC coopératif qui régule la tension du MG en tenant compte de l'état et des actions de contrôle de chaque agent du voisinage en utilisant l'algorithme piloté par les données pour définir la matrice d'entrée des systèmes interconnectés. Troisièmement, il est présenté un algorithme d'optimisation itérative basé sur l'ADMM distribué. Dans ce cas, le prédicteur de Koopman est défini comme un problème de régulation au lieu d'un problème de suivi. Nous étudions la convergence et les hypothèses de chaque algorithme pour réguler la tension à travers les changements de charge, de ligne de transmission et de graphe.

Chapitre 4 - Conception en ligne pilotée par les données et MPC amélioré

Ce chapitre présente une commande en ligne pilotée par les données pour les MG, basée sur un algorithme récursif des moindres carrés issu de la littérature. L'algorithme en ligne utilise l'EDMD au lieu du DMD en définissant un petit ensemble de fonctions dans le dictionnaire. L'algorithme proposé régule la tension de la MG tout en mettant à jour sa matrice d'état avec les données stockées et la dernière paire de mesures à venir. Ce chapitre présente également un dictionnaire de fonctions à utiliser pour déterminer la matrice de Koopman d'un système avec des mesures bruitées. Enfin, nous avons présenté un algorithme de rejet du bruit incluant un terme de perturbation dans le prédicteur linéaire en utilisant EDMM.

Exemples de résultats

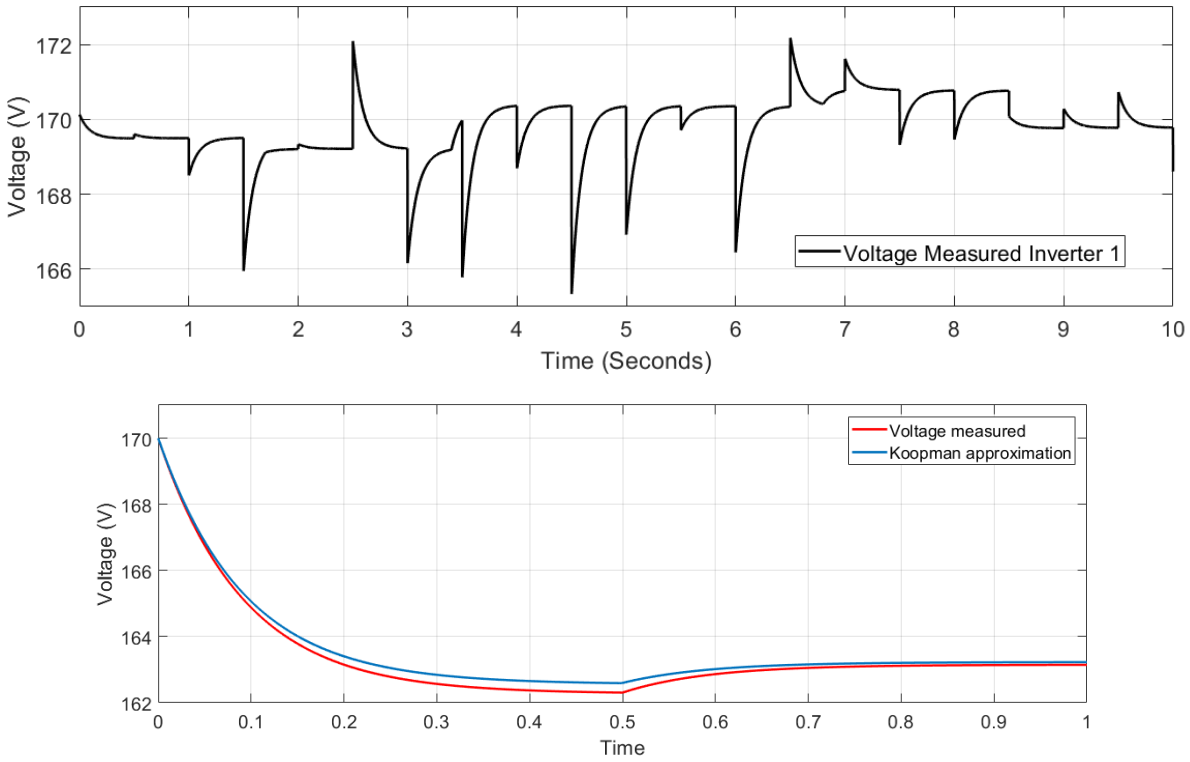

Figure 1 – Données utilisées pour déterminer l'opérateur de Koopman et l'approximation de Koopman.

Figure 1 montre un segment des données utilisées pour déterminer l'opérateur de Koopman. Les conditions initiales varient de 0 à 200 v. Un deuxième ensemble est utilisé pour vérifier la validité de l'approximation. Ceci est montré ci-dessous, notez que pendant les premières étapes, la matrice de Koopman se rapproche étroitement de la dynamique du système.

Conclusions et perspectives

Ce document se concentre sur l'identification et le commandement des MG. L'identification correcte des systèmes électriques peut être un problème difficile en raison de leur structure en réseau, de leur hétérogénéité, de leurs différentes échelles de temps et de leurs incertitudes. La disponibilité de grandes quantités de données du système permet l'utilisation

de techniques orientées données pour améliorer l'identification et le commande des systèmes interconnectés. Nous avons considéré différentes approches de la vue du commande de MG comme un système distribué. Ce document présente plusieurs contributions à la commande des systèmes en réseau et des MG. La commande décentralisée pilotée par les données utilisant la théorie de l'opérateur de Koopman régule la tension du MG à l'aide de techniques d'optimisation. L'algorithme proposé exige que chaque générateur utilise des mesures locales en ligne pour générer des actions de commande. Nous concevons un prédicteur linéaire local dont les valeurs propres sont à l'intérieur du cercle unitaire qui garantit la convergence de l'algorithme. Le prédicteur linéaire réduit également le temps de calcul par rapport aux autres commandes qui proposent un MPC non linéaire. L'algorithme proposé est sensible aux changements dans les lignes de transmission, en particulier dans l'inductance locale, ce qui nécessite également de recalculer la représentation de Koopman. De plus, l'algorithme omet certains comportements des systèmes en réseau, tels que l'effet des actions de commande entre les agents.

Un ensemble d'algorithmes, pour les systèmes distribués avec un prédicteur linéaire basé sur Koopman, a été présenté : ADMM non coopératif, coopératif et distribué. L'algorithme non coopératif utilise uniquement les mesures de tension des autres générateurs pour réguler sa tension en utilisant un terme de consensus linéaire. Nous avons démontré que le système converge vers la valeur de référence. Pour le MPC coopératif, nous avons proposé un algorithme hors ligne pour obtenir la matrice d'entrée des générateurs en faisant varier le signal de commande et en mesurant son effet sur les autres générateurs. Nous avons proposé un troisième algorithme qui utilise la représentation de l'erreur dans l'espace de Koopman pour concevoir un MPC qui est résolu en utilisant un ADMM distribué. Il réduit le temps de calcul et améliore le coût de calcul. Les trois algorithmes convergent vers les valeurs de référence, en fonction des matrices de poids et de la sélection des paramètres. Nous avons également utilisé un prédicteur linéaire basé sur Koopman qui inclut une matrice de perturbation. La simulation montre que le contrôleur MPC, utilisant la représentation de Koopman, régule la tension dans le MG pour les changements de charge, les changements de ligne de transmission et les changements de graphe de communication. L'identification correcte de la dynamique de chaque générateur dépend du traitement correct des données disponibles dans ce premier cas. Comme les variations rapides, telles que les composantes haute fréquence, pourraient affecter le calcul correct de la représentation de Koopman, un filtre passe-bas a été utilisé.

Un algorithme proposé traite le problème des mesures bruitées pour déterminer la

représentation de Koopman du système. Ces outils sont un facteur clé pour la conception de l'algorithme de Koopman en ligne que nous avons proposé pour réguler la tension dans le MG. Le problème de la mise à jour de la matrice de Koopman est résolu en utilisant un algorithme qui n'utilise qu'une paire de mesures à chaque itération. L'algorithme proposé converge en suivant un ensemble de restrictions liées aux données mesurées, et avec le temps fixé pour faire l'actualisation. Enfin, un algorithme qui utilise un terme pour les perturbations identifiées par EDMD est présenté et utilisé pour concevoir un MPC capable de rejeter les perturbations de tension.

Les perspectives de ce travail sont liées à l'amélioration de l'algorithme en ligne et au problème de travailler avec des mesures bruyantes. Un autre aspect est le traitement des lignes de transmission sans perte, qui devrait inclure l'angle de phase entre la tension et le courant. Le travail avec l'opérateur de Koopman dans les systèmes distribués peut également aborder une identification correcte des nœuds avec plus d'interaction avec d'autres agents dans un réseau couplé, tel que le système électrique.

INTRODUCTION

0.1 General Context

Power systems have changed from the traditional utility network to a more distributed and variable system. This phenomenon is due to the growth of renewable sources, storage systems, and electronic loads. In recent years, it has been a global increase of two times in wind and six times in solar energy production [1], [2]. The number of non-synchronous sources in the power system has increased in recent years, such as the case of the Irish system, with almost 60% of variable renewable energy, and the Australian National Energy Market (NEM) with 30% of wind and 15% of solar capacity installed. It advocates the power system to new challenges, especially when synchronous machines are replaced by non-rotational sources based on electronic converters, causing the system to have a low inertia [3].

A microgrid (MG) is a cyber-physical system that integrates generators, loads, and storage systems in a defined geographical area. It can operate connected to the power grid (utility) or as an isolated unit (islanded mode). On one hand, the MG allows the integration of non-rotational sources in an organized form; on the other hand, it brings new challenges due to non-conventional sources, phenomena such as heterogeneity, multiple time scales, and nonlinear behavior. A system with low inertia presents additional issues because power electronic converters introduce faster dynamics compared to synchronous generators, then, unexpected coupling might be present, and control approaches based on time-scale separation might be less valid, even resulting in unstable behavior [4]. In these systems, an accurate model of electronic converters is needed. However, this is not always available and might be not practical for analysis and control design in interconnected systems [3]. Many control strategies rely on the accurate description of the system by a dynamical model, usually by a set of nonlinear equations. It can be difficult, especially if there is high uncertainty about the model parameters, or even prohibitive when a model is unavailable because of a high-order system or the nature of the physical phenomena.

Low prices in sensors have given access to a large amount of data. Simultaneously, the increase in computing power has brought new developments in tools for data analysis.

One perspective uses data series to generate a dynamic system model and to design control strategies. In particular, the Koopman operator is a linear operator that acts in the space of observables and allows the representation of a nonlinear system as a linear combination of new bases in an infinity dimension [5]. Koopman operator has been used in different fields such as power systems control [6], [7], [8] flow dynamics, video processing, neuroscience, and economy [9]. For power systems, the Koopman operator has been used to control individual systems, such as synchronous generators. Also, it was used to determine zones inside a distribution system based on its phase angle. Some works have presented a theoretical approach to the Koopman-based distributed control problem. However, it has not been applied to distributed power systems such as MGs. The distributed nature of power systems makes implementing a centralized controller difficult, making the design of a distributed one, based on data gathered from the system, attractive. It sets a particular problem due to the coupled nature of power systems; i.e., a change in the voltage in one node changes the voltage in the other nodes. There is a need to develop a distributed data-based algorithm using the Koopman operator to overcome the limitations in the modeling and control of distributed power systems and MGs. It allows establishing a linear representation of the system and facilitates the use of optimization tools such as model predictive control (MPC) [10], [11]. It is desirable to fulfill optimization objectives in MGs, such as energy minimization in control signals, reference tracking, and economic dispatch. In the centralized frame, optimization processes are developed more easily than in a distributed one; however, the inherent delay in the measurements and loss of packages can limit the operation of centralized control. On the other hand, distributed optimization allows overcoming the limitations of a fixed structure within the communication system. However, several considerations have to be made to reach the optimal values. Based on Lagrange multipliers, several distributed optimization algorithms have been developed: Newton-Raphson, penalty, and ADMM [12], which are also useful for solving nonlinear optimization problems.

This thesis addresses the design of data-driven algorithms based on the Koopman operator to be applied, in particular, to MGs control. The objective of this work is designing and simulating a set of algorithms for MGs control, following decentralized and distributed approaches. The controller includes optimization processes that consider the restrictions for the control action, communication, noise, and physical constraints. Also, to design an online algorithm to identify and control the dynamics in a MG.

0.2 Motivation

Power systems are large-scale systems with a wide variety of phenomena, where the MG appears as a solution for integrating elements with heterogeneous behavior. When the MG works in islanded mode can be managed as a distributed problem, including optimization objectives. Several algorithms for the control of AC-MGs have been developed. However, they rely on the identification of a model for each agent, which most times is difficult and restricts the scalability and plug-and-play capacity of the system. The Koopman operator and data-driven Koopman-based techniques allow identifying nonlinear systems. However, these techniques are used for single-agent systems, lacking the effect introduced by interconnected agents. Several limitations appear when optimization objectives are searched in distributed systems, making the problem difficult to solve or very restricted. This work focuses on modeling and controlling MGs using a Koopman-based data-driven technique with a distributed approach that makes the optimization problem more treatable by enabling the use of distributed optimization tools.

0.3 Contribution

In this thesis, we develop a framework for the design of Koopman-Based controllers following optimization strategies. The idea is to improve the performance of controllers in MGs while dealing with optimization objectives, such as minimizing the control effort and the voltage deviation while keeping the power-sharing.

This thesis is divided into four chapters as follows:

1. The first chapter presents a literature review of MG control and modeling, focusing on the voltage regulation problem. Then, the general approach to data-driven techniques is presented with its application to control, emphasizing the Koopman operator. Some aspects of the distributed control and optimization approach are presented too.
2. The second chapter describes the MG model used for a decentralized Koopman-based voltage controller designing. The design includes an MPC controller for an inverter-based MG using a linear predictor.
3. The third chapter presents the design of the Koopman-based controller for a distributed system, following non-cooperative and cooperative approaches. The convergence for the first algorithm is proved. For the second one, it is shown how to

determine the coupling matrices for the system. As an alternative, a distributed ADMM is designed to solve the MPC problem with guarantees given by the Koopman approach.

4. The fourth chapter shows the design of a data-driven online control algorithm for the MG, including the guarantee for convergence. Then, it is shown how to deal with noisy measurements to determine the Koopman operator, and how to design a controller that includes perturbations in the MG based on data.

All the contributions make the basis for voltage control in networked systems, such as MGs, following a distributed approach and using data from the system. Also, the objective function and constraints can be set under non-ideal scenarios.

Finally, this thesis presents a general conclusion and suggests future works aiming to continue the research.

List of publications

Journal Publications

V. Toro, D. Tellez, E. Mojica-Nava, and N.Rakoto-Ravalontsalama, "Data-Driven Distributed Voltage Control for Microgrids: A Koopman-based Approach,", accepted to the International Journal of Electrical Power and Energy Systems IJEPES. Elsevier, 2022.

V. Toro, E. Mojica-Nava, and N. Rakoto-Ravalontsalama, "Multiplex centrality measurements applied to islanded microgrids," International Journal of Control, Automation and Systems, vol. 19, no. 1, pp. 449–458, 2021.

V. Toro, E. D. Baron, and E. Mojica-Nava, "Optimized hierarchical control for an ac microgrid under attack," Ingeniería, vol. 24, no. 1, p. 64–82, Jan. 2019. [Online]. Available: https://revistas.udistrital.edu.co/index.php/reving/article/view/13760

Conference Articles

V. Toro, D. Tellez-Castro, E. Mojica-Nava, and N. Rakoto-Ravalontsalama, "Distributed koopman-based control of improved swing equation," IFAC-PapersOnLine, vol. 55, no. 13, pp. 97–102, 2022.

V. Toro, E. Mojica-Nava, and N.Rakoto-Ravalontsalama, "Distributed Data-Driven Control of Transportation Networks," IFAC-PapersOnLine, 2022, accepted to the 10th IFAC Conference on Manufacturing Modelling, Management and Control 2022.

V. Toro, D. Tellez-Castro, E. Mojica-Nava, and N. Rakoto-Ravalontsalama, "Data-driven voltage secondary control for microgrids," in 2021 IEEE 5th Colombian Conference on Automatic Control (CCAC), 2021, pp. 180–185.

V. Toro, E. Mojica-Nava, and N. Rakoto-Ravalontsalama, "Stability analysis of dc microgrids with switched events," IFAC-PapersOnLine, vol. 54, no. 14, pp. 221–226, 2021, 3rd IFAC Conference on Modelling, Identification and Control of Nonlinear Systems

MICNON 2021.

V. Toro, E. Mojica-Nava, and N. Rakoto-Ravalontsalama, "Multiplex Networks Centrality Measurements for Microgrids," in IFAC-PapersOnLine, 2019, vol. 52, no. 4, pp. 24–29, doi: 10.1016/j.ifacol.2019.08.149.

Chapter 1

Microgrids and Data-Driven Techniques: State of the Art

1.1 Introduction

In this chapter, we present the general aspects of MG modeling and control, and the problem of law inertia in power systems. First, we introduce the hierarchical framework for the centralized, decentralized, and distributed approaches. Then, we show the data-driven approaches emphasizing the Koopman operator, and setting the state-of-the art of these techniques. Finally, we set the bases and the main problems regarding the control of systems based on data-driven techniques.

1.2 Low Inertia in Power Systems and MG

Inertia is a physical magnitude very desirable in power systems. In conventional power systems, rotational generators act as an energy source, an energy converter, and an energy storage [3]. In rotational generators, inertia is directly related to energy storage. However, in converter-interfaced generators, inertia is not a physical magnitude. The variations of frequency in a power system with inertia are given in the form

$$M\dot{\omega}(t) = p_s(t) + p_{ns}(t) - p_l(t) - p_j(t) \tag{1.1}$$

where M is the inertia, p_s is the power generated by synchronous machines, p_{ns} is the generation of non-synchronous generators, p_j is the power demanded, and p_l are the power losses.

In a system with no inertia, that means, without synchronous generators, there is no

relation between inertia and the frequency of the system, (1.1) becomes

$$0 = p_s(t) + p_{ns}(t) - p_l(t) - p_j(t),$$

then, the frequency can vary freely, increasing the fluctuations. Any non-synchronous generator reduces the inertia of the system [3]. This lack of inertia is compensated by managing the DC side of the converters. The faster dynamics of converters compared with rotating machines introduce rapid changes in frequency and voltage in the system and possibly a coupling with the dynamics of the transmission lines. Non-synchronous sources also provide faster primary frequency control. Power systems with high penetration of renewables present low inertia, which brings new challenges to frequency and voltage control. The problem is more notorious in MGs working as independent units due to their very low or complete lack of inertia. The MG concept is a potential solution to deal with those limitations and to include the increasing number of interconnected distributed generators and storage devices.

1.3 Microgrid Concept and Framework

The MG is a set of distributed electrical generators, storage systems, and loads connected through a common electrical bus, and limited over a geographic area [13]. The connection bus determines the type of MG: alternative current (AC) or direct current (DC) [14] (see Figure 1.1). A MG connected to a utility network follows the references given by the network, which is known as a grid-following configuration. When the MG operates in islanded mode, the reference values of voltage and frequency must be set for the MG itself; this is known as grid-forming configuration [15].

A three-level hierarchical framework was proposed for the MG's control [16]. At the first level, the local voltage and current controllers, and droop control act when sources are connected in parallel [17]. At the secondary level, we find controllers that set the voltage and frequency reference values for the primary control. The third level is a global controller used for management objectives such as economic dispatch and energy flux management between the MG and the utility network or among MGs [18], as it is shown in Figure 1.2. Primary control is often made in a decentralized mode, which only requires local information. In contrast, secondary control needs to share information among elements to compare magnitudes. Here, the control might be centralized or distributed

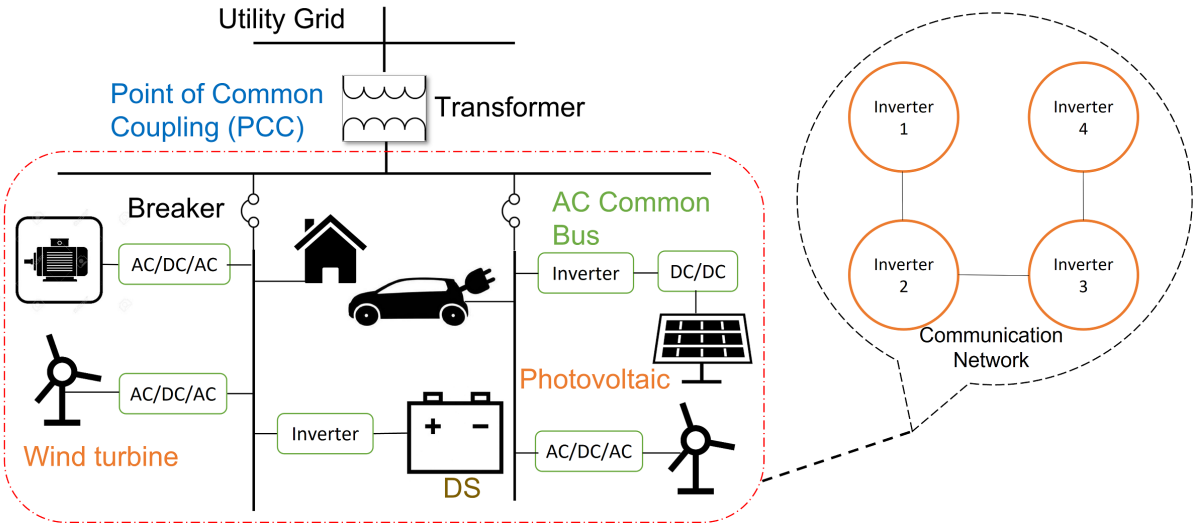

Figure 1.1 – General representation of the MG where the inverter is the key element for the interconnection.

[19].

In AC MGs, several magnitudes can be controlled: active power, reactive power, voltage magnitude, frequency, and phase. The nature of the MG is inherently distributed; loads and generators can be connected and disconnected at any moment. A centralized control frame, although possible, can be very prohibitive due to the cost of the communication system and the limitation of the plug-and-play capacity [19].

In this work, we consider a networked dynamical system represented by a graph, in which the dynamic of each agent can be nonlinear and have a heterogeneous behavior. The dynamics of individual agents are analyzed using a data-driven approach. However, the coupling between agents adds an effect to be considered when modeling with data.

1.4 Data-Driven Methods

Control methods rely on the correct identification of the plant model; this approach is also known as Model-Based Control (MBC). Proper identification from first principles could be demanding and sometimes prohibitive. However, the use of first principles models together with uncertainty models allows the correct design of controllers [20]. The increasing use of sensors and storage devices in industrial processes gives access to big amounts of data, which can be used either online or offline for control design, predic-

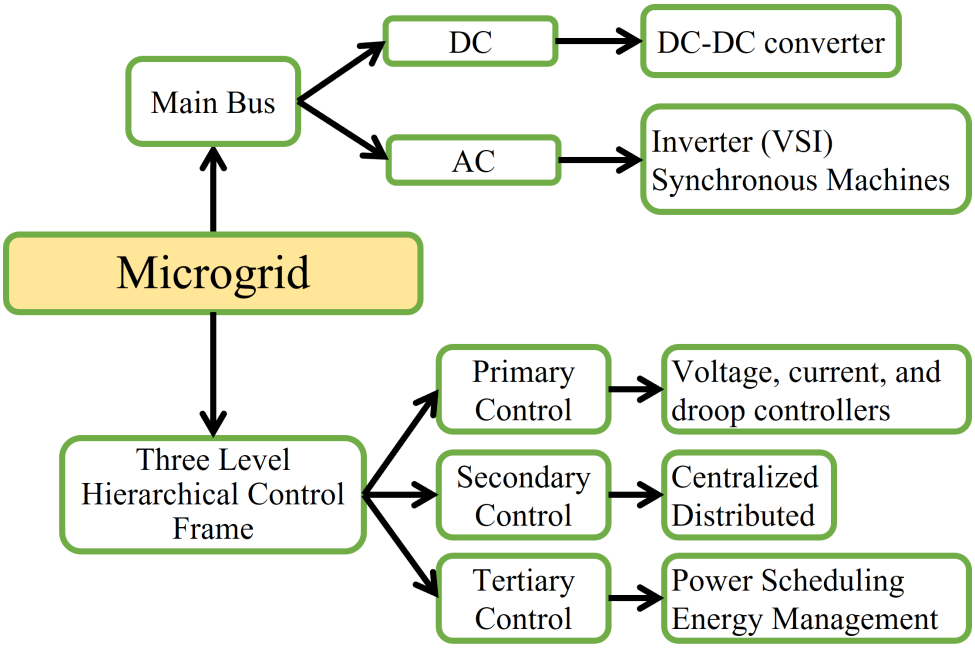

Figure 1.2 – Microgrid three-level hierarchical framework.

tion, and performance evaluation. That can be very advantageous when there is not an accurate model, making the use of data a powerful tool for control design.

Several mathematical tools have been developed for data analysis, some of them used to identify and control dynamical systems. There is a general classification of control with data:

— Data Based Control (DBC)
— Data Driven Control (DDC)

The main difference between DBC and DDC is that the former works in an open loop, while the second one works in a closed loop to generate control actions [20]. DDC can also be classified depending on how the control action is generated if data is gathered and action is then decided. DDC is online when batches of data are used to design the controller, and the control is offline. A hybrid mode is also possible. On the other hand, the controller can work with a pre-establish structure or a completely black box unknown model.

DDC Definition: Based on the definition given by [20], DDC gathers all theories and methods that use data from the system or knowledge from the data processing. DDC does not use explicit information from the mathematical model to design controllers

that work either online or offline and guarantee stability, convergence, and robustness mathematically.

Remark The restriction of explicit information from the mathematical model is broken in many cases by using a very detailed mathematical model (i.e., partial differential equations) to generate data used for DDC designing.

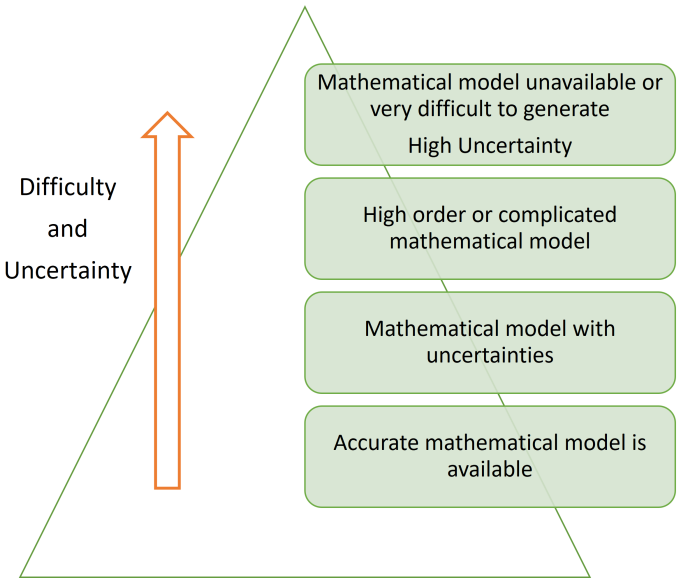

Figure 1.3 – DDC techniques appear as an alternative when it is difficult to define a precise process model or there is increasing uncertainty in the parameters.

DDC methods can be applied to several scenarios, particularly when the degree of the mathematical model is high enough to make it intractable or when the parameters uncertainty increases, making difficult the proper tuning of the proposed model as shown in Figure 1.3. Also, aiming to optimize processes, to have a high order mathematical model with high uncertainty, makes the algorithms go slowly and, in many cases, be non-convex.

DDC has some features that make it useful for system identification and control design. Also, there are a set of new challenges to be addressed as follows

- Similar to traditional control, the analysis of stability and convergence is paramount
- Online DDC overcomes the problem of time-varying parameters in contrast to MBC and off-line design
- The DDC can deal properly with nonlinear systems; for the online method, there is no difference between linear and nonlinear systems because it is only based on

data

- Non-modeled dynamics and uncertainty are involved in the data measured. Then, some robustness is involved in the DDC; however, an alternative robustness definition is necessary
- It is possible to have an exchange between DDC and MBC; both approaches can be complementary
- The different approaches of DDC should be studied and applied according to the features of each system.

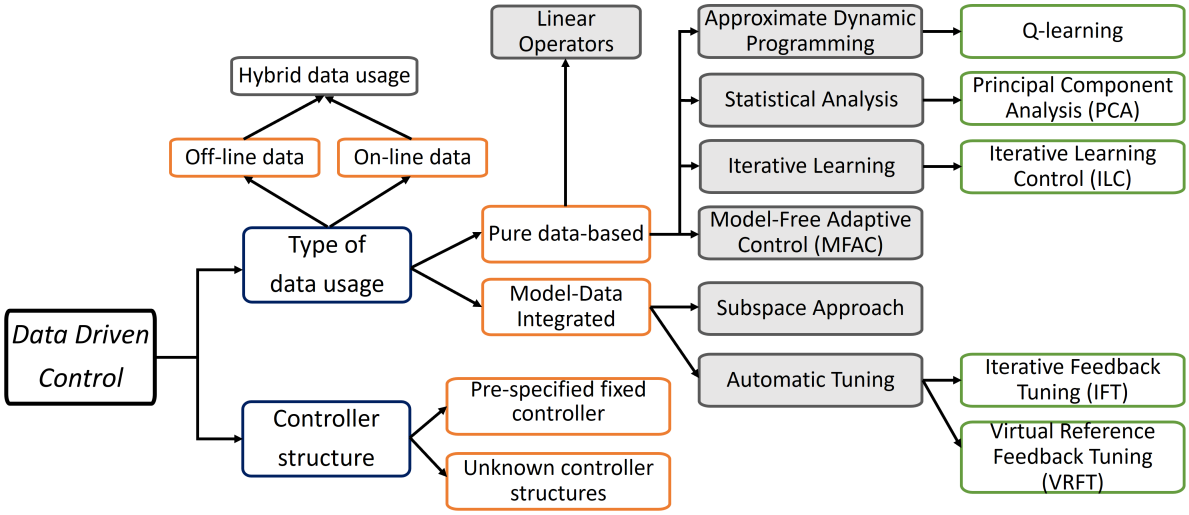

Figure 1.4 – Data-Driven methods can be classified according to different features. The main approaches are based on the way the data is used to represent the system

Several DDC algorithms have been developed recently and applied to industrial processes and control. As shown in Figure 1.4, apart from the online and off-line classification, the main variety of control approaches appears depending if data are used to find a model only based on them, or if data are used to improve an already defined model [21].

Model-Data Integrated is used to improve the performance of an existing model, particularly for tuning. IFT and VRFT are techniques that, based on a control structure, search for the appropriate parameters to tune it. IFT uses a cost function that can be optimized by gradient iterations. VRFT minimizes a performance criterion by using a model. Both techniques are mainly used in motion control systems such as robots and can be limited by their capacity to process huge amounts of data from rapidly changing and strong nonlinear systems.

Subspace Approach search to identify parameters from data coming from system measurements, it can deal with uncertainty and its results are analytically based on data arrangement as Hankel matrix and singular value decomposition (SVD). However, it needs more development to be used in nonlinear systems.

Iterative Learning Control is based only on data from the system and is training for repeating industrial control. ILC uses data to generate the control action based on the error [21].

Model-Free Adaptive Control can be applied to linear and non-linear systems; MFAC makes a dynamic linearization using a pseudo-partial derivative (PPD) using input-output measurements [22]. This approach has some limitations for highly complex processes, and it has been successfully implemented in motor control, industrial processes, and economic prediction [22].

Principal Component Analysis (PCA) PCA projects data to a lower dimensional frame with the principal components of a system. PCA minimizes the mean squared distance between data points by subtracting the mean and setting the variance to unity. Then, SVD is performed over the data. The principal components are orthogonal to each other. PCA allows using either direct data from the system or data from simulations; and also managing large quantities of data. However, PCA has some limitations in dealing with nonlinear and complex systems [23].

Dynamic Programming and Reinforcement Learning Dynamic programming can be based on Markov Decision Processes (MDP), and used to solve optimization problems forward-in-time. This method works offline and requires a complete knowledge of the system transitions. Dynamic programming uses the average reward and the Bellman equation, which relates the present values of a state with their future values [24] [25]. The controller defines the best policy (action) based on reward.

Reinforcement Learning (RL) works similar to dynamic programming; however, in RL the optimal policies (actions) are determined not by knowledge of the system dynamics but by observing the results of policies and following those that are optimal. Dynamic programming assumes that transitions among states are not probabilistic but entirely known, which is restrictive for real-world applications. In RL, transitions between states are unknown, and the system learns through several iterations, enabling the option to design adaptive controllers that work online. There are several methods to perform policy and value iteration. The three most common are: exact computation, Monte Carlo Methods, and temporal difference learning [24], with Q-learning and SARSA as the most

common algorithms as shown in Figure 1.5.

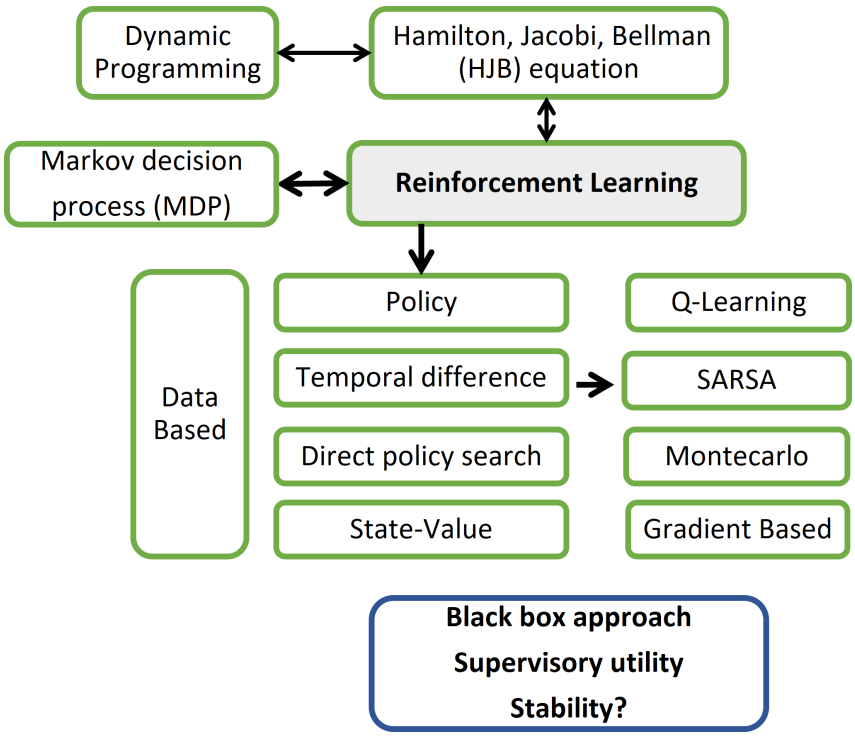

Figure 1.5 – Main aspects and algorithm for reinforcement learning.

Several results about control and stability have been obtained for RL [24], [26] including some extensions of data-based techniques to multi-agent theory such as distributed RL [27], [28]. RL has been applied to artificial intelligence and deep learning. The following features can be emphasized:

— Despite several results on stability and convergence of the approximate dynamic programming and RL, in many cases, there is not any knowledge about the system behavior, so there is a black-box model approach. Therefore, there is no information about the system structure, and how the stability is affected by changes in the system parameters.

— The training of RL-based controllers can be demanding and needs to consider several scenarios to be implemented. In many cases, some knowledge of the system is required. Also, it is necessary to define an adequate exploration method.

— A policy-based approach may not be computationally efficient; therefore, other model-free approaches can be more suitable. When the number of states is high,

the possible combinations among states and inputs grow exponentially, which is known as the course of dimensionality.

Linear Operators Approach Data-driven methods derived from linear operators are based on Koopman and Perron-Frobenius operators and their connection with Dynamic Mode Decomposition (DMD). These operators offer an interesting set of tools for nonlinear systems analysis, such as the possibility of determining stability and the domain of attraction by finding associated eigenvalues. A data approach allows for finding an approximated operator representation based on measurements useful for several applications. In the following subsections, we define the Koopman operator and its data-driven approach.

1.4.1 Koopman Operator

Koopman operator is a linear operator that allows for predicting the future state of a dynamic system based on the analysis of observables [5]. The operator extracts spatial-temporal information from the system variables and represents it as a linear one but of infinite dimension, making it very suitable for nonlinear systems analysis and control with uncertainty [29]. Koopman operator has been successfully used in different areas, such as power system control [6], [8], [30], fluid dynamics [31], video processing, neuroscience, and economy. And analysis tools such as systems with noisy data [32], control of partial differential equations (PDEs) [33], data-driven dimension reduction [34].

Suppose a dynamical system with vector field $v : X \to \mathbb{R}$ defined by a differential equation of the form

$$\frac{d}{dt}x = f(x), \tag{1.2}$$

where f is a nonlinear function.

The set of solutions or trajectories for S^t or flow $S^t : X \to X$ (1.2) with initial condition $x_0 \in X$ is given by

$$S^t(x_0) = x_0 + \int_{t'=0}^{t} f(x(t'))dt'$$

Observables: the function space $\mathcal{F}$ of observables $g(x)$ with $g : X \to \mathbb{C}$ is a set of measurement from the system.

Koopman operator [5] associated to S^t, also known as the composition operator,

is a family of linear operators of the form:

$$\mathcal{K}^t g(x) = (g \circ S_t)(x). \tag{1.3}$$

As shown in (1.3), the Koopman operator acts over the set of observables. Then, the choice of the function space $\mathcal{F}$ determines the properties of $\mathcal{K}^t$.

Square integrable functions in X with respect to a measure μ can be a good selection for $\mathcal{F}$ [29] as follows

$$\mathcal{F} = L^2(X, \mathbb{C}, \mu) := \int_X |g(x)|^2 d\mu(x) < \infty \tag{1.4}$$

where $L^2(X, \mathbb{C}, \mu)$ has an inner product

$$\langle g_1, g_2 \rangle := \int_X g_1(x)\overline{g_2(x)} d\mu(x) \tag{1.5}$$

where the bar over $g_2(x)$ denotes the complex conjugation.

Koopman Spectral Properties Koopman operator has spectral properties by selecting the function space $\mathcal{F}$ of observables $g(x)$. An **eigenfunction** ϕ_λ with discrete map S satisfies

$$\mathcal{K}\phi_\lambda = \phi_\lambda \circ S = \lambda \phi_\lambda \tag{1.6}$$

where $\lambda \in \mathbb{C}$ is the associated **eigenvalue** to ϕ_λ.

Similarly, for a continuous-time system, eigenfunctions and eigenvalues are defined as follows

$$\mathcal{K}^t \phi_\lambda = \phi_\lambda \circ S^t = e^{\lambda t}\phi_\lambda \quad \forall t \geq 0 \tag{1.7}$$

Based on the last fundamental properties of the Koopman operator, the following decomposition is defined.

Koopman Mode Decomposition (KMD) The set of observables can be defined as a vector of measurements containing information about the system variables. The vector of observables can be written as

$$\mathbf{g}(x) = [g_1(x) \quad g_2(x) \quad \dots \quad g_n(x)]^\top \tag{1.8}$$

$$g_i(\mathbf{x}) = \sum_{j=1}^{\infty} v_{ij}\phi_j(\mathbf{x}), \tag{1.9}$$

Then, the vector of observables is rewritten as

$$\mathbf{g}(\mathbf{x}) = \sum_{j=1}^{\infty} \phi_j(\mathbf{x})\mathbf{v}_j, \tag{1.10}$$

where $\mathbf{v}_j$ is known as the **Koopman mode** associated with the eigenfunction ϕ_j [23], [29].

Applying the Koopman operator with (1.10) we have

$$\mathcal{K}\mathbf{g}(\mathbf{x}) = \mathcal{K}\sum_{j=1}^{\infty} \phi_j(\mathbf{x})\mathbf{v}_j = \sum_{j=1}^{\infty} \mathcal{K}\phi_j(\mathbf{x})\mathbf{v}_j = \sum_{j=1}^{\infty} \lambda_j\phi_j(\mathbf{x})\mathbf{v}_j. \tag{1.11}$$

Then, a nonlinear system can be represented by an *infinite sum* of three elements: eigenvalues λ_j, eigenfunctions ϕ_j, and its modes v_j as it is represented in Figure 1.6.

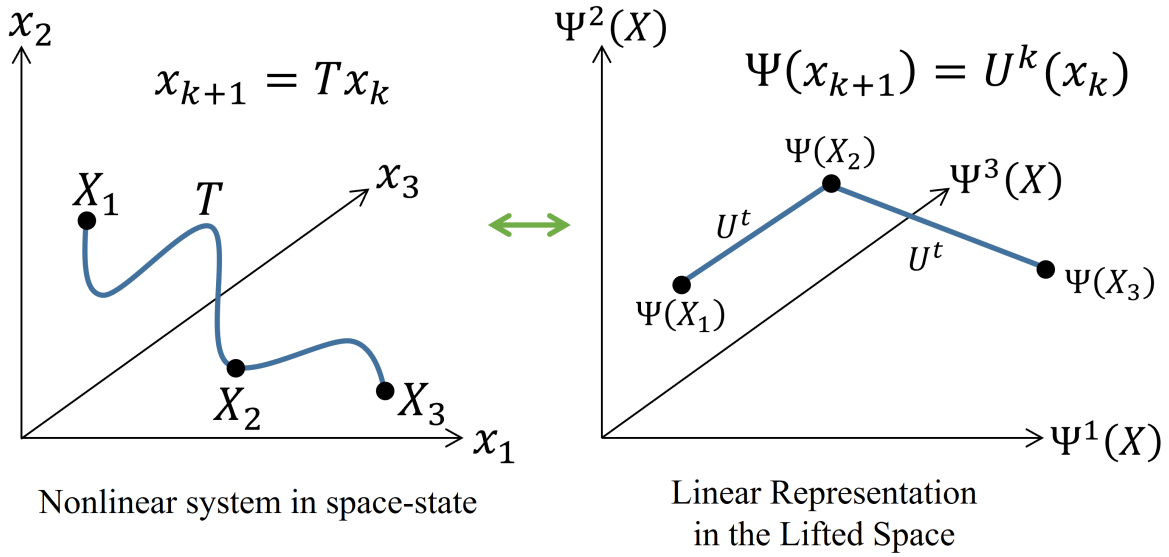

Figure 1.6 – A nonlinear system is represented in the state-space to the left, and the same system appears in the Koopman space to the right.

Several algorithms have been proposed to approximate the eigenfunctions and eigenvalues of a set of observables [29] by calculating the Koopman matrix [35], [36].

The Koopman operator has the following properties [36]:

— If the flow or map exists and is unique, then the Koopman operator exists and is unique.

— The Koopman operator is generally bounded, which makes it continuous in usual

spaces.

— If the space is finite, the Koopman operator is compact; otherwise is not compact and not self-adjoint.

— The Koopman operator is unitary for conservative systems.

1.4.2 Data-Driven Koopman Approach

Several algorithms have been developed to determine the Koopman operator of a dynamical system using data such as Galerkin Projections, Laplace averages, and DMD with its variants [37], [38]. Next, we will focus on the DMD algorithm.

Dynamic Mode Decomposition (DMD):

DMD is a data-driven method used to get a linear representation, in an equation-free form, of a nonlinear dynamical system. It was developed to analyze high-dimension dynamical systems such as fluids [39]. The most common DMD algorithm, known as exact DMD [40], works with data irregularly sampled and concatenated. It is based on Proper Orthogonal Decomposition (POD) and Singular Value Decomposition (SVD), which allows dimensional reduction. DMD decomposes a complex system in spatial-temporal structures that can be used to predict the states in a short-time window, which appears to be very practical for prediction and control [9].

As was shown by [41], there is a connection between DMD and the Koopman operator theory. In this case, DMD approximates the Koopman operator based on direct measurements from the system.

Suppose a discrete-time linear dynamic system of the form:

$$x_{k+1} = Ax_k$$

where the idea is to determine the matrix A only based on data gathered from the system. First, using N measurements of the system, two vectors of $N-1$ are created as follows

$$X_1^{N-1} = \begin{bmatrix} x_1(t_1) & x_2(t_2) & \cdots & x_{N-1}(t_{N-1}) \end{bmatrix} \tag{1.12}$$

$$X_2^{N} = \begin{bmatrix} x_2(t_2) & x_3(t_3) & \cdots & x_{N}(t_{N}) \end{bmatrix} \tag{1.13}$$

where X_2^N is the matrix of shifted measurements, and $\Delta t = t_2 - t_1$ is the sampling time that can be uniform.

Then, the SVD of data-set X_1^{N-1} is given by

$$X_1^{N-1} \approx U\Sigma W^\top,$$

where the value of X_1^{N-1} is approximated by reducing its rank r and the number of singular values, with $U \in \mathbb{C}^{n\times r}$, $\Sigma \in \mathbb{C}^{r\times r}$, and $V \in \mathbb{C}^{m\times r}$.

The approximated matrix A after the rank reduction is denoted by $\tilde{A}$, and is given by

$$\tilde{A} = U^T X_2^N W \Sigma^{-1}.$$

$\tilde{A}$ defines a linear approximation of the dynamical system in reduced coordinates of the form:

$$\tilde{x}_{k+1} = \tilde{A}\tilde{x}_k.$$

The eigenvalues and eigenvectors of the reduced matrix $\tilde{A}$ are given by

$$\tilde{A}V = V\lambda$$

where the columns of λ V correspond to the eigenvectors and the values of the diagonal matrix λ correspond to the eigenvalues of $\tilde{A}$. Finally, the DMD modes are given by

$$\Phi = X_2^N W \Sigma^{-1} V$$

DMD allows to find the principal modes of a dynamic system, what it is useful to reduce the dimension of the system.

Several extensions of DMD have been proposed, such as multi-resolution DMD, DMD with control, and extended DMD (EDMD) [35], [42], [38]. In particular, DMD has been applied to a wide variety of problems, such as fluid dynamics, epidemiology, trading [43], video processing [44], and power systems [45] among others.

Extended Dynamic Mode Decomposition (EDMD):

EDMD is an algorithm based on least-squares minimization to determine the Koopman operator of a dynamical system using data gathered from the system [38]. From a series of M data samples from the system, two data sets are built of the form $y_i = F(x_i)$, where

F is a nonlinear mapping. Defining the matrix of measurements X and Y where Y is the shifted version of X

$$X = \begin{bmatrix} x_1 & x_2 & \cdots & x_M \end{bmatrix} \qquad Y = \begin{bmatrix} y_1 & y_2 & \cdots & y_M \end{bmatrix}.$$

In EDMD, rather than in DMD, a set of N_k functions or a dictionary is defined as follows

$$D = \{\psi_1, \quad \psi_2, \ldots, \psi_{Nk}\},$$

where $\psi_i \in \mathcal{F}$ with span $\mathcal{F}_D \subset \mathcal{F}$. Then, it is defined the vector of value functions $\mathbf{\Psi}(x) : \mathcal{M} \to \mathbb{C}^{1\times N_k}$ where the measurements from the systems are evaluated at each dictionary function as

$$\mathbf{\Psi} = [\psi_1(x) \quad \psi_2(x), \ldots, \psi_{Nk}(x)],$$

the functions of the dictionary are selected for each particular problem, with the only condition that the dictionary should be rich enough to approximate the dynamic system.

The next two matrices are built from the two sets of measurements evaluated on the dictionary

$$G = \frac{1}{M}\sum_{i=1}^{M} \Psi(x_i)^\top \Psi(x_i), \qquad A = \frac{1}{M}\sum_{i=1}^{M} \Psi(x_i)^\top \Psi(y_i).$$

Then, the next minimization problem is set up

$$K = \arg\min_{\tilde{K}} ||A - \tilde{K}G||_F \tag{1.14}$$

where $||\cdot||_F$ is the Frobenius norm.

Finally, the finite-dimensional approximation of the Koopman operator matrix is calculated as

$$K = G^\dagger A, \tag{1.15}$$

where $\dagger$ denotes the pseudo-inverse, $\top$ is the transpose, and $G, A, K \in \mathbb{C}^{N_k\times N_k}$.

EDMD with control The nonlinear system identification by using measurements requires defining a finite-dimensional matrix, which also includes the effect of inputs on the system. A continuous-time **affine** system with full- state measurements is described by

$$\frac{d}{dt}x(t) = f(x) + \sum_{i=1}^{q} b_i(x)u_i, \tag{1.16}$$

where $f(x) : \mathbb{R}^n \to \mathbb{R}^n$ is the nonlinear continuously differentiable dynamic, $u \in \mathbb{R}^q$ is the set of inputs, and $b_i(x)$ is a vector field acting on the state-space. Applying the Koopman operator to (1.16) can be rewritten as

$$\frac{d}{dt}\varphi(x) = \lambda\varphi(x) + \nabla\varphi(x)\sum_{i=1}^{q} b_i(x)u_i, \tag{1.17}$$

where φ represents the set of basis functions. Then, there is a term related to the unforced system and the other to the control input, which is linear in φ [46].

Assuming a linear system of the form

$$\frac{d}{dt}x(t) = f(x) + \mathbf{Bu}, \tag{1.18}$$

where the control matrix $\mathbf{B} \in \mathbb{R}^{n\times q}$. System (1.17) is rewritten as

$$\frac{d}{dt}\varphi(x) = \lambda\varphi(x) + \nabla\varphi(x)\mathbf{Bu}. \tag{1.19}$$

If we have a nonlinear **non-affine** dynamic system of the form

$$\frac{d}{dt}x(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}), \tag{1.20}$$

where $\mathbf{f}(\mathbf{x}; \mathbf{u}) : \mathbb{R}^n \times \mathbb{R}^q \to \mathbb{R}^n$ is a continuously differentiable function. We can apply the Koopman operator as

$$\frac{d}{dt}\varphi(\mathbf{x}, \mathbf{u}) = \lambda\varphi(\mathbf{x}, \mathbf{u}) + \nabla_{\mathbf{u}}\varphi(\mathbf{x}, \mathbf{u}) \cdot \dot{\mathbf{u}}, \tag{1.21}$$

where $\dot{\mathbf{u}}$ correspond to the input multiplied by a factor given by $\nabla_u\varphi(x, u)$, plus the response given by the unforced system [46].

Linear Predictor and Control Koopman's representation allows state prediction in nonlinear systems. The dynamics can be predicted by a linear representation in the lifted space of a higher dimension using DMD or EDMD to generate an approximate representation of a finite dimension. The eigenfunctions of the Koopman operator define linear coordinates useful for designing linear predictors for nonlinear systems, making it

possible to predict the evolution of the observables and output of the system. A linear predictor for a system with Koopman representation given by a set of eigenfunctions $\phi_1, \ldots, \phi_N$, with associated eigenvalues $\lambda_1, \ldots, \lambda_N$, which not necessarily are different. The linear predictor is given by

$$\begin{aligned} \dot{z} &= Az \\ z_0 &= \phi(x_0) \\ y &= Cz, \end{aligned}$$

where

$$A = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_N \end{bmatrix} \qquad \phi = \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_N \end{bmatrix}.$$

This linear predictor allows the use of well-known linear tools for estimation and control, reducing the computational cost for optimization processes such as MPC and simplifying the control design [46], [47], [36].

1.5 Distributed Systems and Optimization

Several real-life phenomena are modeled by using a multi-agent approach that considers particular dynamics (agents) and global behaviors given by the interactions among them. Multi-agent theory has been well-developed in the last two decades and uses graph theory for modeling and controlling dynamical systems [48], [49]. Control theory has been extended to the multi-agent system with results in optimal control, discrete control, stability, and game theory [50]. In power systems, several distributed approaches have been developed for MGs control, because of their intrinsically distributed nature [51], including distributed optimization [52].

1.5.1 Graph Theory and Consensus

A graph is defined by a set $\mathcal{G} = (V, E, W)$, where $V = \{v_1, \quad v_2, \quad \ldots \quad v_n\}$ is the set of nodes or vertex, and E is the set of edges or links defined by $E_{ij} = (v_i, v_j)$, also, in some cases the edges can have a weight assigned of the form w_{ij} [48]. In graph theory, the set of elements with local dynamics that share information about their states are named *agents*, and the communication among those agents is known as *links*. Links can represent just

communication among nodes or can represent physical connections, such as current in a circuit or flux in a water network. If there is communication between two nodes in both directions, it is said that the graph is *directed*, otherwise it is *undirected*. When the agents dynamics are identical, the agents are considered *homogeneous*; otherwise, the agents are *heterogeneous*, which is particularly useful for considering more realistic problems.

Some practical definitions of graphs are given next:

Degree Matrix The degree matrix D is a diagonal matrix containing the degree of each node; that is, the number of connections that the node has.

Adjacency Matrix The adjacency matrix A of a graph is a symmetric matrix of $n \times n$ where n is the number of nodes in the graph, defined by

$$A(\mathcal{G}) = \begin{cases} 1 & \text{if } (v_i, v_j) \in E, \\ 0 & \text{if Otherwise} \end{cases} \tag{1.22}$$

Laplacian Matrix The graph Laplacian matrix $\mathcal{L}(\mathcal{G})$ of an undirected graph is given by

$$\mathcal{L}(\mathcal{G}) = D(\mathcal{G}) - A(\mathcal{G}).$$

This matrix contains most of the relevant information about the graph, such as its connectivity. Also, the Laplacian matrix analysis determines the spectral properties of the graph. The sum of the rows or columns of the Laplacian matrix is zero; then it has a zero eigenvalue $\lambda_1 = 0$ and associated eigenvector $\mathbf{1} = [1 \quad 1 \ldots \quad 1]^\top$.

Continuous-Time Consensus The basic idea of consensus implies that several interactive agents achieve a common goal, then in dynamic systems that implies that agents reach a common state. The continuous-time consensus equation, for a set of N agents with a simple integrator dynamic of the form $\dot{x}_i = u_i$, can be written as

$$\dot{x}_i(t) = \sum_{j \in \mathcal{N}_i}^{N} (x_j(t) - x_i(t)) \quad x_i(0) = z_i \in \mathbb{R}, \tag{1.23}$$

where $\mathcal{N}_i$ represents the set of neighbors of the i^{th} agent.

Consensus (1.23) can be written in the matrix form as

$$\dot{x} = -\mathcal{L}x. \tag{1.24}$$

Consensus (1.23) and (1.24) is known as the average consensus. The equilibrium of the system $x^* = [\alpha \quad \alpha \dots \quad \alpha]^\top$, where α is the average of the initial conditions, is given by $\alpha = \frac{1}{n}\sum_i z_i$, and it is globally exponentially stable [53].

Continuous-Time Tracking The continuous-time tracking equation is given by

$$\dot{x}_i(t) = \sum_{j \in N_i}^{N} (x_j(t) - x_i(t)) + b_j(x_i(t) - x_{ref}), \quad x_i(0) = z_i \in \mathbb{R} \tag{1.25}$$

where x_{ref} denotes the reference value to be tracked.

This result has been extended and enriched to solve several problems, such as discrete-time consensus, consensus with delayed information, and consensus with packet loss.

1.5.2 Distributed Optimization

Several problems for control of dynamical systems are solved, including optimization objectives. In networked systems, the optimization objectives can be solved in a centralized mode, which requires a complete communication system. It can be problematic when the number of agents grows, making the system susceptible to failures. An alternative to the centralized mode is a distributed approach, in which the agent shares information only with a small group of agents known as the neighborhood.

A global objective function, which is the sum of all local objective functions, should be minimized. The optimization problem for $x \in \mathbb{R}^n$ is written as

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^{N} f_i(x),$$

where f_i is the local objective function.

Most of the distributed optimization algorithms are discrete-time consensus-based, among them stand out the gradient-based algorithm given by

$$x_i(k+1) = \sum_{j=1}^{N} w_{ij}(k)x_j(k) - \alpha(k)s_i(k),$$

where $x_i(k)$ is the state of the i^{th} agent at time k, w_{ij} is the edge weight between agents ij given by matrix W which is double stochastic, and $s_i(k)$ is the gradient of the local objective function $f_i(x)$ that is convex [12].

The distributed constraint optimization problem is given by

$$\begin{aligned} \min_{x \in R^n} & \sum_{i=1}^{N} f_i(x) \\ \text{s.t.} \quad & x \in \cap_{i=1}^{N} \Omega_i \\ & g(x) \leq 0_m \\ & h(x) = 0_p. \end{aligned}$$

where $f_i : R^n \to R$ is the local convex objective function for the agent i, Ω_i is the local constraint set. $g : R^n \to R^m$ is the global inequality constraint, and $h : R^n \to R^p$ is the global equality constraint [54].

The problem of distributed optimization is enhanced when dealing with non-convex spaces; in this case, several assumptions about the reaching capacity at a finite time have to be revised [55].

Alternative Direction Multipliers Method (ADMM) The centralized ADMM applied to solve the convex optimization problem with linear restrictions is defined as follows

$$\min_x f(x) + g(y)$$

$$\text{s.t} \quad Ax + By = C,$$

with Lagrangian given by

$$\mathcal{L} = f(x) + g(y) + w(Ax + By - C) + \frac{\rho}{2}||Ax + By - C||^2.$$

Then, the ADMM algorithm is given by

$$\begin{aligned} x(k+1) &= \arg\min_x \mathcal{L}(x, y(k); w(k)) \\ y(k+1) &= \arg\min_y \mathcal{L}(x(k+1), y; w(k)) \\ w(k+1) &= w(k) + \rho(Ax(k+1) + By(k+1) - C) \end{aligned}$$

where arg min can be found by any established method, such as gradient descent or

Newton-Raphson. This algorithm will be extended to solve an optimization problem in a distributed mode.

1.6 Discussion

After presenting the main features of the MG, data-driven control and distributed optimization, we are going to review the problem complexity, and set the research lines.

1.6.1 Nonlinear Problem

Nonlinear system identification and control can be challenging, especially for high dimensional and high uncertainty systems. We chose the Koopman operator-based techniques compared with other data-driven ones because of their capability of establishing features for control, such as stability. In general, many data-driven methods suffer from what is known as the curse of dimensionality [56]. As the number of variables and inputs increases, the number of possible combinations used to define a policy in learning techniques can grow exponentially. In contrast, the Koopman approach allows reducing the system order by using DMD or by defining an appropriate dictionary in EDMD. However, in the case of interconnected systems, as the number of nodes increases, the number of basis functions increases until the problem becomes unpractical.

In power systems, the dictionary of functions can be defined by using trigonometric functions [36]. [30] uses deep neural networks to find a dictionary of functions for EDMD; however, the distributed nature of the problem is not considered in this works. Besides those limitations, the main advantage of using the Koopman representation is that it enables the use of tools for linear systems, including those such as linear MPC using Koopman-based linear predictors [57]. Also, in this case, control of a distributed system increases the complexity of the problem. The simplification of the calculation by using a linear representation allows using distributed optimization techniques as ADMM [58]. Figure 1.7 shows a global map for the control of interconnected systems, such as power systems using data-driven techniques. Notice that it is necessary to get a connection between a single agent and multi-agent control.

One desirable feature in data-driven systems is the capacity to learn or tune its systems parameters online. The problem of updating a system using its measurements lies in the quantity and the quality of the necessary data. Here, we propose an online algorithm to

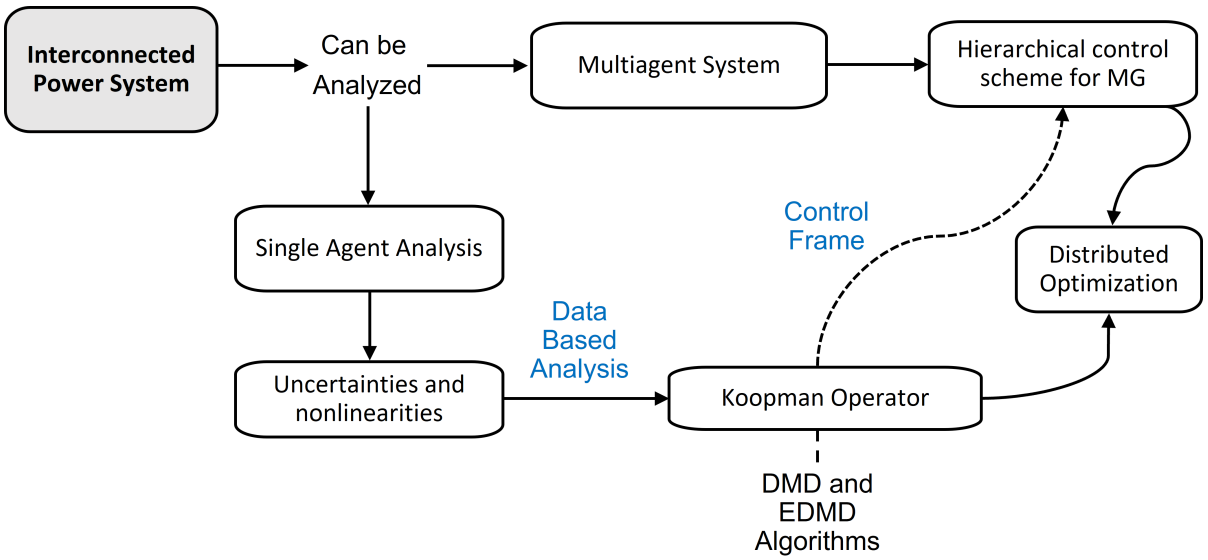

Figure 1.7 – The nonlinear power system that represents the MG is controlled using a Koopman approach. It is also used to control the system in a distributed mode.

update the representation of the system by using a series of data measured following the study by [59]. Finally, it is important to determine the Koopman operator when there is noisy-data. Some techniques to process the data are based on regularization and can be used to have a better approximation of the eigenvalues of the Koopman representation [60], [61].

1.6.2 Research Perspectives

The literature review presented in this chapter brings a series of research perspectives to be discussed:

First, the voltage control in the MG requires regulating the voltage and the reactive power flow, a nonlinear relation that also depends on the type of load and its value. Some data-driven techniques, such as those based on learning, use batches of data to determine the best policy to follow. Similarly, the Koopman approach needs data from several initial conditions to determine the trajectories of the system. Both approaches have some limitations when working online and acting if there are changes in the system structure. However, the Koopman operator allows working in a reduced dimension space, which could be practical for online control or together with learning techniques.

Second, in other data-driven approaches, the optimization tools presented are re-

stricted and do not allow the implementation of established control techniques, such as LQR or MPC, which are well-known techniques to be used with linear systems. Some properties of the Koopman operator enable the study of the convergence of those optimization algorithms, even including restrictions.

Third, the problem of controlling a distributed system is still open. Many models rely on agent modeling; however, the dynamics among interconnected agents are neglected in many cases. As mentioned, data-driven identification allows including the dynamics of interconnected agents whose correct identification should be addressed to not increase the complexity of the problem (curse of dimensionality).

Fourth, several interconnected systems search to resolve global optimization objectives. Distributed control and optimization techniques have been applied to several of these problems. Solving global optimization problems with limited time and resources is challenging in islanded MG. Thus, the Koopman approach with optimization techniques, that require less computational effort, is of high interest. It enables the design of online algorithms to update the Koopman representation using data from the system. Also, the data coming from power converters and generators are not ideal as in simulation environments; in fact, the correct treatment of these data to determine the Koopman operator for the system is paramount.

Finally, identification and control of MG based on data using the Koopman operator are still limited. Also, the nature of the variables in power systems makes them noisy and the changes in loads arbitrary. The correct identification of the system, the definitions of its features, its distributed nature, and the data treatment are relevant aspects to be considered to solve the problem studied. All these reasons justify the study of this problem that could be also applied to similar scenarios in other disciplines.

The work on this thesis continues and enhances existing research in several forms:

In chapter two, we presented a novel secondary voltage control based on the Koopman operator following a decentralized approach, which optimizes a local cost function considering restrictions for reactive power, voltage, and control input.

In chapter three, we present a distributed approach for the control and optimization problem. Based on the data-driven identification of the system using the Koopman operator, we proposed a distributed non-cooperative approach, a distributed cooperative approach, and an ADMM algorithm to regulate the voltage in MGs following optimization criteria.

In chapter four, we address the problem of data treatment to determine the Koopman

representation in a noisy environment. Also, we proposed an algorithm to control MGs using data from the system in an online approach.

1.7 Chapter Summary

In this chapter, after introducing the principal features of MG, we presented the main data-driven techniques for modeling and control of nonlinear systems. Then, we introduce the Koopman operator and the state-of-the-art of data-driven techniques to find it. After, we made a general introduction to the problem of distributed systems and optimization. We focused on the problem of identifying the dynamic of an agent in a distributed system and how the Koopman operator can set up the use of control and optimization techniques for linear systems. This model and techniques facilitate the design of controllers for MG systems that include restrictions such as voltage regulation and reactive power flow control.

In the next chapter, we present a mathematical model for the MG to be used to design a Koopman-based observer. The main problem feature includes modeling the inverter to generate data and how to approach the multiagent problem in a decentralized form.

Chapter 2

Decentralized Koopman-Based Control of MG

2.1 Introduction

Voltage and power regulation in low inertia power systems such as islanded MG requires improving the identification and control processes. In conventional power systems based on synchronous generators, the changes in frequency and voltage are dominated by the inertia of the largest generator. In contrast, in systems with high penetration of converter-based generators, inertia is very low, and transient stability depends on the reactive power and the topology of the network [62].

Despite the availability of measurements from MGs, these contain noise and uncertainty. Also, the variations over transmission lines and loads, the connections and disconnection of inverter-based generators, and changes in the parameters make the MG modelling difficult. Thus, improving the control and identification of the dynamics of a networked system is relevant. In the literature, several approaches have been studied; some of them try to model the whole MG based on linearization and fundamental principles [63]. Others use an agent approach, in conjunction with physical principles [17], facilitating the growing of the network and the plug and play capacity. Several studies and designs, based on those models, are suitable for regulating voltage and frequency in the MG while keeping some restrictions on power. Optimization techniques, such as LQR and nonlinear optimization, are applied to the control of MG. However, some difficult aspects of modeling are omitted or generalized.

In this chapter, we design a Koopman-based voltage control using a mathematical model of the inverter-based MG that allows minimizing the voltage error and the control input. This chapter considers a fixed Koopman model, and a decentralized controller for the networked system. The EDMD algorithm is used to find the Koopman approximation, while a set of basis functions is proposed. Here, there is no considered uncertainty in the

model, and the objective is to reduce the computational time in the optimization process.

2.2 Problem Description

Alternate current MG is based on inverters; the state, when it works disconnected from the utility network, is known as islanded mode, and the inverters should work in grid forming configuration generating their voltage and frequency references [64]. Then, voltage and frequency constitute the principal magnitudes for the regulation of the MG magnitudes [65].

Voltage regulation has some challenges as follows:

— Voltage varies along the transmission lines, unlike the frequency, which is a global variable

— The increasing use of nonlinear loads, such as rectifiers in electronic devices

— There is a relationship between the voltage regulation and reactive power flow

— Voltage regulation can also be affected by the network topology.

Power flow between inverters connected by transmission lines is modeled by defining the admittance values between the i^{th} and the j^{th} inverters as

$$Y^{ij} = G^{ij} + jB^{ij},$$

where G^{ij} is the conductance, and B^{ij} is the susceptance, with $j = \sqrt{-1}$.

The active and reactive power between a set of N inverters is given by the Kundur equation as

$$P^i = \sum_{j \in \mathcal{N}^i}^{N} V^i V^j |Y^{ij}| \cos(\theta^{ij} + \delta^j - \delta^i), \tag{2.1}$$

$$Q^i = \sum_{j \in \mathcal{N}^i}^{N} V^i V^j |Y^{ij}| \sin(\theta^{ij} + \delta^j - \delta^i), \tag{2.2}$$

where V^i, V^j are the magnitudes of the voltages at nodes i and j, respectively. $|Y^{ij}|$ is the magnitude of the admittance between nodes ij, $\mathcal{N}^i$ is the set of neighbors for the i^{th} inverter, θ^{ij} is the admittance angle, δ^i, δ^j are the voltage phase angles at nodes i, j, respectively.

Simplifying (2.2) by assuming a very low resistance value in the transmission lines, we get

$$Q^i = (V^i)^2 \sum_{j \in \mathcal{N}^i} |B^{ij}| - \sum_{j \in \mathcal{N}^i} V^i V^j |B^{ij}| \cos(\delta^j - \delta^i). \tag{2.3}$$

It is important to make some claims about the assumption of a low resistance transmission line

— It allows decoupling the active and reactive power by doing changes over the phase angle and the voltage magnitude, respectively

— This approximation does not always hold and might be affected by the type of loads in the system, such as nonlinear ones [66]

— It can be generated by putting an inductance of higher value at the output of each inverter or by putting a virtual impedance [67]

Reactive power expression (2.3) can be simplified more by considering almost zero the difference between phase angles

$$\delta^j - \delta^i \approx 0,$$

Reactive power (2.2) becomes [68],

$$Q^i = (V^i)^2 \sum_{j \in \mathcal{N}^i} |B^{ij}| - \sum_{j \in \mathcal{N}^i} V^i V^j |B^{ij}|. \tag{2.4}$$

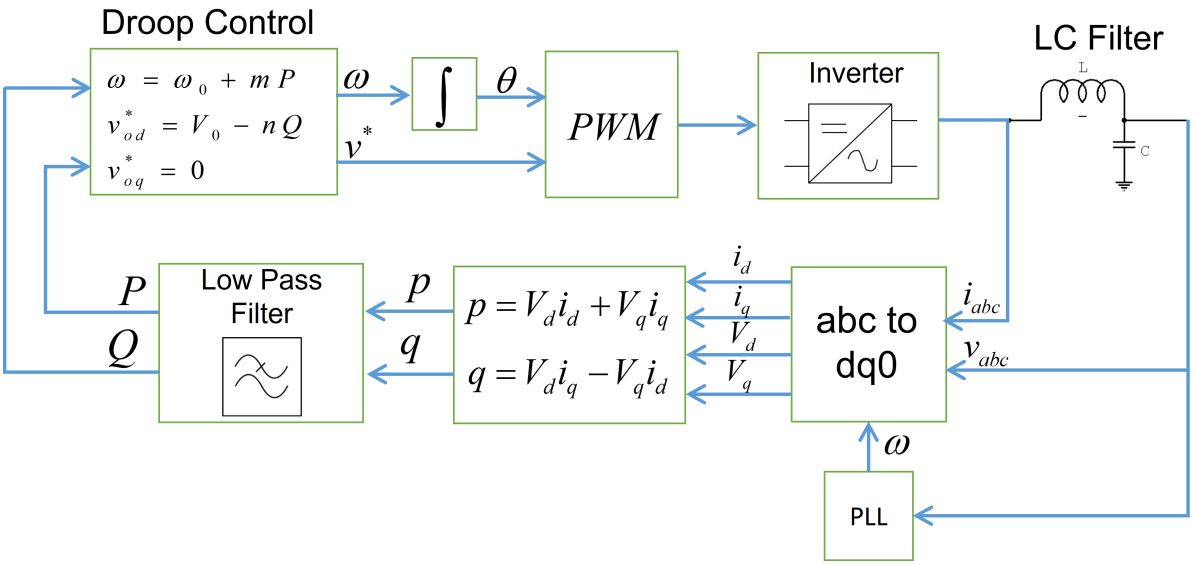

Figure 2.1 – Droop control scheme for a three-phase inverter using d-q transformation.

Based on the hierarchical control scheme presented in [69], [70] and introduced in Chapter 1, the droop control for a three-phase MG is shown in Figure 2.1. In general, droop control is a decentralized proportional control that changes the reference values of frequency and voltage in proportion to the active and reactive power measured. The main objective of droop control is to reduce the circulating currents when sources are connected in parallel and to keep the condition of each inverter supply power according to its maximum power capacity, a condition known as power-sharing [17].

Droop control uses instantaneous active and reactive power measurements to calculate the proportional action for the voltage and frequency references. The instantaneous values are filtered using a low-pass filter to avoid abrupt changes over those reference values [16]. The basic droop control equation for voltage is given by

$$V^i = V^{ref} - n_q^i \hat{Q}^i, \tag{2.5}$$

where V^i is the voltage reference after drop for the i^{th} inverter, V^{ref} is the reference voltage, n_q is the droop coefficient for reactive power, and $\hat{Q}$ is the medium reactive power that can be expressed by the relation

$$\hat{Q}^i = \frac{1}{1+\tau^i s} Q^i, \tag{2.6}$$

where Q^i is the instantaneous reactive power given by (2.4), and τ^i is the low-pass filter constant. This relation can be written as a differential equation as follows

$$\hat{Q}^i + \tau^i \dot{\hat{Q}}^i = Q^i \tag{2.7}$$

Combining (2.4), (2.5), and (2.7), the next differential equation for voltage is generated

$$\tau^i \dot{V}^i = -V^i + V^{ref,i} - n_q^i \left(Q^{L,i} + V^i V^i \sum_{i \in \mathcal{N}^i} |B^{i,j}| - \sum_{i \in \mathcal{N}^i} V^i V^j |B^{i,j}| - Q^{ref,i} \right). \tag{2.8}$$

where $Q^{L,i}$ is the power load, and $Q^{ref,i}$ is the power reference.

Secondary Control Secondary control corrects the deviations made for the droop control over the frequency and voltage reference values, as shown in Figure 2.2. This control layer moves the reference values to keep them constant and can work following a centralized or a distributed approach [16], [19]. The secondary voltage control is given by

$V^i = V^{ref} - n_q^i \hat{Q}^i + u^i$ where u^i is the secondary control term. In the first case, the controller needs measurements from all inverters to generate the action; in the second case, local controllers only need measurements from the set of neighbors. The secondary control must keep the reference values and the power-sharing condition. These two conditions are opposed, generating a trade-off between them because an improvement in voltage regulation degrades power-sharing and, in contrast, a better power-sharing deteriorates voltage regulation.

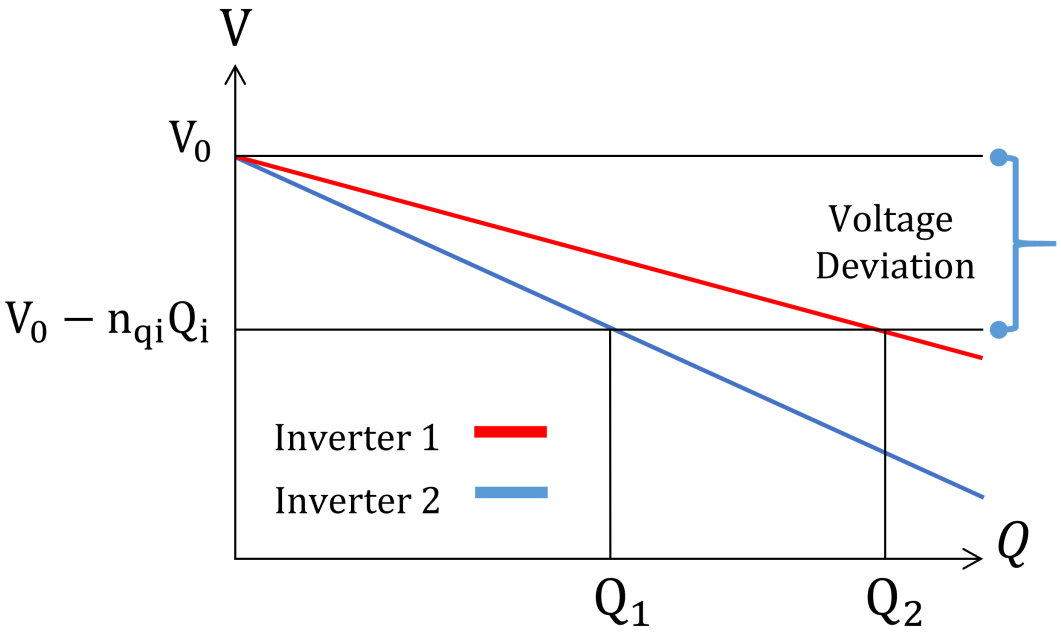

Figure 2.2 – Secondary control scheme for voltage regulation, the secondary control shifts the curve to keep the reference value.

Model Predictive Control Design Voltage, current, and droop-control have several limitations, such as their performance dependence on the control parameters, their load changes sensibility, their tuning process, and the necessity of a pulse-width modulation (PWM) stage. MPC allows overcoming some of those limitations by performing an optimization process on finite time. In inverter-based MGs, predictive control can be made local for converter-level or global for grid-level [71].

For control-level MGs, predictive control can be classified either as a continuous control set (CCS-MPC) or a finite control set (FCS-MPC). CCS-MPC generates continuous signals for the PWM converters, while FCS-MPC generates discrete-time signals. Therefore,

using PWM signals is not necessary [71].

The inverter dynamic is represented as follows

$$x(k+1) = f(\mathbf{x}(k), \mathbf{u}(k)), \tag{2.9}$$

where $\mathbf{x}(k)$ represents the set of states for the converter at time k, $u(k)$ is the control input, and f is a function representing the dynamics of the inverter.

For a converter modeled with f given by a linear function, system (2.9) is written as

$$\mathbf{x}(k+1) = A\mathbf{x}(k) + B\mathbf{u}(k), \tag{2.10}$$

where $A \in R^{n \times n}$ is the matrix of states, $B \in R^m$ is the matrix with the control inputs.

One of the control goals is to reach and keep the state variables or output of the system in their reference values. If reference values are denoted by x^* the next condition represents the control goal

$$x(k+1) = x(k) = x^*. \tag{2.11}$$

Then, the cost function $V(x, u)$ can be set

$$V(x,u) = ||x - x^*||^2_{\hat{Q}} + ||u - u^*||^2_{\hat{R}} + ||x(k+1) - x||^2_{\hat{P}} \tag{2.12}$$

where Q and R are semi-positive definite and P is positive definite [72].

Combining (2.10) and (2.11), the steady-state input can be determined as

$$x^* = (I - A)^{-1} B u.$$

Secondary Voltage Control and MPC Voltage regulation and optimization problems in MGs are presented in works, such as [65], [73]. In [65], a nonlinear relation of voltage-reactive power is used to design a distributed secondary control. However, solving the non-convex optimization problem at each sampling time requires a particular solver and might increase computational time. Also, the model is decentralized, keeping the voltage of the neighbors constant in the prediction horizon.

Secondary control should correct the voltage deviations while maintaining the power-sharing condition. A local optimization process can correct voltage deviations while reducing the control effort; this is presented as an MPC problem with restrictions. The

proposed MPC is defined as follows

$$\min_{u^i_{t+k}} \sum_{k=0}^{H_p} |V^i_{t+k} - V^{ref,i}_{t+k}||^2_{\hat{Q}} + ||u^i_{t+k}||^2_{\hat{R}} \tag{2.13}$$

s.t.

$$V^i_{k+1} = V^i_k + T\left(-V^i_k + V^{ref,i}_k - nq^i\left(Q^{L,i}_k + \sum_{i\in\mathcal{N}} V^i_k V^i_k |B^{i,j}| - \sum_{i\in\mathcal{N}} V^i_k V^j_k |B^{i,j}| - Q^{ref,i}_k\right)\right) \tag{2.14}$$

$$0.95V^{ref} \leq V^i_k \leq 1.05V^{ref} \tag{2.15}$$

where condition (2.14) is the discrete form of (2.8), and T is the sampling time. This optimization problem is solved at each sampling time k for the prediction horizon H_p. The restriction (2.14) is nonlinear and implies a non-convex problem with more computational effort and possible convergence issues.

These non-linearities can be managed by using a data-based approach. Usually, series voltage and current measurements from each inverter are available. The next section presents the Koopman operator and a data-based approach for non-linear systems representing.

2.3 Data-Driven Voltage Secondary Control for MGs

Despite model (2.8), several generalizations and approximations should be made, due to the inherent differences in time constants, tolerances, gains, impedance, and the uncertainty of each system. The interconnection among systems also introduces more difficulties for modeling the whole networked system. That gap can be fulfilled by using data measured from each inverter.

The quadratic term V_i^2 in (2.8) can be problematic because it makes the optimization problem not convex, and it implies more computational time to find the optimal input.

Assumption 1 *The system has enough available power to supply the loads connected to the system.*

Assumption 2 *All the inverters have a double-way configuration, so each source can supply and absorb power.*

Data-Generation Data could be obtained from measurements from sensors or by detailed simulations. In the second case, a detailed simulation of the MG can be made in software packages such as Simulink®or by direct simulations of (2.8) using software such as Python or Matlab®.

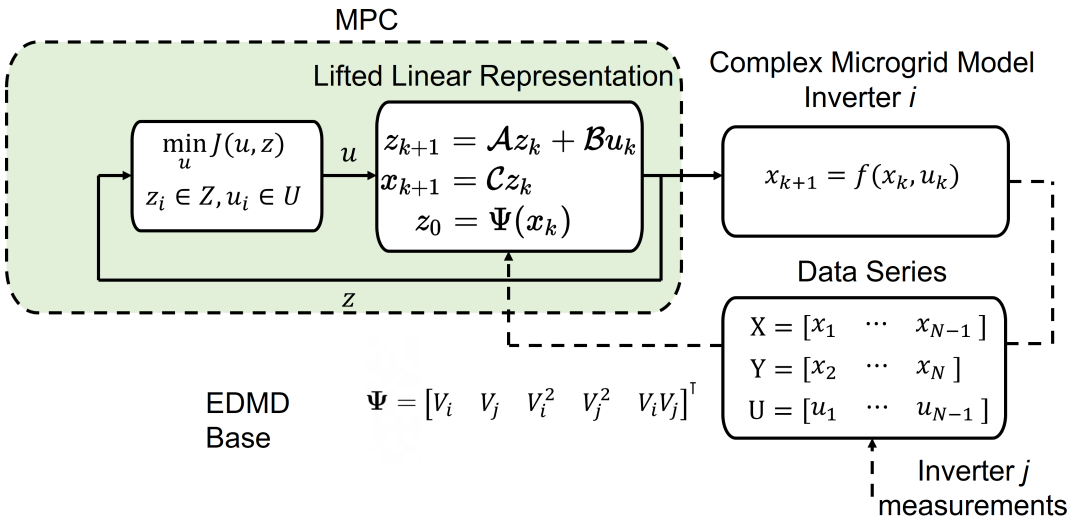

Figure 2.3 – Decentralized Koopman MPC scheme with a linear predictor using EDMD.

For this case, we used Simulink®to create a model of the MG including the droop-control for frequency and voltage using (2.14). The initial conditions for voltage at the i^{th} inverter vary randomly from $[0.5V^{ref}, 1.5V^{ref}]$, loads are considered as perturbations to the system if they are inside the maximum reactive power value, the input u^i varies randomly each 1.7s between $[-2, 2]$, and voltages j^{th} as v^i. The voltage at the output of each inverter is sampled at $T_s = 10$ms together with the input at the i^{th} inverter, as shown in Figure 2.4.

Thus, using the EDMD algorithm with the base defined by

$$\mathbf{\Psi} = [V_i \quad V_j \quad V_i^2 \quad V_j^2 \quad V_iV_j]^\top,$$

with the initial condition given by

$$\mathbf{\Psi} = [V_i(0) \quad V_j(0) \quad V_i^2(0) \quad V_j^2(0) \quad V_i(0)V_j(0)]^\top,$$

the matrices $\mathcal{A}_i$, $\mathcal{B}_i$, and $\mathcal{C}_i$ are determined by EDMD defining the set of basis functions or dictionary as shown in Figure 2.3. The matrix $\mathcal{A}$ has N eigenvalues that are plotted in the

unit circle as shown in Figure 2.5, there is one eigenvalue 1, and four smaller eigenvalues.

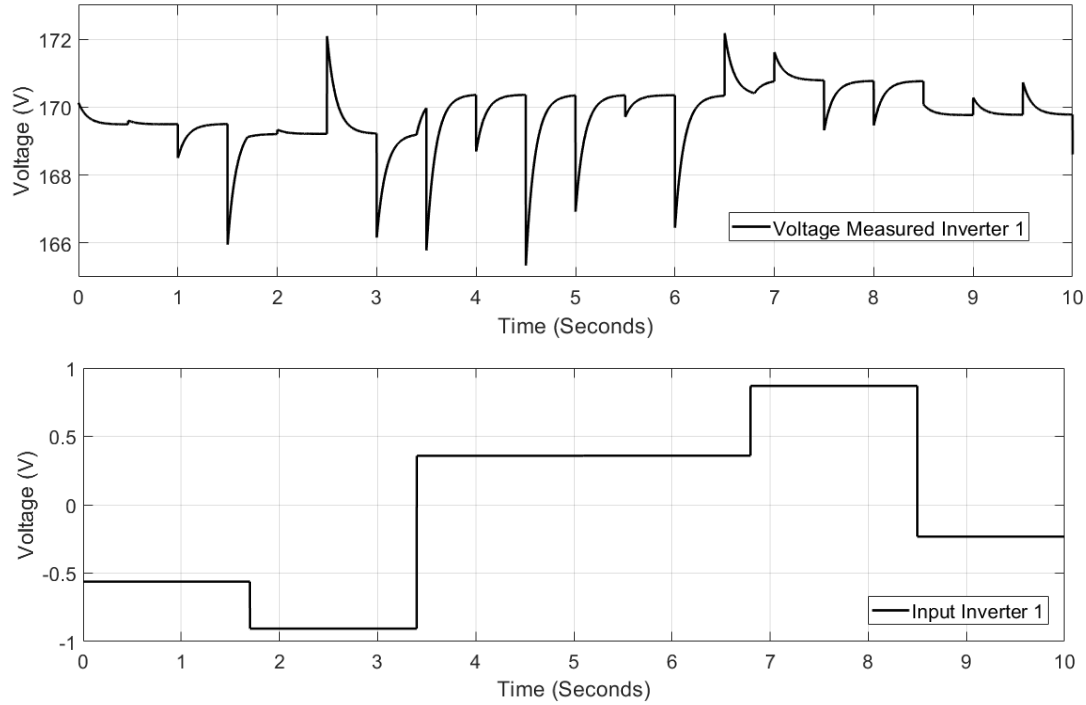

Figure 2.4 – Data Generated from simulation to obtain the Koopman representation of inverter 1.

The comparison between the Koopman approximation for each inverter-based generator and the voltage measured at each inverter's output is shown in Figure 2.6. Data gathered from each source is split into two sets, one to obtain the Koopman matrices $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$, and the other one to verify the validity of the approximation. The approximation is good enough in the first steps, which makes it suitable for MPC.

The comparison between the real measurements and the ones generated by using the Koopman approximation with matrices $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ is made by plotting the error as shown in Figure 2.7. The error is inferior to 1 percent in the first 0.5 seconds, which is suitable for the MPC design.

Assumption 3 *It is assumed that the system with matrices $\mathcal{A}$ and $\mathcal{B}$ is controllable, and the system with matrices $\mathcal{A}$ and $\mathcal{C}$ is observable.*

The last assumption is checked directly from the matrices obtained with EDMD and depends on the selection of the dictionary of functions. The MPC presented in 2.13

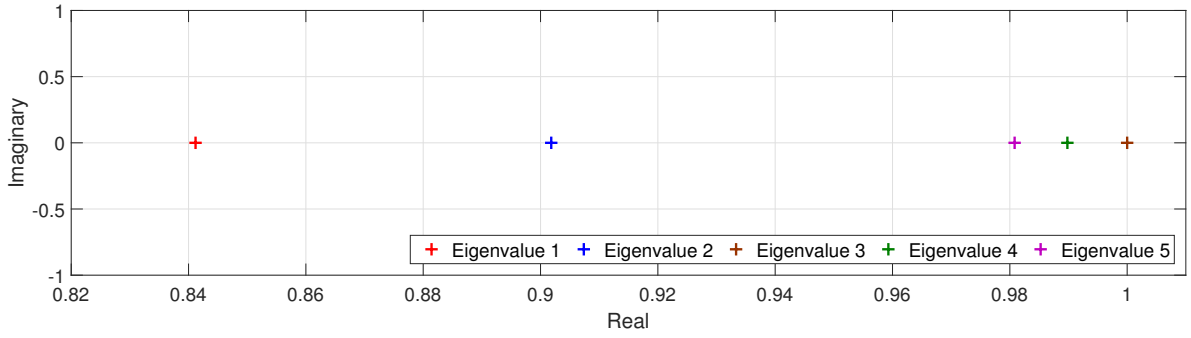

Figure 2.5 – Zoom of the eigenvalues of matrix $\mathcal{A}_1$. All eigenvalues are real, with one in 1 and the others being smaller.

is rewritten using the linear representation of each inverter generated by the EDMD algorithm as follows

$$\begin{aligned} \min_{u^i_{t+k}} \sum_{k=1}^{H_p} & ||V^i_{t+k} - V^{ref,i}_{t+k}||^2_{\hat{Q}} + ||u^i_{t+k}||^2_{\hat{R}} \\ \text{s.t.} \quad & z^i_{k+1} = z^i_k + T_s(\mathcal{A} z^i_k + \mathcal{B} u^i_k) \\ & V^i_k = \mathcal{C}^i z^i_k \\ & 0.90 V^{ref} \leq V^i_k \leq 1.10 V^{ref}. \end{aligned}$$

Algorithm 1 Decentralized MPC Koopman-Based

procedure $u^i(V^i, V^j)$ ▷ Voltage measurements from other inverters
  System Initialization
  Read the value
  Set $\mathcal{A}^i$, $\mathcal{B}^i$, $\mathcal{C}^i$ ▷ Koopman matrices of each subsystem
  Set V^{ref} ▷ Voltage of reference
  Set H_p, T ▷ Prediction Horizon and Sampling Time
  Set Q, R ▷ MPC Gains
  while $k \leq H_p$ **do**
    $u^i_k \leftarrow \min_{u^i} \sum_{k=1}^{Hp} ||V^{ref} - V^i_k||^2_R + ||u^i_k||^2_Q$ ▷ Objective function
    s.t. $z^i_{k+1} \leftarrow z^i_k + T(\mathcal{A}^i z^i_k + \mathcal{B}^i u^i_k)$ ▷ Restrictions
      $0.90 V^{ref} \leq V^i \leq 1.10 V^{ref}$
      $V^i_k \leftarrow \mathcal{C}^i z^i_k$
    Solve u^i ▷ Solve by using a linear solver
    Select $u^i(1)$ ▷ Select the first optimal value from vector u

The decentralized Koopman-based algorithm is summarized in **Algorithm 1**.

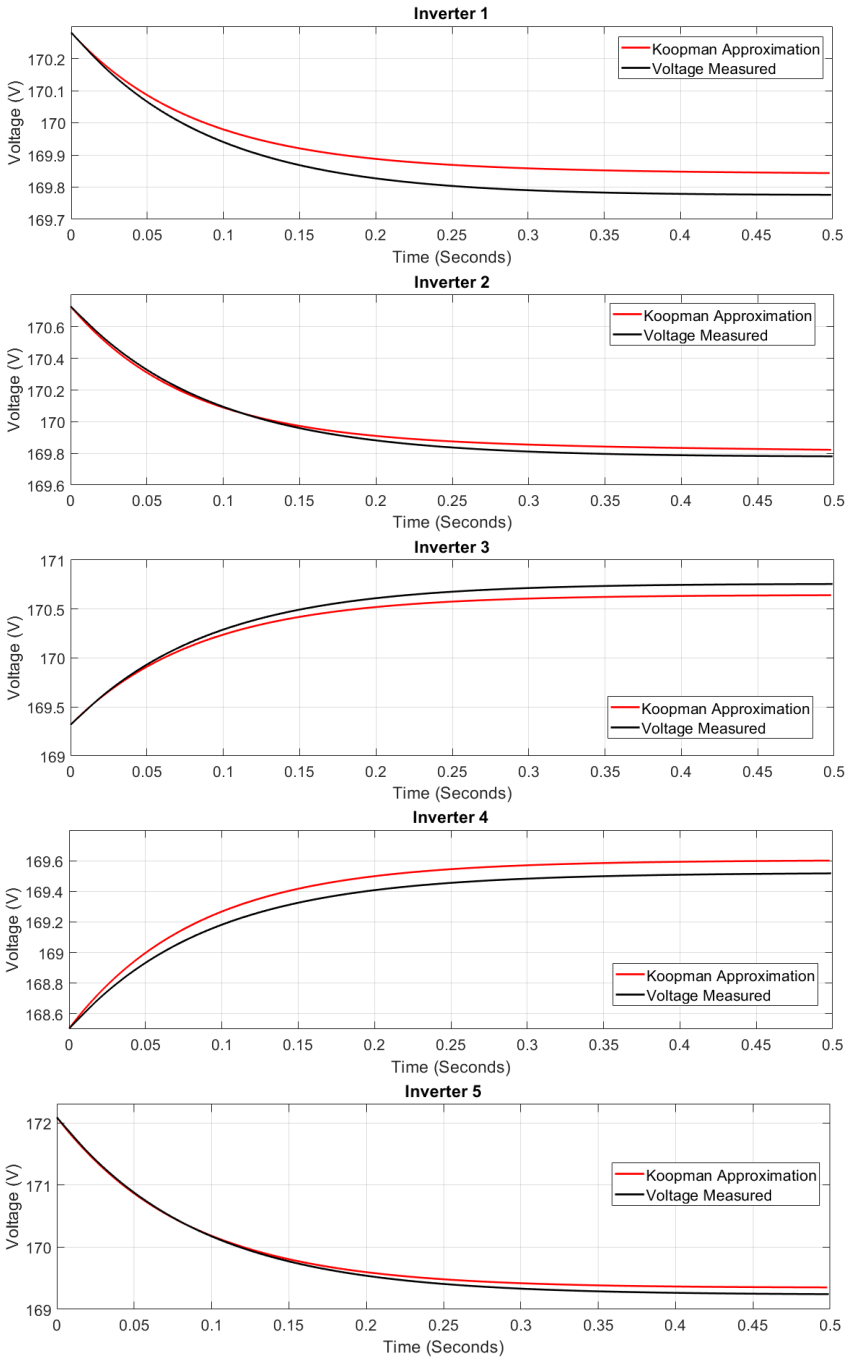

Figure 2.6 – Koopman approximation for each inverter-based generator. The five figures show the comparison between the voltage measured at the output of each inverter and the signal generated by the Koopman representation.

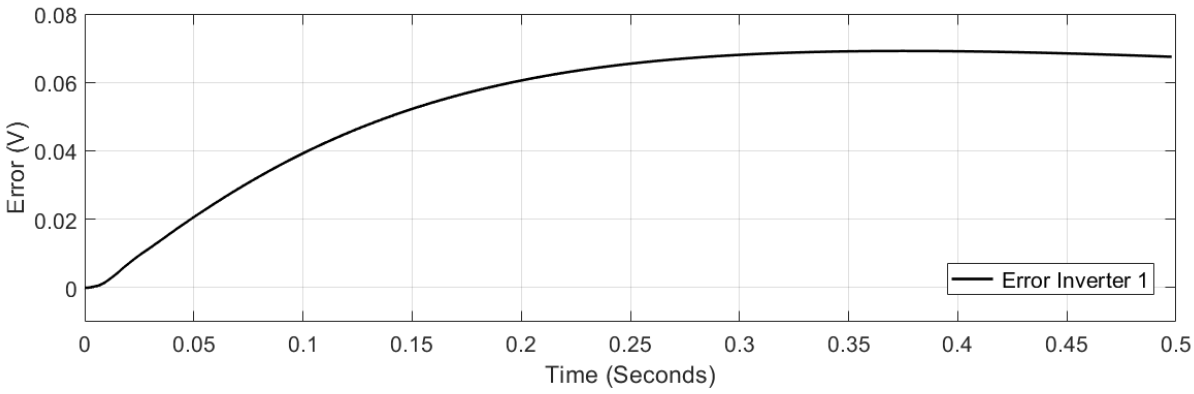

Figure 2.7 – Error between the data-measured and the Koopman approximation for inverter one.

2.4 Simulation Results

In this section, a MG is simulated in Simulink using the 14-node IEEE model [74]. Figure 2.8 shows the schematic model, which consists of five VSC inverter-based generators, 11 PQ loads, and 20 branches. Transmission lines are denoted from B1 to B20 [75]. The predictive controllers were simulated in Matlab using Yalmip [76] as the interface and Gurobi as the optimization solver, on an Intel i7-5500U processor at 2.4 GHz and 4 GB RAM.

Inverter	P^{ref} (kW)	Q^{ref} (kvar)	m_i (1×10^{-4})	n_i (1×10^{-4})	τ (s)	v^{ref}(RMS)
1	5	5	2.00	2.00	0.10	120
2	5	5	2.00	2.00	0.15	120
3	5	5	2.00	2.00	0.20	120
4	8	8	1.25	1.25	0.25	120
5	8	8	1.25	1.25	0.29	120

Table 2.1 – Microgrid parameters for the decentralized case.

The parameters of the MG's inverters are summarized in Table 2.4. Each inverter has a different time-constant τ^i, and there are two sets of reactive power nominal values. In Table 2.2 appears the inductance values for the transmission lines. The load values for active and reactive power are shown in Table 2.3. Finally, the general MPC parameters for each local controller are shown in Table 2.4. The sample time for the controller is given by T_s, and the control horizon is limited to ten steps ahead.

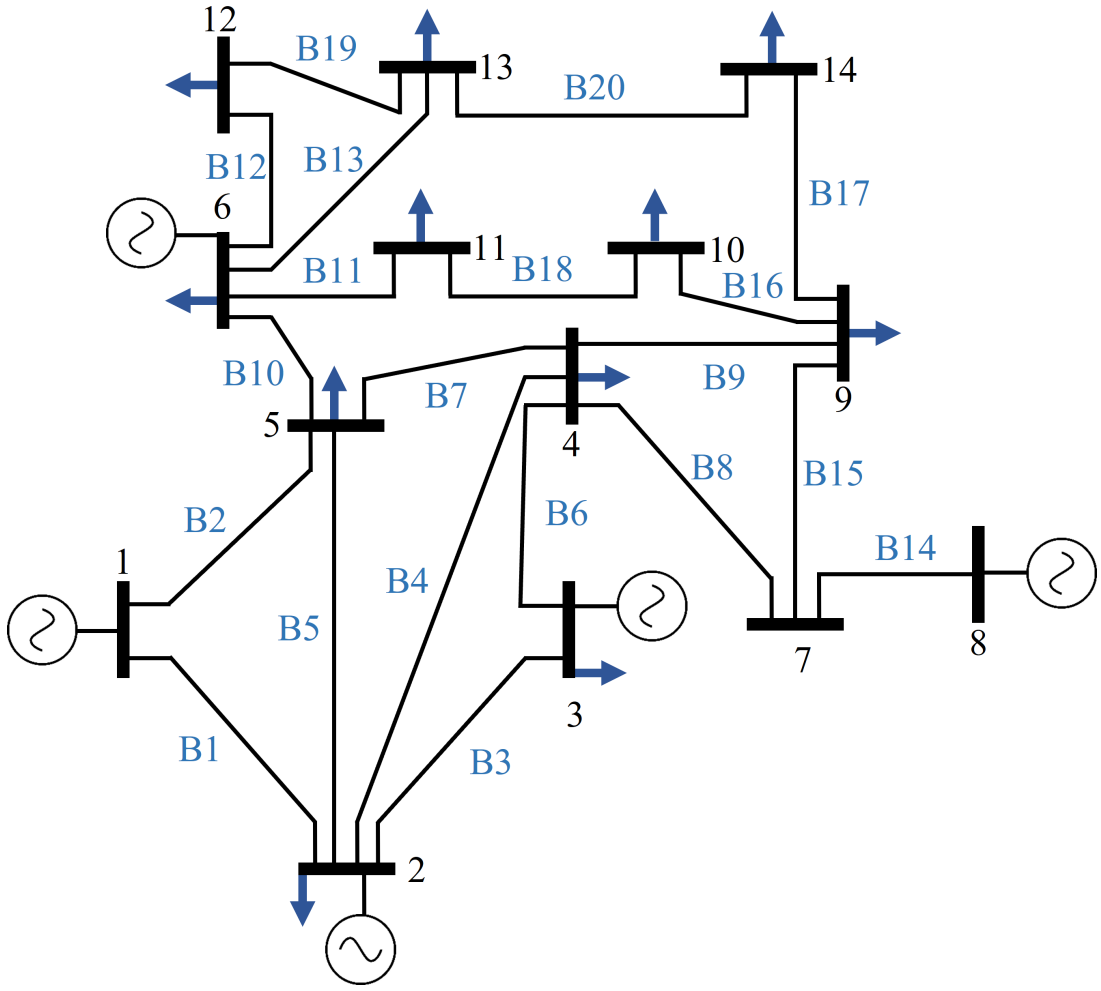

Figure 2.8 – 14-node IEEE model with five inverters and 11 loads

Line	Inductance (mH)	Line	Inductance (mH)
B1	0.83	B11	2.79
B2	3.13	B12	3.59
B3	2.78	B13	1.82
B4	2.47	B14	2.47
B5	2.44	B15	1.54
B6	2.40	B16	1.18
B7	0.59	B17	3.79
B8	2.90	B18	2.69
B9	7.80	B19	2.80
B10	3.54	B20	4.88

Table 2.2 – Transmission line values.

Node	2	3	4	5	6	9
Load (kW)	1	1	1	1	1	1
Node	10	11	12	13	14	
Load (kW)	1	1	1	1	1	

Table 2.3 – Microgrid loads.

2.4.1 Load Changing Simulation

Loads at nodes 3, 5, 6, 9, 14 change from 0 to 1kW and 1kVar at $t = 5$s.

The voltage measured at the output of each inverter-based generator is shown in Figure 2.9. Voltage variations appear first at $t = 0$s due to the initial conditions of the system, and at $t = 5$s when loads at nodes 3, 5, 6, 9, and 14 are connected. The controller regulates the voltage after almost five seconds with some little oscillations inferior to 0.05 V as shown in Figure 2.10, where the signal from inverter 5 is larger than the others to achieve the proper regulation. The reactive power measured at each converter is shown in

Table 2.4 – MPC parameters for the decentralized algorithm.

Parameter	Inverter 1	Inverter 2	Inverter 3	Inverter 4	Inverter 5
State gain Q	12	12	15	15	15
Input gain R	1				
Sampling Time T_ss	0.1s				
Voltage restriction V_i	$115V \leq V_i \leq 125V$				
Control Horizon H_p	10				

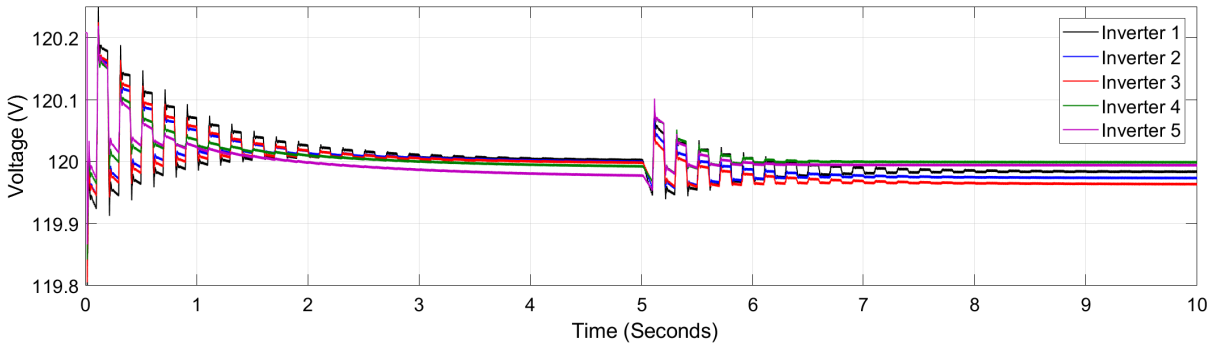

Figure 2.9 – RMS voltage measured at the output of each inverter with load changes at $t = 0$s and $t = 5$s.

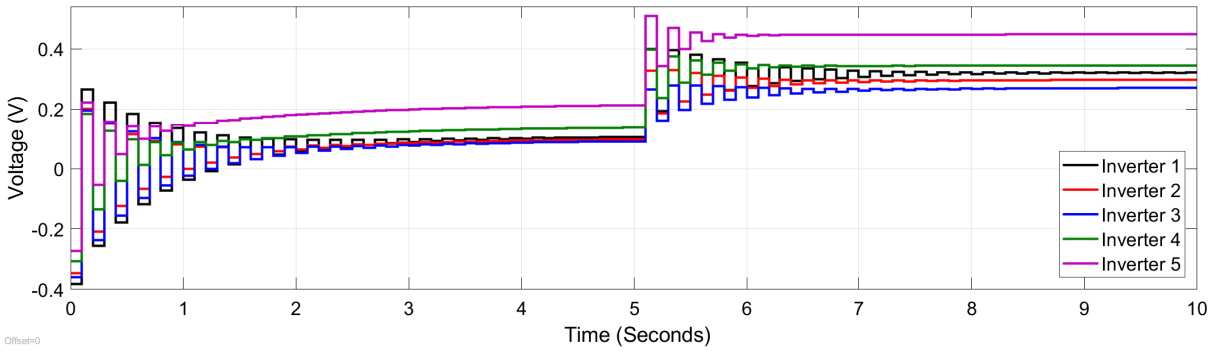

Figure 2.10 – MPC signals at each inverter, acting at $t = 0$s when the system starts. At $t = 5$s additional loads are connected, then the signals act until the reference value is reached

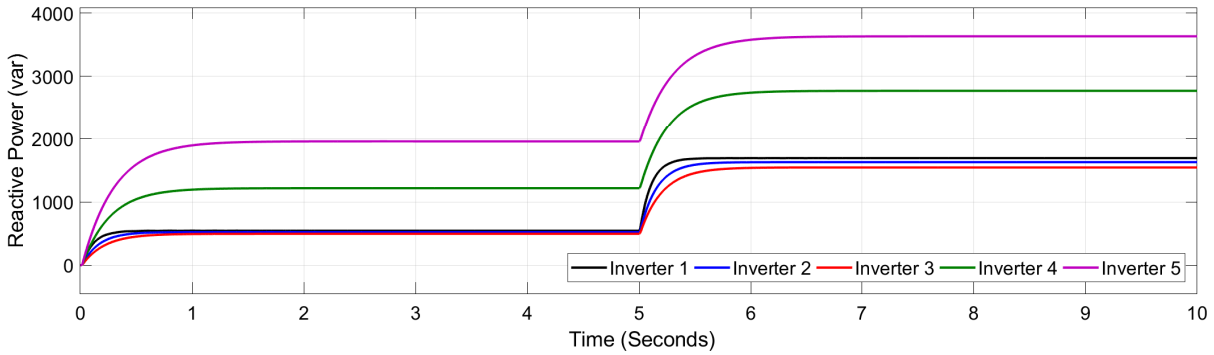

Figure 2.11 – Reactive power measured at the output of each inverter; the power changes when additional load is connected at $t = 5$s.

Figure 2.11. The values change when loads are connected at $t = 5$s. The reactive power supplied by each inverter not only depends on the maximum power rating but also on the transmission line's impedance values. The controller is capable of regulating voltage while maintaining the power-sharing condition, where the fourth and the fifth inverters supply more power than inverters one, two, and three.

2.4.2 Transmission Line Changing Simulation

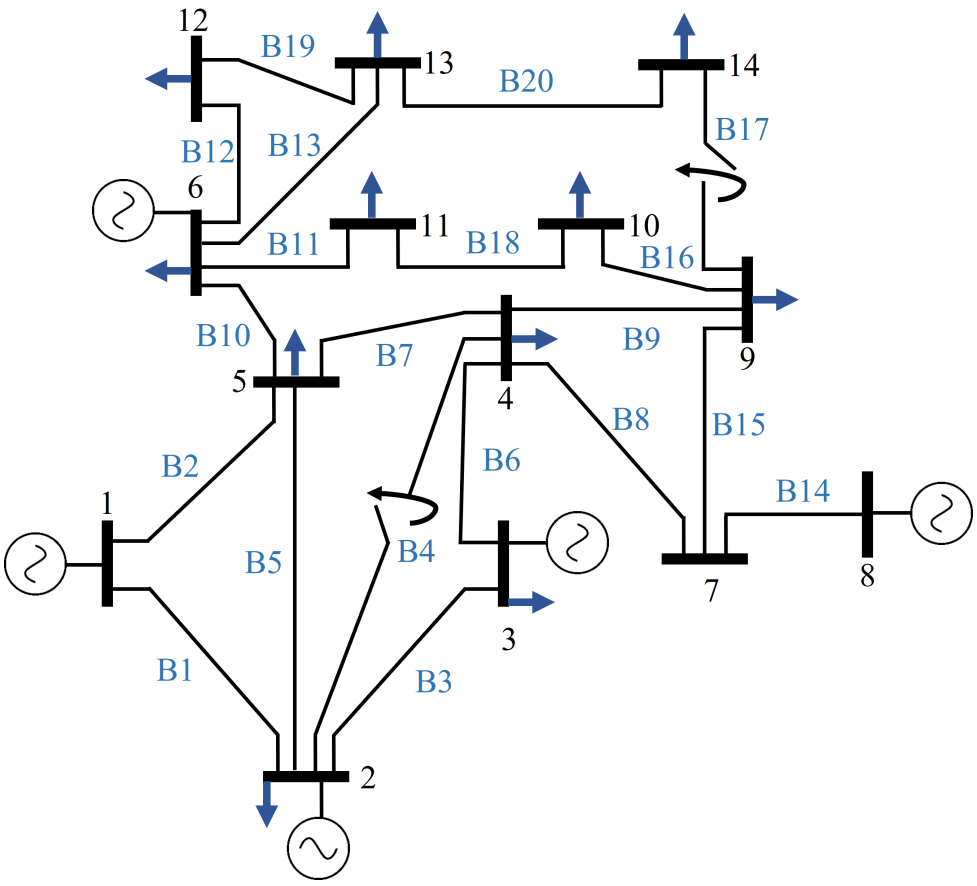

Figure 2.12 – Transmission line changes between nodes two and four, and nodes nine and fourteen.

The change in the transmission lines is simulated by connecting the second, fourth, ninth, and fourteenth nodes, as shown in Figure 2.12. At $t = 0$s the two switches are opened; then at $t = 5$s, the interrupters are closed, connecting nodes two and four and nine and fourteen through the transmission lines B4 and B17, respectively.

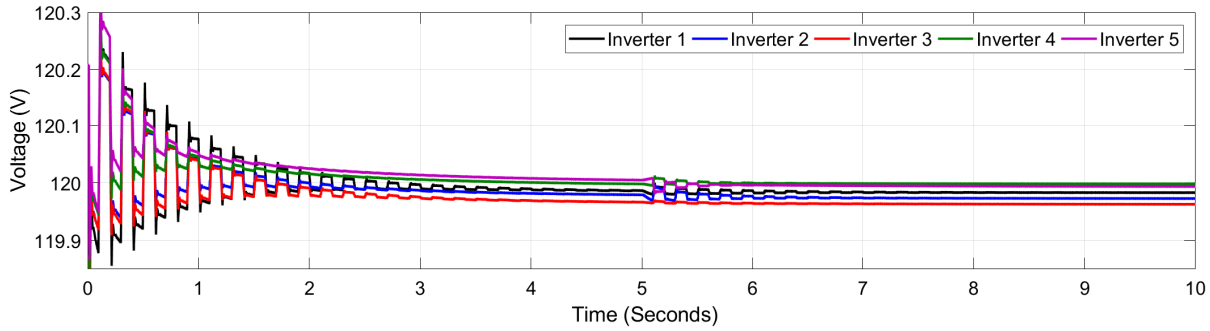

Figure 2.13 – Voltage measured at the output of each inverter. At $t = 5$s, two transmission line segments change, affecting the voltage at each inverter.

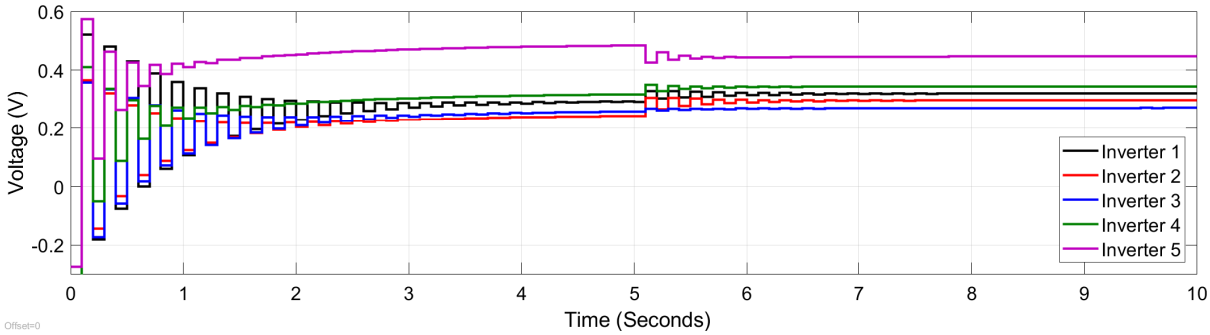

Figure 2.14 – MPC signal for each inverter. The control signals act at $t = 0$s and $t = 5$s when the transmission lines change.

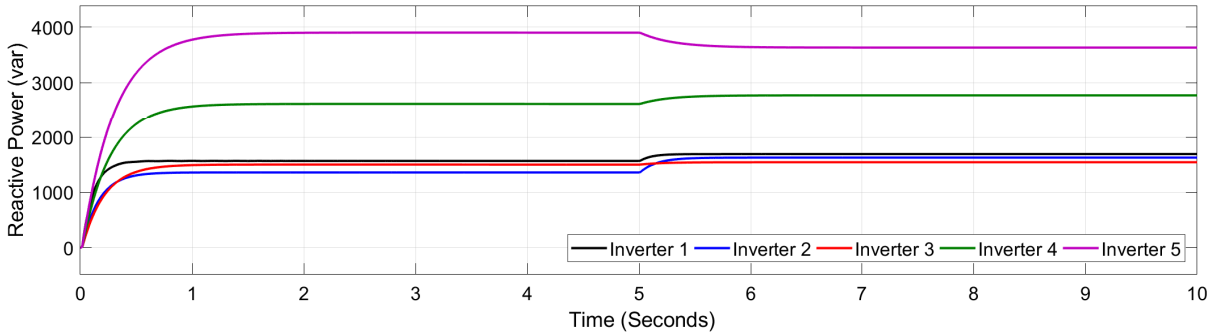

Figure 2.15 – Reactive power supplied for each inverter. Notice the change in the reactive power after the changes in the transmission lines configuration at $t = 5$s.

The voltage magnitude measured at each inverter is shown in Figure 2.13. At $t = 5$s switches are closed in transmission lines B4 and B17 connecting inverters 2 and 4, and 9 and 14, respectively. Voltages stay around the reference value, keeping the power-sharing condition. The MPC signals take less than 5 seconds to reach the desired condition, as it is shown in Figure 2.14. The reactive power supplied by each inverter is shown in Figure 2.15, after the transmission line reconfiguration at $t = 5$s, the reactive power supplied by inverter one, two, and four increase while inverter three keeps the same value than before, and it is reduced in inverter five.

2.4.3 Algorithm Comparison with Nonlinear MPC

Table 2.5 – Comparison between Koopman-based and nonlinear MPC algorithms

	Koopman-Based Algorithm	Non-linear Algorithm
Sampling Time	0.1s	0.1s
Prediction Horizon	10	10
Q	10	10
R	1	1
Solver	Fmincon	Fmincon
V_i, V_j	$[120, 121]$	$[120, 121]$
Time used per cycle	0.2098	0.5700

A comparison between the nonlinear and the Koopman approaches is presented in Table 2.4.3. Both algorithms use identical parameters and the same nonlinear solver to compare the time used per cycle for solving the optimization problem. It is clear that the Koopman-based algorithm uses less time to solve the problem, being about 171.69 percent faster than the one using the nonlinear algorithm.

The output voltages at inverter one for the proposed centralized and the nonlinear MPC are shown in (2.8). Both approaches behave similarly, approximating to the reference value at steady state. The control with the Koopman approach is faster to solve the optimization problem than the one that solves the nonlinear problem. The control signals generated under each approach are shown in Figure 2.16; each signal reaches a different value at steady-state; this can be seen in the difference on the output voltage shown in Figure 2.17.

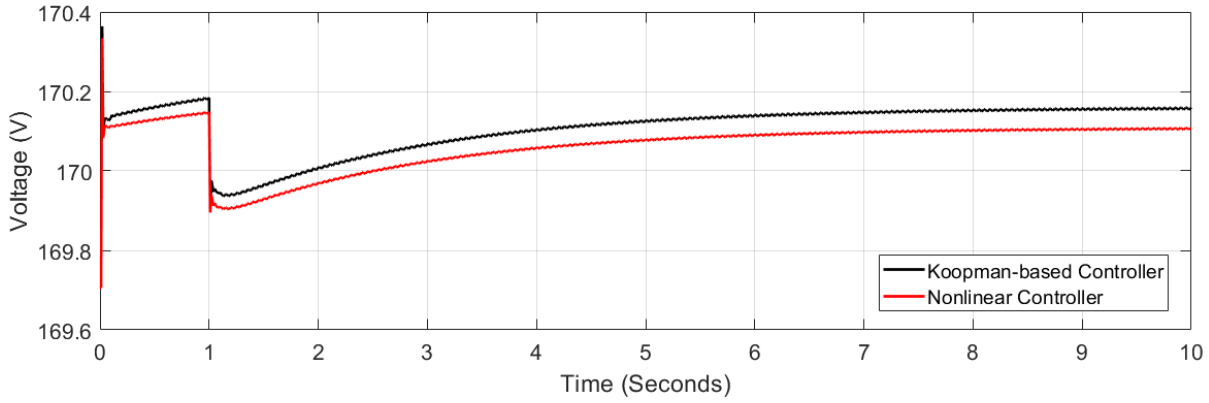

Figure 2.16 – Output voltage at inverter one generated by using the Koopman-based and the nonlinear-based MPC.

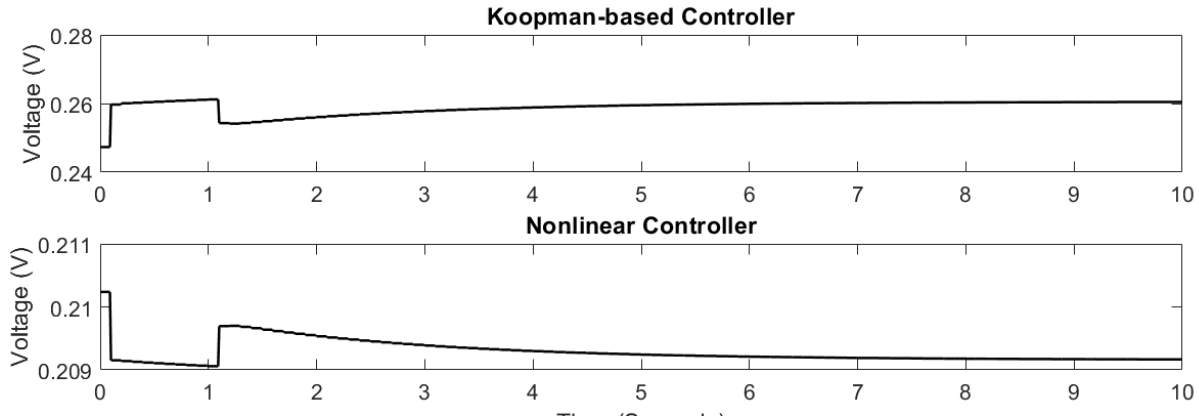

Figure 2.17 – Output Comparison between the MPC signals generated by the Koopman -based and the nonlinear-based approaches.

2.5 Conclusion of the Second Chapter

In this chapter, we present a Koopman-based data-driven controller for MGs with a decentralized perspective. We considered an islanded MG, which only includes inverter-based generators. The MG model is based on power injections decoupled by considering lossless transmission lines. It also assumes that there is enough power available for the control action. We proposed a data-driven control using Koopman operator theory to regulate optimally the MG voltage. The main objective is to minimize the difference in the measured voltage with the reference voltage at each inverter while maintaining the power-sharing condition. The simulation results show that the proposed controller regulates the voltage, and keeps the condition for the reactive power. Also, the Koopman representation of the system by designing a linear predictor reduces significantly the computational time compared with a nonlinear MPC.

This study brings some useful results related to the control of MG and this kind of energy system:

1. The decentralized nature of predictive control allows controlling each agent by only using local information, while the Koopman representation involves some dynamics of the interconnected system.

2. The Koopman representation reduces the computation time for the controller compared with a nonlinear one and simplifies the optimization process.

In this part of the work, it is considered that the dynamics of the system are contained in the data measured. However, the distributed nature of the system has to be considered for the proper regulation and optimization of the system. The next chapters focus the attention on how to deal with this problem.

Chapter 3

Distributed Approaches for Koopman-Based Controllers

3.1 Introduction

Multi-agent theory can manage the problem of interconnected systems, representing the interactions among subsystems as nodes that share a physical magnitude or just information. Electrical systems inherently have a distributed nature, accentuated by the increasing use of inverter-based generators and micro-generation. It was demonstrated that a decentralized control for MGs achieves a proper regulation but fails to bring all the generators to a common point and to achieve global optimum [19]. The natural coupling of power systems implies that any change over the MG components changes the magnitudes of the whole system. In a networked system, the optimization process can be done by a centralized or distributed controller. The second one is preferred because the design of its communication system is more flexible and more reliable to failures.

Several control algorithms have been proposed for the frequency and voltage distributed control in MGs [16]. Most of them rely on the first principles model, such as [19] and [17], which uses the hierarchical control frame based on time-scale to separate each layer. Thus, MG modeling implies the assumption of a complete separation between layers, which is not true. Also, the design of a controller that includes the system restrictions and optimization such as MPC relies on getting a precise model of the system. Different schemes can be used to design a distributed MPC (DMPC) such as non-cooperative and cooperative [77], [78]. The first one optimizes a local cost function and includes a distributed term in the restrictions using local information. The second one defines a cost function for the set of neighbors of each agent, which depends on the information shared. In [65], it is used a nonlinear model of the MG to design a DMPC that uses a nonlinear solver for the optimization algorithm, increasing the computation time.

The limitation of having a nonlinear model can be overcome by using the Koopman representation of the system in the lifted space. The linear representation simplifies the optimization problem, including the design of a DMPC. The non-cooperative MPC only needs local measurements of the state variables, omitting the control inputs of the neighbors, which can be problematic in some scenarios. The cooperative MPC uses control signals from the set of neighbors and also requires the coupling matrix among agents. Here, we determined these matrices that define the interactions among agents by using the EDMD algorithm and the Koopman operator in a centralized approach.

An alternative to the distributed algorithms is the ADMM; this algorithm has been modified to work in a distributed form [79] [58], and it has demonstrated its stability and convergence properties to solve the optimization problem in a distributed form. Distributed ADMM (DADMM) solves an MPC problem while reducing computational time. It enables iterative algorithms to find the minimum, such as gradient descent. The definition of convex cost functions, together with linear restrictions given by the Koopman predictor, simplifies the problem, becoming practical to be used in power systems applications [52], [80].

In this chapter, we developed the non-cooperative, cooperative, and DADMM algorithms by modeling and designing a linear predictor using the Koopman operator. We also determine the coupling matrix in a centralized form for the cooperative problem, and we also show the conditions for the convergence of the algorithms and prove them in a MG.

3.2 Secondary Non-Cooperative Voltage Control

The non-cooperative distributed control includes a distributed term for the state representation of the system [77], which represents the connection of the agent with its set of neighbors as follows

$$\dot{x}_i(t) = A_i x_i(t) + B_i u_i + \sum_{i \in \mathcal{N}_i}^{N} a_{ij}(x_j(t) - x_i(t)), \tag{3.1}$$

where x_j is the state corresponding to the j^{th} agent, $\mathcal{N}_i$ is the set of neighbors of the i^{th} agent, and a_{ij} represents the ij^{th} term of the adjacency matrix.

The additional term depends on the interactions of the agent with its neighbors. This term tends to zero in regulation problems, where all the differences between agents tend

to have the same value. (3.1) can be written in matrix form as

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + Bu + \mathcal{L}\mathbf{x}, \tag{3.2}$$

where $\mathcal{L}$ is the Laplacian matrix of the graph, and $\mathbf{x}$ is the vector of states given by $\mathbf{x} = [x_1 \quad x_2 \quad \dots \quad x_n]$.

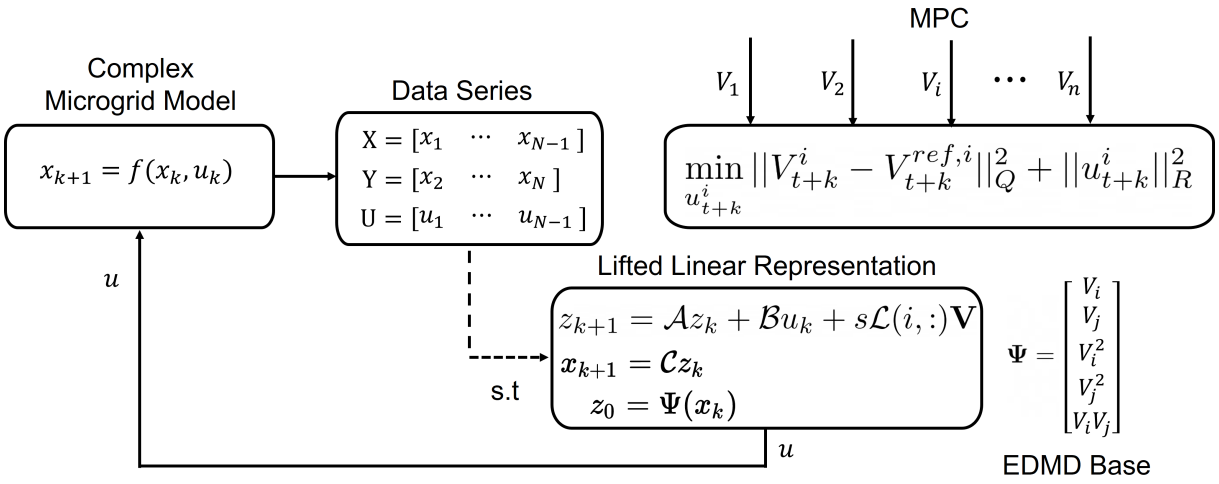

Figure 3.1 – The Koopman-based linear predictor of each agent is used to set a Non-Cooperative MPC by including a consensus term.

Figure 3.1 shows the general scheme for the non-cooperative distributed MPC. Voltage measurements are gathered for the set of neighbors of the i^{th} agent. The MPC is designed to minimize the cost function with the restriction given by the linear representation obtained by the Koopman operator with matrices $\mathcal{A}_i$, $\mathcal{B}_i$, and $\mathcal{C}_i$.

The MPC design, using the linear representation of (2.3) in the lifted space, is complemented by the consensus agreement between the voltage measurements from the set of neighbors of the i^{th} agent. Then, the predictive controller reduces the differences among the voltages of the inverter. The distributed MPC design is given by

$$\min_{u^i_{t+k}} \quad \sum_{k=0}^{H_p} ||V^i_{t+k} - V^{ref,i}_{t+k}||^2_Q + ||u^i_{t+k}||^2_R \tag{3.3}$$

$$\text{s.t.} \quad z^i_{k+1} = z^i_k + T(\mathcal{A}z^i_k + \mathcal{B}u^i_k) + s\mathcal{L}(i,:)\mathbf{V}_i \tag{3.4}$$

$$V_{k+1}^i = \mathcal{C} z_k^i \tag{3.5}$$

$$0.90V^{ref} \leq V_k^i \leq 1.10V^{ref} \tag{3.6}$$

where $\mathcal{L}(i,:)$ corresponds to the i^{th} row of the Laplacian matrix, $s \in \mathbb{R} > 0$ is the gain value for the consensus term and the column vector with the voltage values of each agent

$$\mathbf{V}_i = [V_1 \quad V_2 \quad \dots \quad V_i \quad \dots \quad V_n]^\top.$$

The linear representation in the lifted space allows using already established conditions for MPC design and convergence.

3.2.1 Convergence Analysis

The stability of the control algorithm defined by (3.3)-(3.6) is determined by the behavior of each inverter represented by the linear matrices $\mathcal{A}$, $\mathcal{B}$ in the space of observables, and the selection of the MPC parameters Q, R, and s.

Assumption 4 *The graph $\mathcal{G}$ is connected with $\mathcal{L}$ semi positive definite with an eigenvalue $\lambda_1 = 0$.*

Theorem 3.2.1 *The optimization problem given by (3.3)-(3.6) defined by Q, R, s, and $\mathcal{L}$, with the linear system represented by matrices $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ is asymptotically stable and converges to the optimal solution.*

Proof 3.2.2 *The cost function and the linear system are defined by*

$$L(x,u) = \frac{1}{2}x^\top Q x + \frac{1}{2}u^\top R u \qquad f(x,u) = \mathcal{A}x + \mathcal{B}u + \mathcal{L}\mathbf{x}$$

The optimization problem for the LQ regulator is solved by using the calculus of variations. As shown in [81], the following set of identities should be fulfilled

$$L_x = Qx \qquad L_u = u^\top R \qquad f_x = \mathcal{A} + \mathcal{L} \qquad f_u = \mathcal{B}$$

$$\dot{\lambda}^\top = -L_x - \lambda^\top f_x \qquad L_u + \lambda^\top f_u = 0$$

Assuming $\lambda = Px$

$$u^\top R + \lambda^\top \mathcal{B} = 0 \qquad u = -\mathcal{B}^\top P x R^{-1}$$

$$P\dot{x} = -Qx - A^\top Px - \mathcal{L}Px$$

$$P(\mathcal{A}x + \mathcal{B}u + \mathcal{L}\mathbf{x}) = -Qx - \mathcal{A}^\top Px - \mathcal{L}Px$$

Substituting u

$$P\mathcal{A}x + \mathcal{A}^\top Px + P\mathcal{L}\mathbf{x} + \mathcal{L}Px - P\mathcal{B}\mathcal{B}^\top PR^{-1}x + Qx = 0$$

Then, the next Riccati equation is obtained

$$\mathcal{A}^\top P + P\mathcal{A} + Q - P\mathcal{B}R^{-1}\mathcal{B}^\top P + \mathcal{L}P + P\mathcal{L} = 0$$

For a control input of the form $u = -Kx$*, where* K *is a square gain matrix given by* $K = -ux^{-1}$*, and replacing* $u = -\mathcal{B}^\top PxR^{-1}$ *we obtain* $K = -\mathcal{B}^\top PR^{-1}$*.*

For a Lyapunov function of the form $V(x) = x^\top Px$*, the first derivative of* x *is given by*

$$\dot{V} = \dot{x}^\top Px + x^\top P\dot{x}$$

$$\dot{V} = [(\mathcal{A} - \mathcal{B}K)x + \mathcal{L}\hat{x}]^\top Px + x^\top P[(\mathcal{A} - \mathcal{B}K)x + \mathcal{L}\hat{x}]$$

Replacing K

$$\dot{V} = x[\mathcal{A} - P\mathcal{B}R^{-1}\mathcal{B}^\top P + \mathcal{L}]^\top Px + x^\top P[\mathcal{A} - P\mathcal{B}R^{-1}\mathcal{B}^\top P + \mathcal{L}]x$$

Adding the terms Q*, and* $-Q$ *and comparing them with the Riccati equation*

$$\dot{V} = x[\mathcal{A}P + \mathcal{L}P + P\mathcal{A} - P\mathcal{B}R^{-1}\mathcal{B}^\top P + P\mathcal{L} - Q \\ +Q - \mathcal{B}\mathcal{B}^\top PR^{-1}P]x$$

$$\dot{V} = x[-Q - P\mathcal{B}R^{-1}\mathcal{B}^\top P]x$$

as matrices $Q > 0$*,* $P > 0$*, and* $R^{-1} > 0$ *the system fulfills the condition of* $\frac{d}{dx}V(x) < 0$*, then the closed system is asymptotically stable.*

The next section presents some simulation results for the proposed algorithm and the data-driven simulation to get the Koopman representation of each inverter.

Algorithm 2 Non-Cooperative Distributed MPC Koopman-Based

procedure $u^i(V^i, V^j)$ ▷ Voltage measurements from other inverters
  System Initialization
  Read the value
  Set $\mathcal{A}^i$, $\mathcal{B}^i$, $\mathcal{C}^i$
  Set $\mathbf{V} = [V^i \quad V^2 \quad \dots \quad V^j]$, ▷ Vector of voltages and voltage of reference
  Set V^{ref} ▷ Voltage of reference
  Set H_p, T ▷ Prediction Horizon and Sampling Time
  Set Q, R, s ▷ MPC Gains
  Set $\mathcal{L}$ ▷ Laplacian Matrix
  while $k \leq H_p$ **do**
    $u^i \leftarrow \min_{u^i} \sum_{k=1}^{H_p} ||V^{ref} - V_k^i||_R^2 + ||u_k^i||_Q^2$
    ▷ Cost function
    s.t. $z_{k+1}^i \leftarrow z_k^i + T(\mathcal{A}^i z_k^i + \mathcal{B}^i u_k^i) + s\mathcal{L}(:, i)\mathbf{V}_k$ ▷ Restrictions
      $V^i \leftarrow \mathcal{C}^i z^i$
      $0.90V^{ref} \leq V_k^i \leq 1.10V^{ref}$
    Solve u^i ▷ Solve by using a linear solver
    Select $u^i(1)$ ▷ Select the first optimal value from vector u

3.2.2 Non-Cooperative DMPC Simulations

In this section, we used the same MG model from Figure 2.8. The MG parameters are shown in Table 3.1, and the communication among converters is defined by a graph as shown in Figure 3.2 with adjacency matrix denoted A_1 and Laplacian matrix $\mathcal{L}_1$

$$A_1 = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \qquad \mathcal{L}_1 = \begin{bmatrix} 2 & -1 & 0 & -1 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 3 & -1 & -1 \\ -1 & 0 & -1 & 2 & 0 \\ 0 & 0 & -1 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad \mathcal{L}_2 = \begin{bmatrix} 4 & -1 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

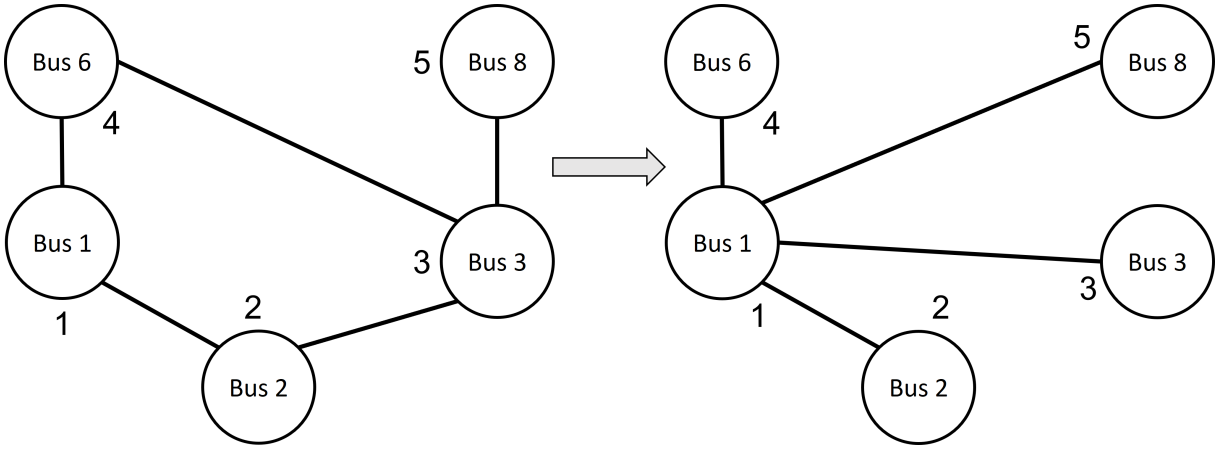

Figure 3.2 – Communication graph among generators. The left graph represents the initial configuration of the communication system, which changes to the configuration represented by the graph at the right.

Load Changing Simulations For this scenario, the MG starts with loads connected at $t = 0$s, then at $t = 10$s, loads are connected to the nodes 3, 5, 6, 9, 14.

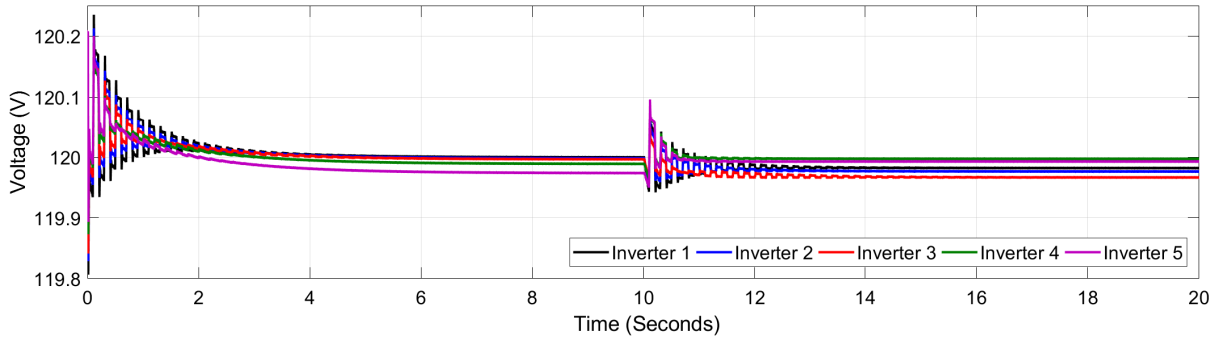

Figure 3.3 – Voltage measured at the output of each inverter using the non-cooperative distributed control. The load changes at $t = 10$s.

Figure 3.3 shows the voltage measured at each inverter with loads connected at $t = 0$s and $t = 10$s. Voltages approximate the reference value at a steady state but have a small gap compared with the decentralized controller. The controller also needs more seconds to reach the reference value. The MPC signals are shown in Figure 3.4, the magnitude of the signals decreases depending on the maximum capacity of each inverter, which guarantees the power-sharing condition as it is shown in the reactive power plot shown in Figure 3.5.

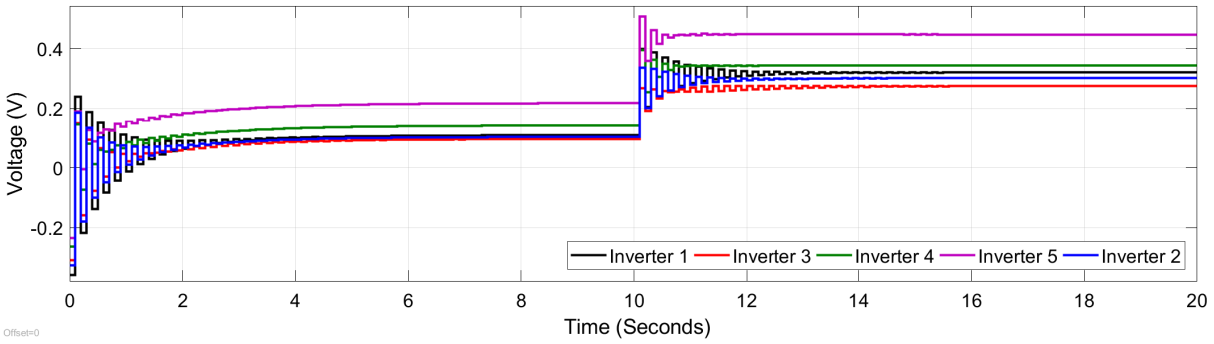

Figure 3.4 – MPC signals for all the inverters when there is a load change at $t = 0$s and $t = 10$s using the non-cooperative scheme.

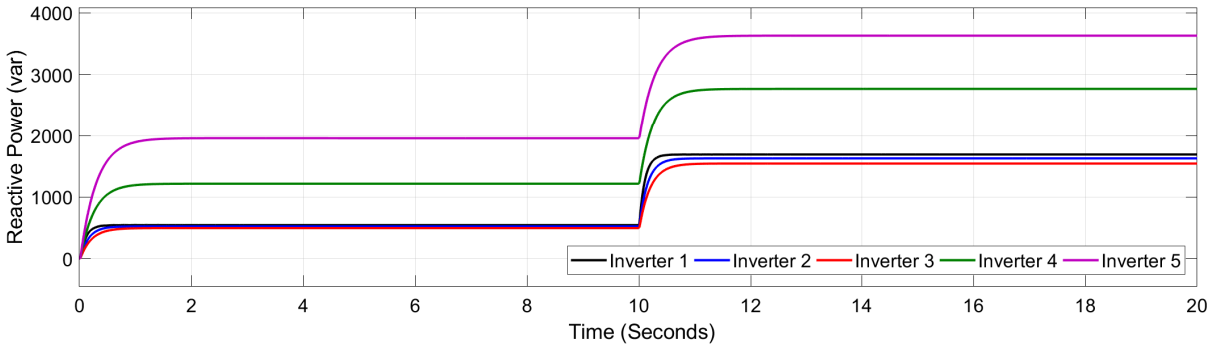

Figure 3.5 – Reactive power measured at each inverter when the load changes at $t = 10$s.

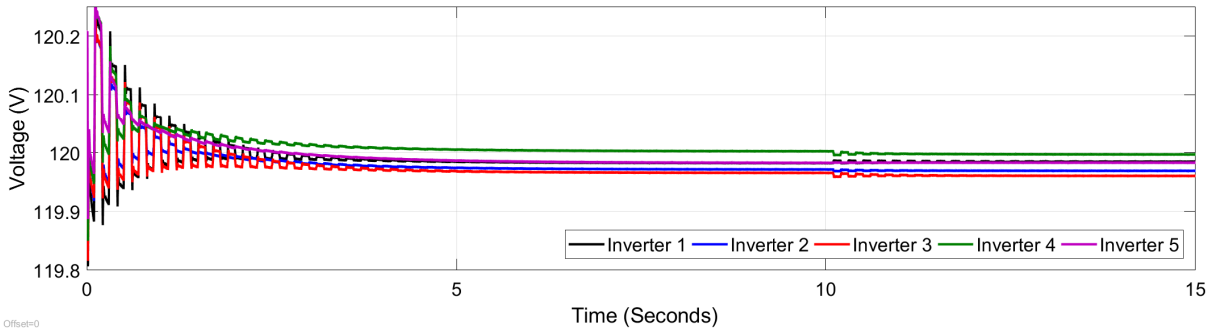

Figure 3.6 – Secondary control voltage at each inverter after a graph change at $t = 10$s.

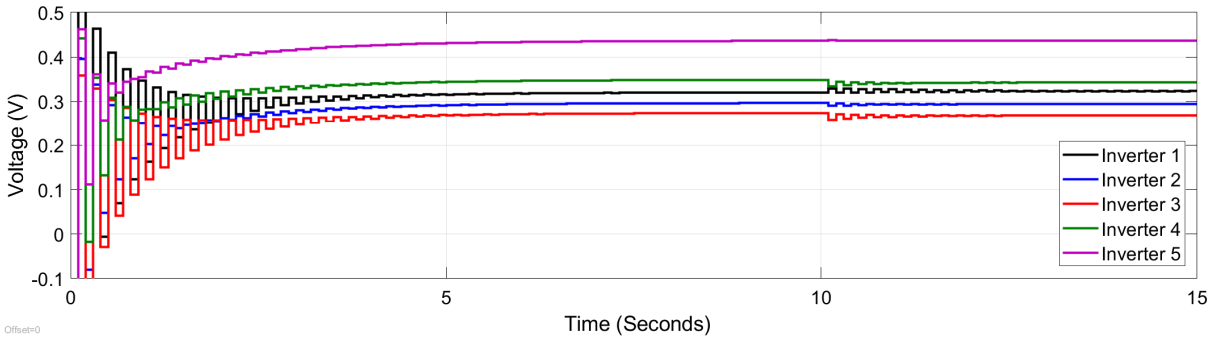

Figure 3.7 – MPC signals when there is a graph changing at $t = 10$s, there is no change in inverter five.

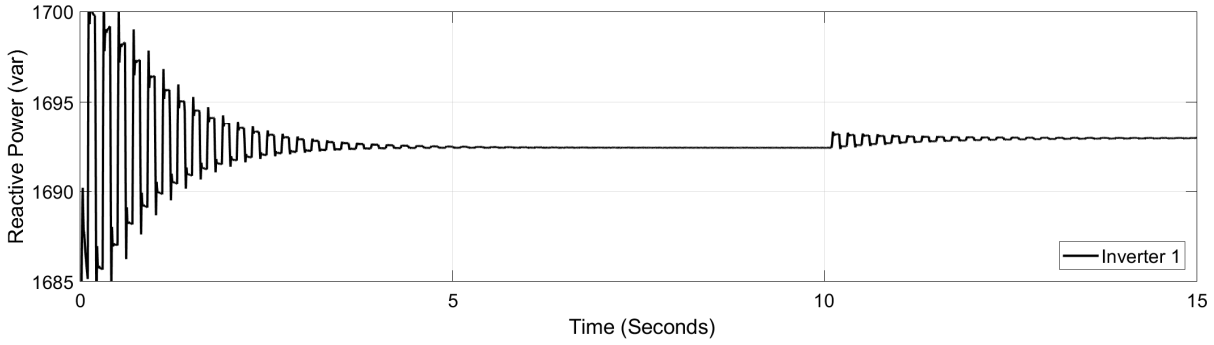

Figure 3.8 – Instantaneous reactive power measured at inverter one for a graph changing at $t = 10$s. There is a little oscillation after the graph changes.

Inverter	1	2	3	4	5
P^{ref} (kW)	10	10	10	10	10
Q^{ref} (kVar)	5	5	5	8	8
m_i (1×10^{-4})	1	1	1	1	1
n_i (1×10^{-4})	0.200	0.200	0.200	0.125	0.125
$v^{ref}(V)$	120	120	120	120	120
τ^i(s)	0.10	0.15	0.20	0.25	0.28
State gain Q^i	11	11	14	14	13
Input gain R^i	1	1	1	1	1
Consensus gain s^i	0.02	0.02	0.02	0.03	0.04
Prediction horizon H_p	10	10	10	10	10
f(Hz)	60	60	60	60	60

Table 3.1 – Microgrid parameters for non-cooperative case.

Graph Changing Simulations For this scenario, the communications graph among inverters changes at $t = 10$s, as it is shown in Figure 3.2 where the condition of having at least a spanning tree for consensus reaching is guaranteed for both graphs. The voltage from each inverter is shown in Figure. 3.6, where voltages gather around the reference value. The MPC signal of each inverter is shown in Figure 3.7 where the magnitude of the signals changes at $t = 10$s, especially at inverter five. Finally, the instantaneous reactive power is shown in Figure 3.8, where there are small changes in the reactive power after the graph changing, with oscillations that vanish after two seconds.

3.3 Secondary Cooperative Voltage Control

In decentralized MPC, each subsystem carries out the optimization problem by only using local measurements; one of its advantages is its low computational cost. However, it may lead to a low performance in the control optimization or even stability issues [82]. In the non-cooperative distributed MPC, each agent communicates its state to other agents, and uses the state measurements to generate local control signals. This type of control includes the coupling among agents; however, it can have some limitations when the dynamic coupling is very strong. In non-cooperative MPC, the system might not reach a global optimum and only a Nash equilibrium [83].

The cooperative distributed MPC combines the control outputs of local controllers to solve a global MPC objective function, as shown in the general scheme for cooperative

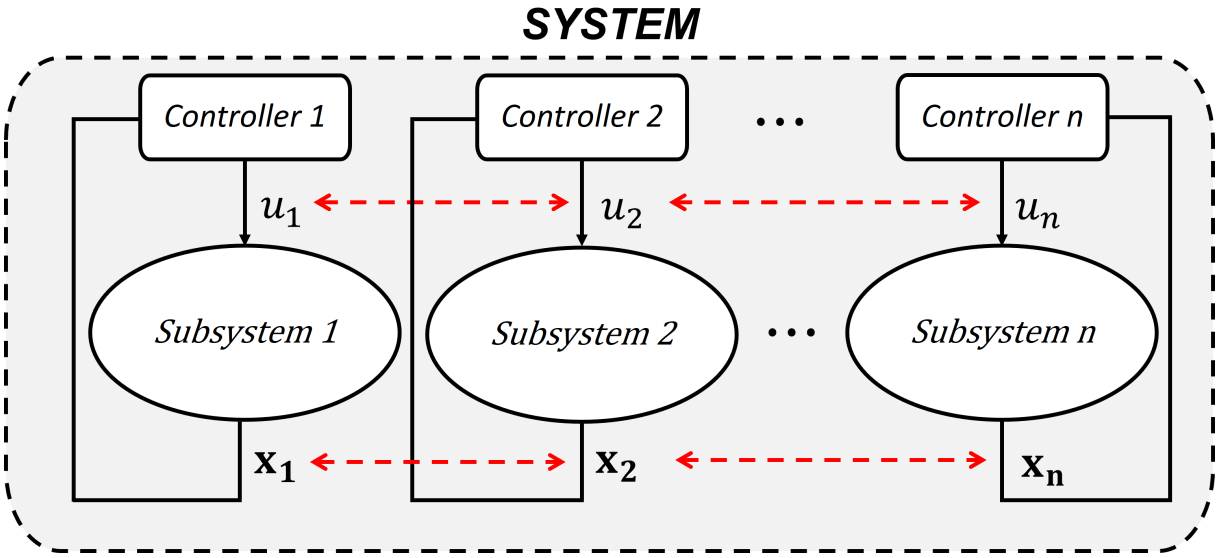

Figure 3.9 – Cooperative DMPC scheme.

MPC in Figure 3.9. In this case, the global problem might converge to a Pareto optimum with the disadvantage that requires communication among all the agents to know the control signals, which is in many cases unfeasible [83]. It is possible to use an approximation of the global objective function restricted to the set of neighbors of an agent. However, this reduces the control performance by only reaching a sub-optimal control signal [83], in [84] it is presented a cooperative DMPC including convergence and stability features.

The state for the i^{th} agent is given by

$$\dot{x}_i(t) = A_{ii}x_i(t) + B_{ii}u_i + \sum_{j=1,j\neq i}^{N} A_{ij}x_j(t) + B_{ij}u_j(t)), \tag{3.7}$$

where x_j is the state corresponding to the j^{th} agent, A_{ii} represents is the local state matrix, B_{ii} is the local input matrix, A_{ij} is the global matrix that represents the state interaction with the neighbors, and B_{ij} is the global input matrix.

Different from the non-cooperative control presented previously, each agent should know the input matrix B_{ij}. As each agent only knows the state and input of its neighbors, we propose to determine the input matrices by using data gathered from the set of neighbors.

3.3.1 Determination of the Coupling Matrices

The local and global matrices can be approximated by using the Koopman operator with the EDMD algorithm. For a system with N distributed subsystems, the input matrix $B_{ij} \in \mathbb{C}^{(N\times n)\times(N\times n)}$ determines the relation of each local input with the set of neighbors of the i^{th} agent, with the input matrix of each agent denoted by $B_{ii} \in \mathbb{C}^{(n\times 1)}$.

$$\mathbf{B} = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1\times(N\times n)} \\ B_{21} & B_{22} & \cdots & B_{2\times(N\times n)} \\ \vdots & \cdots & \ddots & \vdots \\ B_{(N\times n)\times 1} & B_{(N\times n)\times 2} & \cdots & B_{(N\times n)\times(N\times n)} \end{bmatrix}$$

In [85], it is proposed a data-driven algorithm to determine the Koopman operator of a distributed system. In contrast, with a centralized problem, the distributed algorithm uses a series of data from the set of neighbors of each agent to determine a global Koopman operator similar to the Laplacian of a distributed system or supra-Laplacian. This algorithm requires defining the data for each agent depending on its set of neighbors, and it does not consider the effect of inputs in a coupled system.

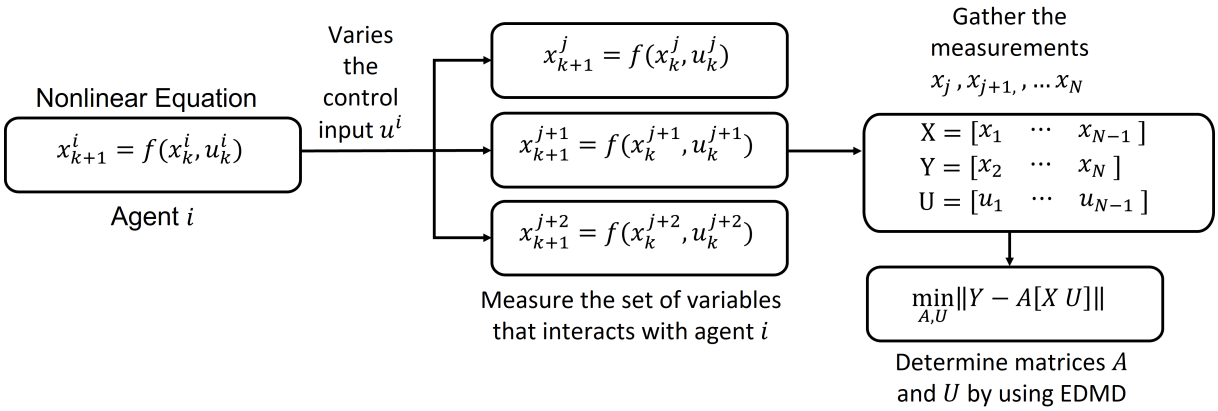

Figure 3.10 – General scheme to determine the input matrix B_{ij} by using EDMD. The experiment should be done for each agent to determine how the local input affects its neighbors.

To build the coupling matrix based on data, we use a centralized approach by considering only the effect of the i^{th} agent over the j^{th} if it belongs to its set of neighbors, as it is shown in Figure 3.10. Similarly, we consider the effect of the j^{th} input over the other agents.

Cooperative Distributed MPC Design For this distributed MPC, it is assumed that the global objective function is the sum of the weighted local objective functions, in which each agent shares its state and control signal at each sampling time. The local objective function for the i^{th} agent is given by

$$L_i(x_i) = \sum_{k=0}^{H_p-1} h_i(x_i(k), u_i(k)), \tag{3.8}$$

where H_p is the prediction horizon, and h_i is the convex objective function for the i^{th} agent that depends on the measured state and input at iteration k. As each agent has a local function of this type, the global objective function can be written as the convex weighted combination of local functions as [83]

$$\mathbf{L}(x) = \sum_{i=1}^{N} \omega_i L_i, \tag{3.9}$$

$$\mathbf{L}(x) = \sum_{i=1}^{N} \omega_i \left[\sum_{k=0}^{H_p-1} h_i(x_i(k), u_i(k)) \right], \tag{3.10}$$

where $\omega_i \geq 0$ is the weight assigned to each local function that must fulfill the condition

$$\sum_{i=1}^{N} \omega_i = 1. \tag{3.11}$$

Each subsystem solves the global function given by (3.10), obtaining an optimal local signal at iteration p as u_i^*. All the agents share their optimal local control signals and calculate their final control input as

$$u_i^p = \omega_i u_i^* + (1 - \omega_i) u_i^{p-1}, \tag{3.12}$$

what is an update of the type Gauss-Jacobi for distributed optimization [83].

The state equation for the coupled system (3.7), is rewritten as

$$x_{k+1} = \mathbf{A}x_k + \mathbf{B}\mathbf{u}, \tag{3.13}$$

where $\mathbf{A}$ is given by the matrices A_{ii} in its diagonal, and A_{ij} which are equal to zero if there is no interaction between agent i and j. The matrix $\mathbf{B}$ contains local matrices B_{ij} determined by the EDMD algorithm. The cooperative distributed optimization problem

is defined as

$$\min_{u^i} L(z) \tag{3.14}$$

$$\text{s.t.} \qquad z^i_{k+1} = z^i_k + T(\mathcal{A}^i z^i + \mathcal{B}^i u^i) + s\mathcal{L}(:,i)\mathbf{Z} \tag{3.15}$$

$$0.9z^{ref} \leq z^i \leq 1.1z^{ref} \tag{3.16}$$

$$z^i = \mathcal{C}^i x^i, \tag{3.17}$$

where $Z = [z^1 \quad \dots z^{\mathcal{N}}]$ is the vector of measurements from the set of neighbors of the i^{th} agent.

Assumption 5 *For all agents, i*

- *The inputs $u^i \in \mathbb{U}_i$ and $u^j \in \mathbb{U}_j$ subsets are convex.*
- *The constraints are uncoupled. The optimization region of the optimal input is not affected by the inputs of the other agents.*
- *The systems given by Koopman matrices $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ are stabilizable and detectable.*
- *Local systems do not have unstable modes. The eigenvalues of $\mathcal{A}$ are located inside the unit circle.*

Lemma 1 *[86] The cost function $\mathbf{L}(z)$ decreases and converge as the iterations $k \to \infty$.*

Proof 3.3.1 *[86] Assuming that the local cost functions h_i are convex, the global cost function can be written as*

$$L(z, u^{p+1}) \leq \omega_1 L(z, u_i, u^p_j, u^p_N) + \omega_2 L(z, u^p_i, u_j, u^p_N) + \cdots + \omega_N L(z, u^p_i, u^p_j, u_N)$$

As $p \to \infty$, local inputs tend to the optimal value, then

$$L(z, u^{p+1}) \leq \omega_1 L(z, u^p_i, u^p_j, u^p_N) + \omega_2 L(z, u^p_i, u^p_j, u^p_N) + \cdots + \omega_N L(z, u^p_i, u^p_j, u^p_N)$$

as the condition to combine the local inputs is of the form $\omega_1 + \omega_2 + \cdots + \omega_N = 1$, the last expression can be written as

$$L(z, u^{p+1}) \leq L(z, u^p)$$

As the algorithm decreases as $p \to \infty$ the algorithm converges.

Algorithm 3 Cooperative Distributed MPC Koopman-Based

procedure $u_i(V^i, V^j), u_j^{p-1}$ ▷ Voltage measurements from other inverters
  System Initialization
  Read the value
  Set $\mathcal{A}^i$, $\mathcal{B}^i$, $\mathcal{C}^i$
  Set $\mathbf{V} = [V^i \quad V^2 \quad \dots \quad V^j]$ ▷ Vector of voltages and voltage of reference
  Set V^{ref} ▷ Voltage of reference
  Set H_p, T ▷ Prediction Horizon and Sampling Time
  Set Q, R, s ▷ MPC Gains
  Set $\mathcal{L}$ ▷ Laplacian Matrix
  while $k \leq H_p$ **do**
    $L(x) = \sum_{i=1}^{N} \omega_i \left[\sum_{k=0}^{H_p-1} h_i(x_i(t,k), u_i(t,k))\right]$ ▷ Global Objective Function
    $u_i^* \leftarrow \min_{u_i} L(x)$ ▷ Solve for each agent
    s.t. $z_{k+1}^i \leftarrow z_k^i + T(\mathcal{A}^i z_k^i + \mathcal{B}^i u_k^i) + s\mathcal{L}(:,i)\mathbf{V}_k$ ▷ Local restrictions
      $V_k^i \leftarrow \mathcal{C}^i z_k^i$
      $0.90V^{ref} \leq V_k^i \leq 1.10V^{ref}$
    Update the input $u^{p,i} = \omega_i u_i^* + (1-\omega_i)u_i^{p-1}$ ▷ Gauss-Jacobi Update
    Select $u_i^* \leftarrow u_i^p$
    $k \leftarrow k+1$

3.3.2 Cooperative DMPC Simulations

The MG model from Figure 2.8, whose parameters are shown in Table 3.2, with communication among converters defined by the graph shown in Figure 3.2 whose adjacency matrix A_1 and Laplacian matrix $\mathcal{L}_1$ are the same.

Load Changing Simulations For this scenario, the MG starts without any load connected at $t = 0$s, then at $t = 5$s, five 2kvar loads are connected to nodes 3, 5, 6, 9, 14.

The RMS voltage at each inverter is shown in Figure 3.11, after the load connection at $t = 5$s, the voltage stabilizes after approximate 1s. The final voltages are close to the reference value. The reactive power does not show a particular behavior; it varies according to the low-pass filter of the local controllers, as shown in Figure 3.12. Finally, the control signals are shown in Figure 3.13, where the major control actions are made by the controller of the inverter.

Graph Changing Simulations We used the one shown in Figure 3.2 for the simulations when the graph changes. The system starts with five 2kvar loads connected at

Table 3.2 – Cooperative DMPC and MG parameters.

Inverter	1	2	3	4	5
P_{ref} (kW)	10	10	10	10	10
Q_{ref} (kVar)	5	5	5	8	8
m_i $(1 \times 10^{-}4)$	1	1	1	1	1
n_i $(1 \times 10^{-}4)$	0.200	0.200	0.200	0.125	0.125
$V_{ref}(V)$	170	170	170	170	170
τ_i(s)	0.10	0.15	0.20	0.25	0.28
Consensus gain $\mathcal{S}$	0.1	0.1	0.1	0.1	0.1
State gain Q	10	10	10	10	10
Input gain R	1	1	1	1	1
Prediction Horizon H_p	10	10	10	10	10
Frequency (Hz)	60	60	60	60	60
Cooperative MPC gain ω	$\omega_1 = 0.8$ $\omega_2 = 0.1$ $\omega_3 = 0.1$	$\omega_1 = 0.8$ $\omega_2 = 0.1$ $\omega_3 = 0.1$	$\omega_1 = 0.8$ $\omega_2 = 0.1$ $\omega_3 = 0.1$	$\omega_1 = 0.7$ $\omega_2 = 0.1$ $\omega_3 = 0.1$ $\omega_4 = 0.1$	$\omega_1 = 0.9$ $\omega_2 = 0.1$

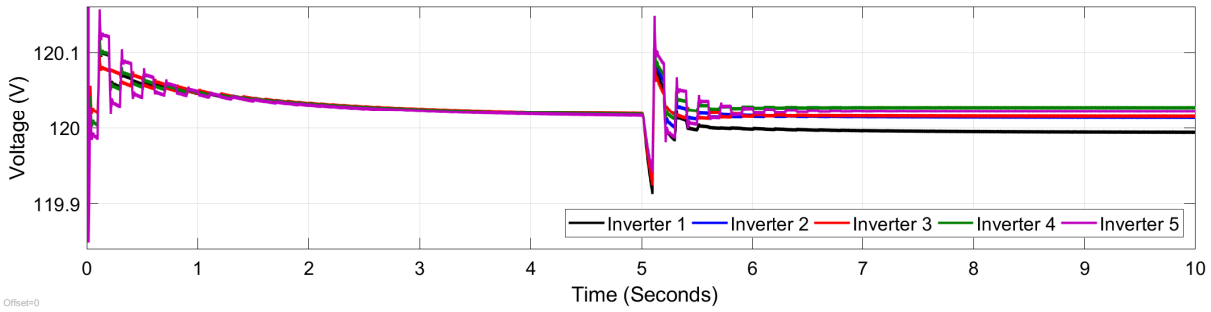

Figure 3.11 – RMS voltage at each inverter when loads are connected at $t = 5$s using the distributed cooperative MPC. After some oscillations, the voltage stabilizes after almost one second around the reference value of 120V.

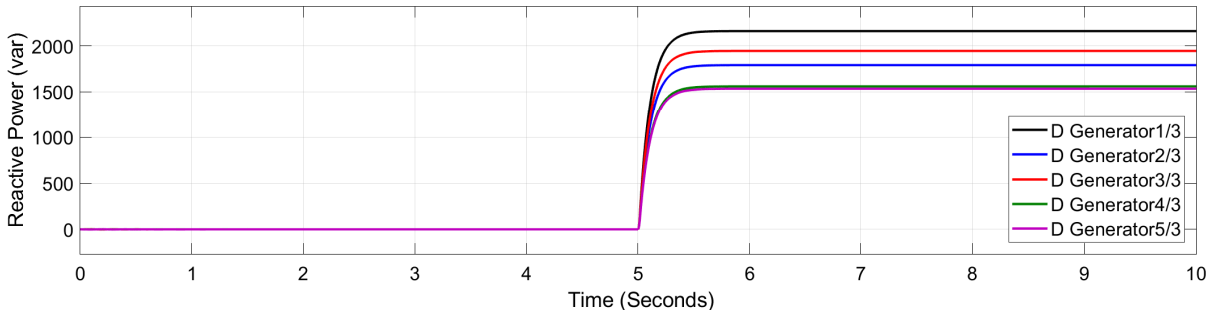

Figure 3.12 – Reactive power measured at each inverter with loads connected at $t = 5$s and using the cooperative distributed MPC.

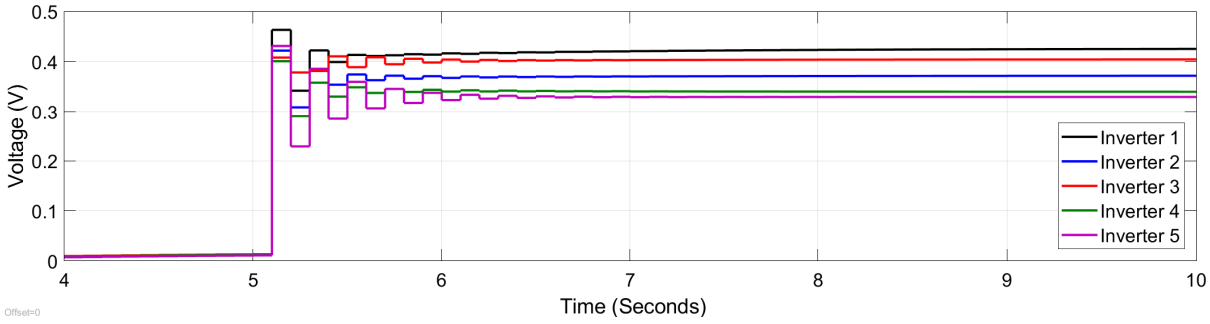

Figure 3.13 – Control signals generated by the cooperative DMPC and loads are connected at $t = 5$s.

nodes 3, 5, 6, 9, 14 at $t = 0$s, then at $t = 5$s the graph changes.

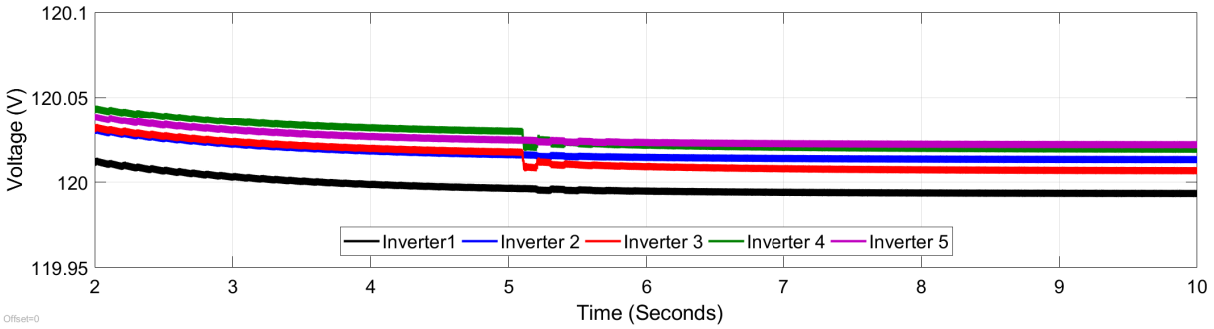

Figure 3.14 – RMS voltage at each inverter for a graph changing at $t = 5$s, the loads are connected at $t = 0$s.

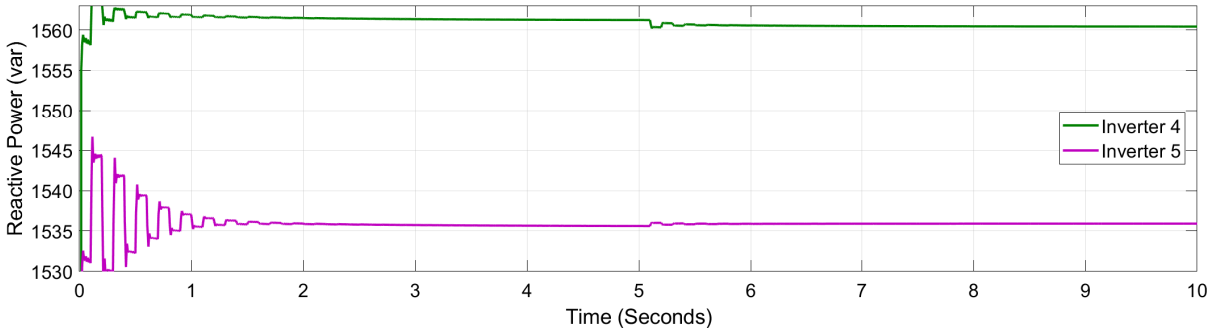

Figure 3.15 – Instantaneous reactive power measured at each inverter with load connected at $t = 0$s, and graph changing at $t = 5$s. The change in the graph produces only a small variation in the reactive power.

For a graph changing at $t = 5$s, the voltage for each inverter is shown in Figure 3.14, inverter one reaches the reference value while the other inverters voltage moves around the reference value. In a different graph, the instantaneous reactive power is shown in Figure 3.15, the graph changing also produces only a small voltage variation that appears in the reactive power. The control signals are shown in Figure 3.16 where we can appreciate the variations in the signals due to the change in the graph, especially in inverters three and four.

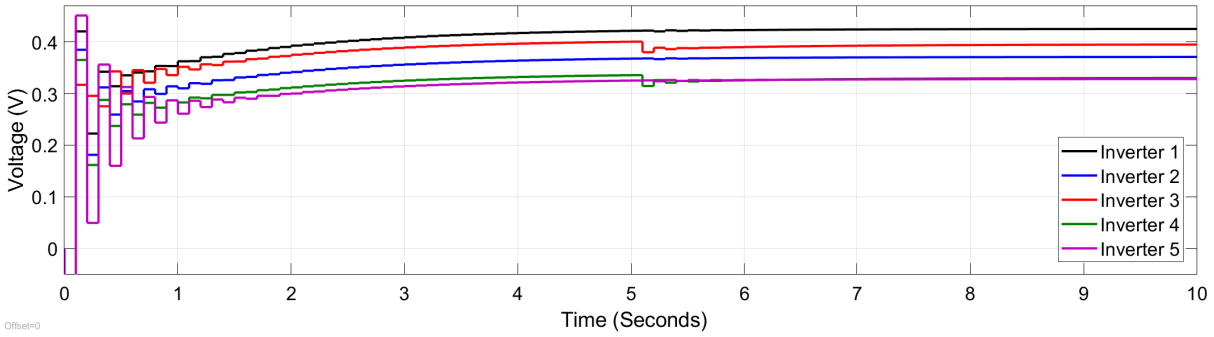

Figure 3.16 – Control signals for the cooperative DMPC, with load connected at $t = 0$s and graph changing at $t = 5$s.

3.4 Koopman-Predictor and ADMM

The previous algorithms solved the optimization problem by including a consensus term in the restrictions for the non-cooperative case. For the cooperative case, it is necessary to know the control input for the set of neighbors and include the coupling terms in the restrictions.

Now, we solve the optimization problem in a distributed and iterative form. The main idea is that each agent only knows its objective function and shares information with a limited set of neighbors. The MPC design allows us to convert the local optimization problem, including the restrictions, to a linear problem defined by a set of matrices. The state update can be nonlinear, which is set as a linear problem using the Koopman-based linear predictor.

3.4.1 MPC Matrix Definition

The tracking problem is defined as a regulation problem by eliminating the constant term during the identification to facilitate the MPC design. Thus, the new objective is to make zero the steady-state error. To do this, we use the following objective function

$$V = X^\top QX + U^\top RU, \tag{3.18}$$

as we can predict future states based on the initial state and the vector of inputs, the next two matrices are defined

$$G_u = \begin{bmatrix} B & 0 & 0 & 0 \\ AB & B & 0 & 0 \\ \vdots & 0 & 0 & 0 \\ A^{H_p-1} & A^{H_p-2} & \dots & B \end{bmatrix} \qquad G_x = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{H_p} \end{bmatrix}$$

using both matrices we can find a global variable X that depends on the initial value of the form

$$X = G_x x(0) + G_u U. \tag{3.19}$$

Substituting the value of X in (3.18), and eliminating the terms that do not depend on U we have

$$H = G_u^\top Q G_u + R \qquad F = x^\top G_x^\top Q G_u \tag{3.20}$$

The restrictions are given by

$$A_x x(k) \leq b_x \qquad A_u u(k) \geq b_u \tag{3.21}$$

The state and input restrictions for the whole prediction horizon are written as

$$\hat{A}_x X \leq \hat{b}_x \qquad \hat{A}_u U \geq \hat{b}_u, \tag{3.22}$$

with

$$\hat{A}_x = \begin{bmatrix} A_x & 0 & 0 & 0 \\ 0 & A_x & 0 & 0 \\ 0 & 0 & A_x & 0 \\ 0 & 0 & \dots & A_x \end{bmatrix} \quad \hat{b}_x = \begin{bmatrix} b_x \\ \vdots \\ b_x \\ b_x \end{bmatrix} \quad \hat{A}_u = \begin{bmatrix} A_u & 0 & 0 & 0 \\ 0 & A_u & 0 & 0 \\ 0 & 0 & A_u & 0 \\ 0 & 0 & \dots & A_u \end{bmatrix} \quad \hat{b}_u = \begin{bmatrix} b_u \\ \vdots \\ b_u \\ b_u \end{bmatrix}$$

The global MPC matrices for the whole prediction horizon including the restriction, are written as

$$\begin{bmatrix} \hat{A}_x G_u \\ \hat{A}_u \end{bmatrix} \leq \begin{bmatrix} \hat{b}_x - \hat{A}_x G_x x(0) \\ \hat{b}_u \end{bmatrix},$$

which is rewritten as $\hat{A}U \leq \hat{b}$, with

$$\hat{A} = \begin{bmatrix} \hat{A}_x G_u \\ \hat{A}_u \end{bmatrix} \qquad \hat{b} = \begin{bmatrix} \hat{b}_x - \hat{A}_x G_x x(0) \\ \hat{b}_u \end{bmatrix}.$$

Finally, the optimization problem is set as follows

$$\frac{1}{2}U^\top HU + FU \tag{3.23}$$

$$\text{s.t} \quad \hat{A}U \leq \hat{b} \tag{3.24}$$

3.4.2 Distributed ADMM Design

The MPC problem is set to be solved by an alternating method, such as multipliers or ADMM algorithms. The general scheme of the problem is shown in Figure 3.17, where we have the nonlinear system, the linear predictor, the MPC problem matrices, and the general optimization problem that is solved in a distributed manner by ADMM. Here we use the distributed algorithm proposed by [87], [88] where for a distributed system with N agents with $x_i \in \mathbb{R}^n$ is the local state of the i^{th} agent, as follows

$$\begin{aligned} x_i(k) &= \arg\min_{x_i} \left\{ f_i(x_i) - \sum_{j\in\mathcal{N}} z_{ji}(k)x_i + \frac{1}{2}\rho d_i x_i^2 \right\} \\ z_{ij}(k+1) &= (1-\alpha)z_{ji}(k) - \alpha z_{ij}(k) + 2\alpha\rho x_j(k) \end{aligned} \tag{3.25}$$

where $f_i : \mathbb{R} \cup \{+\infty\}$ is a convex local cost for each node, $\alpha > 0$ is the step size, and ρ is the penalty term.

Definition 1 *[58] The convex function f is an extended value function, strongly convex whose Hessian is positive definite of the form $\nabla^2 f(x) \succeq mI$ with $m > 0$.*

Solving problem (3.23) applying (3.25) we have

$$\begin{aligned} u(k) &= \arg\min_{u} \left\{ \frac{1}{2}u^2 H + F'u - \sum_{j\in\mathcal{N}} z_{ji}(k)u + \frac{1}{2}\rho d u^2 \right\} \\ z_{ij}(k+1) &= (1-\alpha)z_{ji}(k) - \alpha z_{ij}(k) + 2\alpha\rho x_j(k) \end{aligned} \tag{3.26}$$

Optimal input u can be obtained by directly solving (3.26) as follows

$$\frac{d}{du}u(k) = \left\{ u\left(H + \frac{1}{2}\rho d\right) + F' - \sum_{j\in\mathcal{N}} z_{ji}(k) \right\} = 0,$$

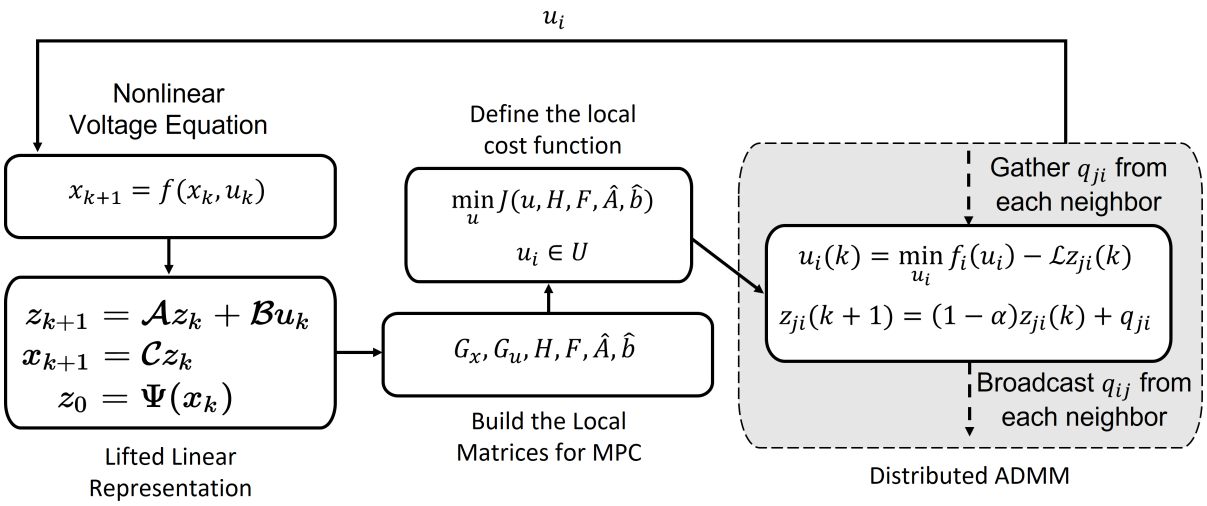

Figure 3.17 – General schematics for the ADMM algorithm using the error of the system.

then, the optimal input u is given by

$$u(k) = \left(H + \frac{1}{2}\rho d\right)^{-1} \left(-F' + \sum_{j \in \mathcal{N}} z_{ji}(k)\right) \tag{3.27}$$

An alternative to solve by using (3.27) that needs to calculate the inverse matrix H, we can use a gradient descent method that, through iterations, finds the optimal solution. We need to define the Lagrangian of (3.23) and (3.24) as

$$L(u, w) = \frac{1}{2}U^\top HU + FU + w(\hat{A}U - \hat{b}), \tag{3.28}$$

and then use an iterative method, such as ADMM.

To design the MPC controller with ADMM, we use the error instead of the direct measurements of the voltages of the systems. Thus, we defined the error for the i^{th} node as

$$e^i = V^{ref,i} - V^i, \tag{3.29}$$

the error identification can be written in the Koopman mode as

$$z^i_{k+1} = \mathcal{A}^i z^i_k + \mathcal{B}^i u^i_k \qquad e^i = \mathcal{C}^i x^i \tag{3.30}$$

$$\begin{bmatrix} e^i \\ e^j \\ (e^i)^2 \\ (e^j)^2 \\ e^i e^j \end{bmatrix}_{k+1} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{bmatrix} \begin{bmatrix} e^i \\ e^j \\ (e^i)^2 \\ (e^j)^2 \\ e^i e^j \end{bmatrix}_k + \begin{bmatrix} b_{11} \\ b_{12} \\ b_{13} \\ b_{14} \\ b_{15} \end{bmatrix} u_k^i \tag{3.31}$$

Algorithm 4 Distributed ADMM for MPC Koopman-Based

procedure INPUT: STEP SIZE α, PENALTY TERM ρ, TERMINATION CONDITION K, $u^i(0)$, $z_{ji}(0)$ $\quad j \in \mathcal{N}$, V^i, V^j ▷ Voltage measurements from other inverters
    System Initialization
    Read the value
    Set $\mathcal{A}^i$, $\mathcal{B}^i$
    Set V^{ref} ▷ Voltage of reference
    Set H_p, T ▷ Prediction Horizon and Sampling Time
    Set Q, R ▷ MPC Gains
    procedure FOR($k = 1 : H_p$)
        Set G_x, G_u, H, F, $\hat{A}$, $\hat{b}$
    end
    while $k \leq K$ **do**
        $e_k^i \leftarrow V^{ref} - V_k^i$
        $e_k^j \leftarrow V^{ref} - V_k^j$ ▷ Determine the error
        $\arg\min_{u^i}\{\frac{1}{2}u_k^2\left(H + \frac{1}{2}\rho d\right) + Fu_k + w(\hat{A}u_k - \hat{b} - \sum_{j\in\mathcal{N}} z_{ji}u_k\}$
        $z_{ji}(k+1) \leftarrow (1-\alpha)z_{ji} + q_{ji}$
        $q_{ij} \leftarrow -z_{ji} + 2\rho u(k)$
        Select $u^i(1)$ ▷ Select the first optimal value from vector u
        $z_{k+1}^i \leftarrow z_k^i + T(\mathcal{A}^i z_k^i + \mathcal{B}^i u_k^i)$
        $V_k^i \leftarrow \mathcal{C}^i z_k^i$
        Broadcast q_{ij} to each neighbor

3.4.3 Distributed ADMM Simulations

Using the MG model from Figure 2.8 whose parameters are shown in Table 3.3, and the communication among converters defined by a graph as shown in Figure 3.2, we obtain the next simulation results.

Load Changing Simulations For this scenario, the MG starts with loads connected at $t = 0$s, then at $t = 5$s, five loads of 2kvar are connected to nodes 3, 5, 6, 9, 14.

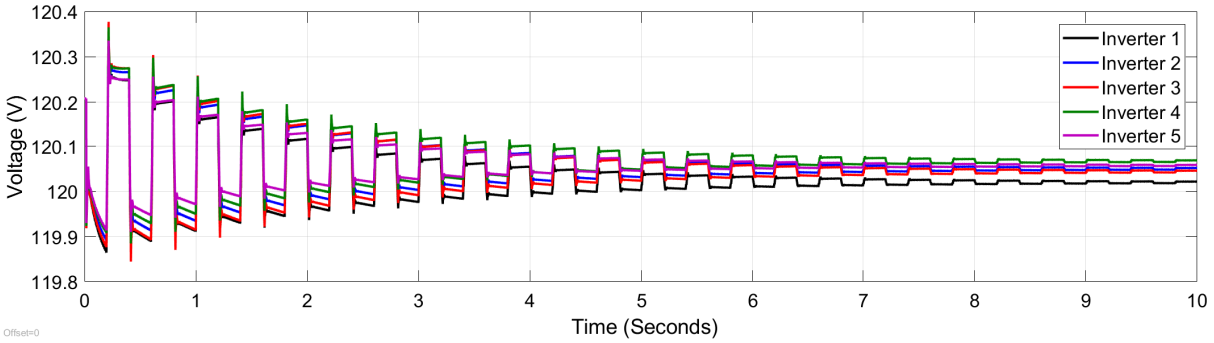

Figure 3.18 – Voltage RMS measured at each inverter after connecting a load at $t = 0$s by using the distributed ADMM algorithm.

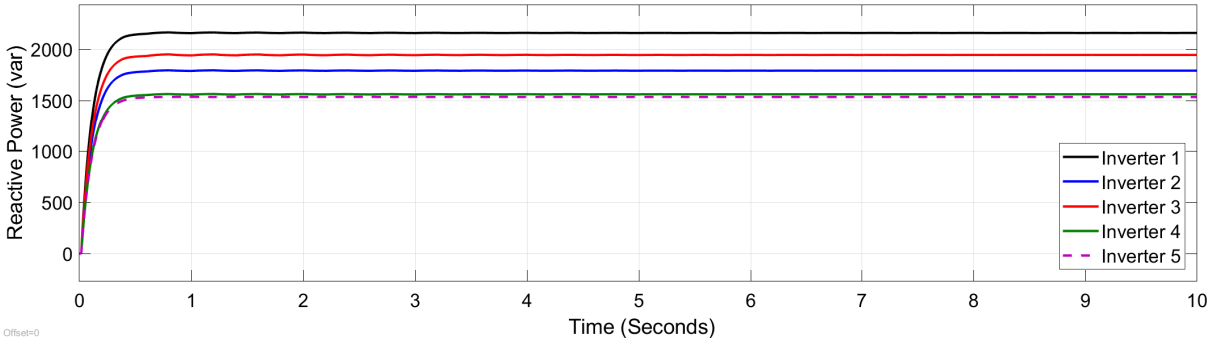

Figure 3.19 – Reactive power measured at each inverter after a load changing at $t = 0$s, using the distributed ADMM.

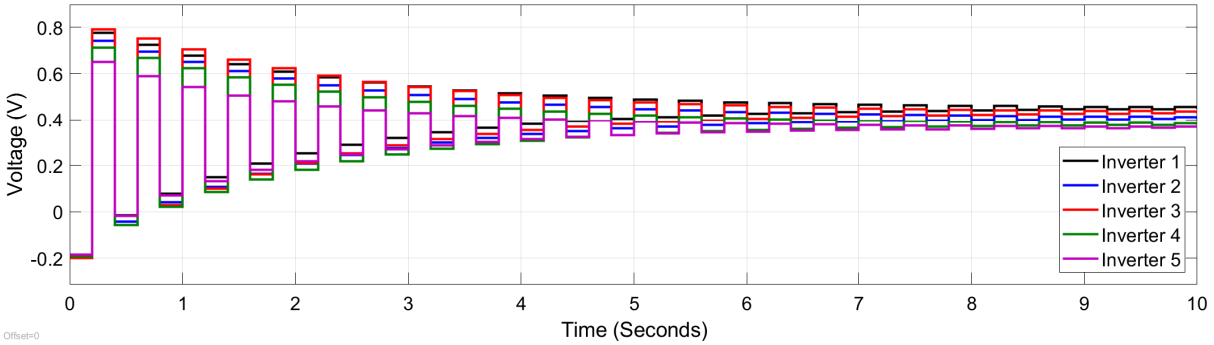

Figure 3.20 – Control signals generated by the distributed ADMM with a load change at $t = 5$s.

Inverter	1	2	3	4	5
P^{ref} (kW)	10	10	10	10	10
Q^{ref} (kVar)	5	5	5	8	8
m_i (1×10^{-4})	1	1	1	1	1
n_i (1×10^{-4})	0.200	0.200	0.200	0.125	0.125
$V^{ref}(RMS)$	120	120	120	120	120
τ^i(s)	0.10	0.15	0.20	0.25	0.28
f(Hz)	60	60	60	60	60
Penalty term ρ	0.1	0.1	0.1	0.1	0.1
Step size α	0.5	0.5	0.5	0.5	0.5

Table 3.3 – Microgrid parameters for control with distributed ADMM.

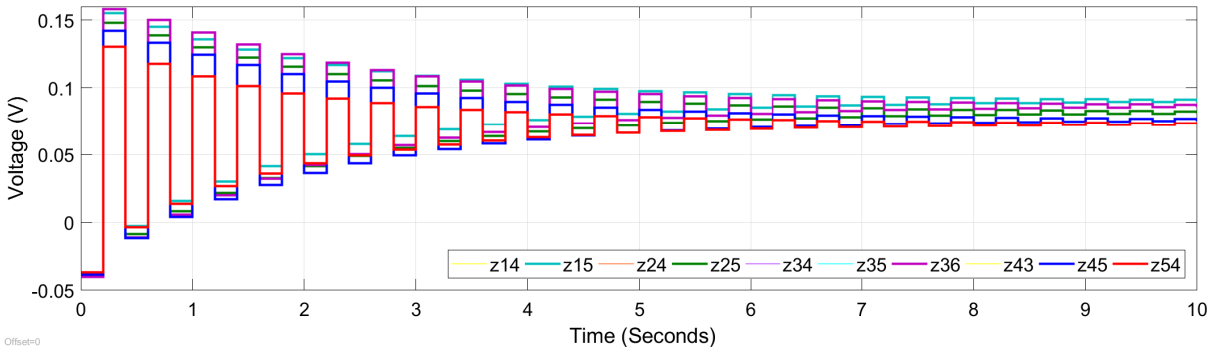

Figure 3.21 – Auxiliary variables z_{ij} for the distributed ADMM algorithm when connecting a load at $t = 5$s.

The RMS voltage at each inverter is shown in Figure 3.18; the system response at steady-state is similar to the cooperative control. However, the system takes more time to reach the reference value. The correct setting of parameters α and ρ determines the convergence speed. However, a high penalty value could affect the convergence of all the agents, while a high step value allows finding the optimal value faster, but it can make the system unstable. The medium reactive power and the control signals are shown in Figure 3.19, and in Figure 3.20, the five control signals converge to a common value in more than ten seconds. Finally, the magnitude of the auxiliary variables is shown in Figure 3.21. Those values change when there is a variation in the communication graph. The algorithm converges faster if the control action is applied in a smaller sampling time.

Graph Changing Simulations The graph changes as it is shown in Figure 3.2, the five loads of 2kvar are connected to the nodes 3, 5, 6, 9, 14 at $t = 0$s, then the graph changes at $t = 5$s.

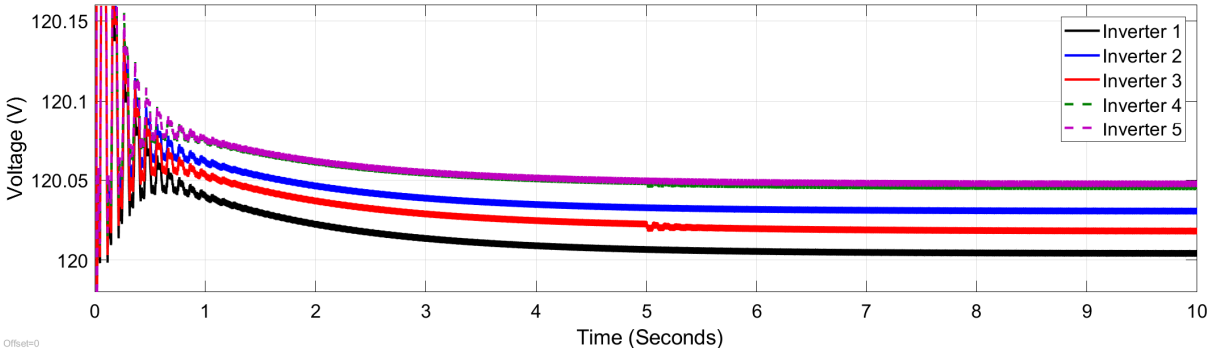

Figure 3.22 – RMS voltage at each inverter with loads connected at $t = 0$s and graph changing at $t = 5$s. There is a little voltage variation at inverters three and four.

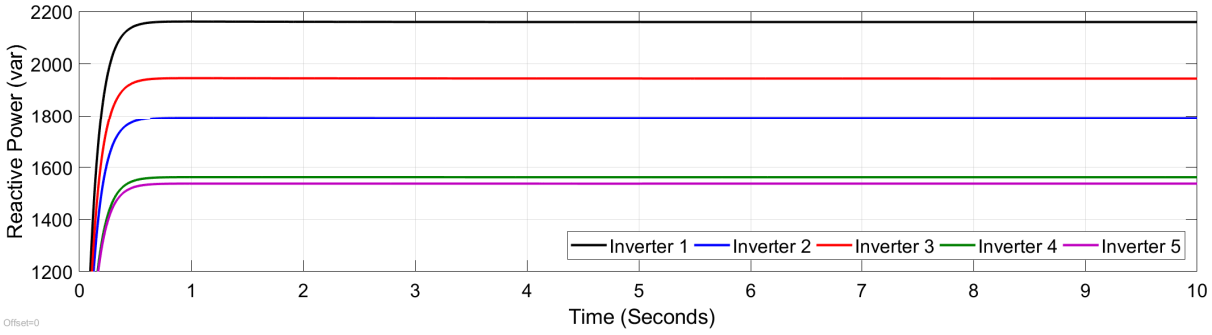

Figure 3.23 – Reactive power at each inverter with loads connected at $t = 0$s and graph changing at $t = 5$s using the distributed ADMM. There is no variation in the reactive power due to the filtering effect of droop control.

After a change in the communication graph, each inverter adjusts its control signal. The RMS voltage is shown in Figure 3.22 each inverter moves closer to the reference value, which is only reached by inverter 1. In this case, the reactive power keeps the same conditions as in other control approaches, as appears in Figure 3.23. Finally, the auxiliary variables and control signals are shown in Figure 3.24, and Figure 3.25, respectively. The variations due to the graph change appear in the signals of inverters three and four.

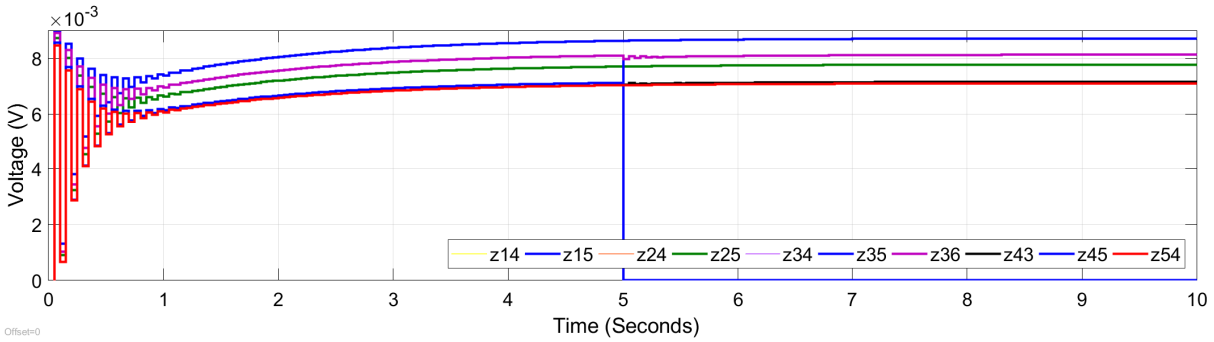

Figure 3.24 – Auxiliary variables z_{ij} for the distributed ADMM with graph changing at $t = 5s$, and loads connected at $t = 0$s. The auxiliary variables $z34$ and $z43$ drop to zero when the graph changes.

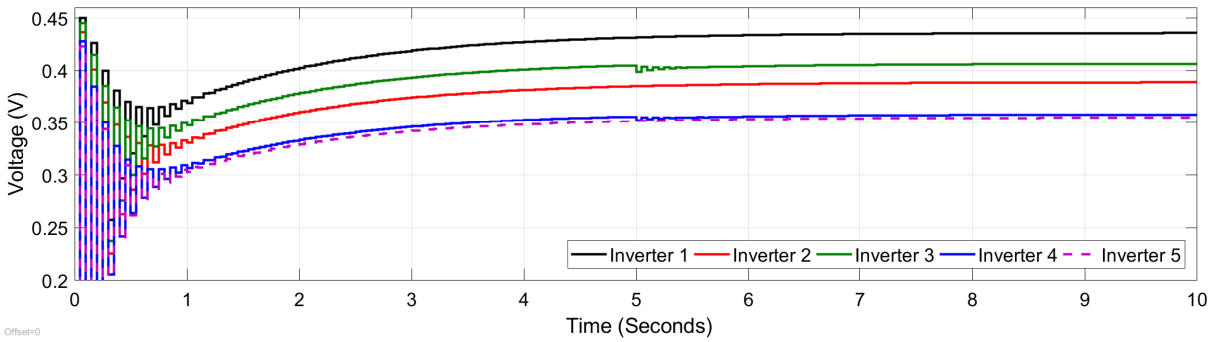

Figure 3.25 – Control signals at each inverter using the distributed ADMM algorithm with graph changing at $t = 5$s. There is a small variation at inverters three and four due to the graph variation.

3.5 Conclusions of the Chapter

In this chapter, we presented three strategies to deal with the problem of distributed MPC control using a Koopman-based linear predictor: non-cooperative, cooperative, and distributed ADMM. The three algorithms can be applied to a wide variety of distributed problems, enhancing control design and using the Koopman-based linear predictors. The first algorithm sets a consensus term that deals with coupling among agents, and it is focused only on measurements without any knowledge of the other control signals and state matrices. The algorithm has a good speed of convergence and a low computational cost. However, in highly coupled systems, the inputs of other agents can affect their performance. In the second case, we propose a cooperative MPC algorithm based on the Koopman linear predictor; it uses the inputs of the neighbors to determine the optimal control signal. It requires calculating the input coupling matrix for the set of neighbors of each agent. We proposed a mode to determine the coupling matrix using the Koopman operator with a centralized method. We show that the proposed algorithm converges to the reference values, but the algorithm has a larger computational cost when compared to the non-cooperative one. The third algorithm uses the representation of the error in the Koopman space to design an MPC that is solved using distributed ADMM. This algorithm reduces the computational cost and is suitable for applications with limited hardware resources, such as MGs. The Koopman-based linear predictor facilitates the design. Finally, the convergence of the three algorithms depends on the weight matrices and parameter selection. The simulation results show that the MPC controller using the three approaches with the Koopman predictor regulates the voltage in the MG for load changes, transmission line changes, and communication graph changes.

Chapter 4

Online data-driven design and improved MPC

4.1 Introduction

Data-driven design and analysis usually work offline due to their inherited use of large quantities of data. In practical applications, such as control design, it is desirable to be able to tune the control parameters by using data from the system. Data-driven techniques such as reinforcement learning and dynamic programming are based on the reward concept, in which the learning process requires storing large quantities of data. In contrast, finding the approximated Koopman algorithm allows reducing the number of dimensions to represent the dynamics of a nonlinear system [34]. In practical systems with low computational capacities, the learning should be based on a limited data storage that allows updating the approximated Koopman operator in a few optimization steps.

Determining the Koopman matrix approximation by data collected from the system involves dealing with noisy measurements. Noise has the effect of shifting the eigenvalues of the Koopman matrix, deteriorating or making the approximation unfeasible. Some approaches have been proposed to deal with it. [60] proposes a second set of observables to determine the effect of noise in the Koopman operator calculation using EDMD. [61] uses a statistical approach, including a 1-norm, to calculate the approximation with EDMD. In the case of power systems, measurements are rich in noise and other phenomena, such as offset. Measurements can be enriched by shifting them together with the proposed approaches presented before [89]. The proper determination of the Koopman approximation enables the design of MPC controllers, including noise rejection.

In this chapter, we present a data-driven online algorithm to regulate the voltage of the MG. The algorithm updates the Koopman matrix based on EDMD using the last measurements gathered to deal with the online requirements, reducing the computational effort significantly. To complement this work, we applied a robust process to determine

the Koopman matrix using noisy measurements. We also include a perturbation term in the Koopman identification, to include the effect of noise.

4.2 Online Data-Driven Design

The identification of systems with nonlinear behavior or systems whose behavior is difficult to model is a paramount aspect of the correct control design. In many cases, it is desirable to be able to model the system online using only measurements from sensors. Data-driven techniques, such as DMD and EDMD, generate a linear representation of the system using the data available. DMD only uses direct measurements, while EDMD allows using a dictionary of nonlinear functions to represent the system. Some algorithms have been proposed to update the representation of a system online using DMD that can be extended to EDMD, enhancing the option of using a wide set of nonlinear functions [59].

Five online DMD algorithms are compared: batch, minibatch, streaming, online, and windowed. The updating of matrix $\mathcal{A}$ needs to fulfill some requirements to guarantee that its eigenvalues move inside the desired region [59]. Figure 4.1 shows a general schematic of how the online EDMD identification works.

We use the next set of observables for the EDMD algorithm

$$\Psi = [V_i \quad V_j \quad V_i^2 \quad V_j^2 \quad V_i V_j]^\top. \tag{4.1}$$

Here we emphasize the online algorithm for the MG's identification and control.

4.2.1 Online EDMD

In this algorithm, the matrix A_{k+1} is updated when a new pair of measurements x_{k+1}, y_{k+1} is available, assuming that the matrix A_{k+1} is relatively similar to A_k. As the A_k matrix is given by

$$A_k = Y_k X_k^\dagger$$

where the pseudo-inverse is calculated as

$$X_k^\dagger = X_k^\top (X_k X_k^\top)^{-1},$$

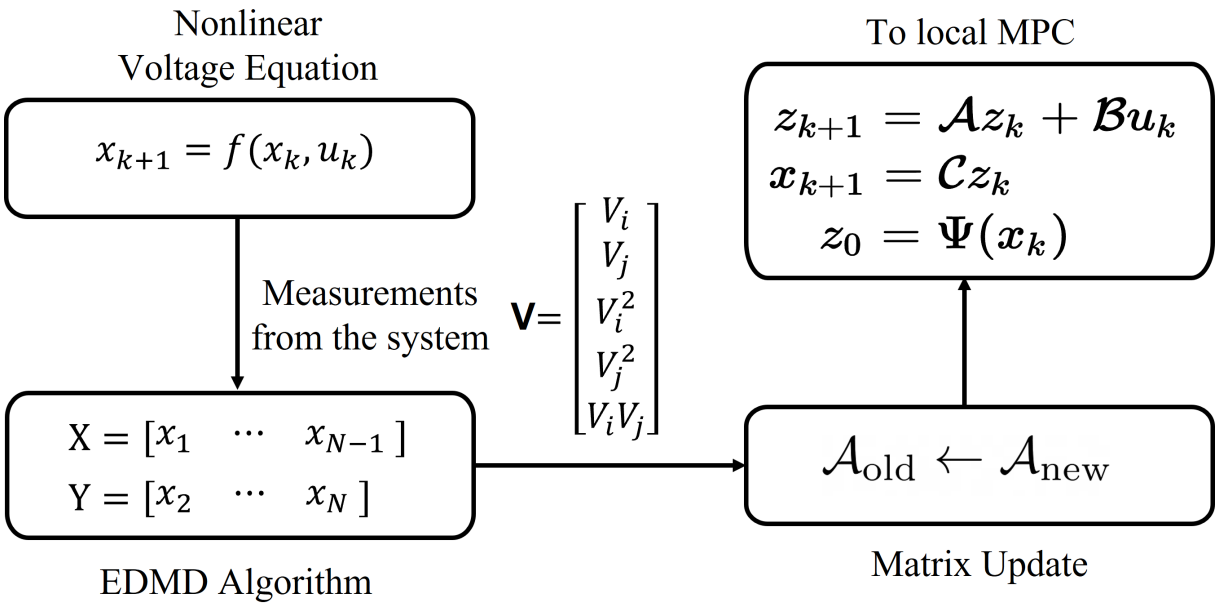

Figure 4.1 – Online EDMD matrix updated scheme, the EDMD uses nonlinear bases to approximate the system matrix $\mathcal{A}$.

assuming that the inverse of the matrix $X_k X_k^\top$ exists. The matrix A_k is rewritten as

$$A_k = Y_k X_k^\top (X_k X^\top X_k)^{-1} = Q_k P_k$$

$$Q_K = Y_k X_k^T$$

$$P_k = (X_k X_k^\top)^{-1}.$$

Then, as defined by [59], the matrix P_k is invertible, well-defined, and positive definite. The update of matrices Q_k and P_k at $k+1$ is given as follows

$$Q_{k+1} = Y_{k+1} X_{k+1}^\top = [Y_k \quad y_{k+1}][X_k \quad x_{k+1}]^\top = Y_k X_k^\top + y_{k+1} x_{k+1}^\top$$

$$P_{k+1}^{-1} = X_{k+1} X_{k+1}^\top = [X_k \quad x_{k+1}][X_k \quad x_{k+1}]^\top = X_k X_k^\top + x_{k+1} x_{k+1}^\top$$

The matrices Q_{k+1}, and P_{k+1} are updated by adding a column form by the new available data x_{k+1} and y_{k+1} of the form

$$Q_{k+1} = Q_k + y_{k+1} x_{k+1}^\top,$$

$$P_{k+1}^{-1} = P_k^{-1} + x_{k+1}x_{k+1}^\top.$$

The updated state matrix A can be obtained by

$$A_{k+1} = Q_{k+1}P_{k+1} = (Q_k + y_{k+1}x_{k+1}^\top)(P_k^{-1} + x_{k+1}x_{k+1}^\top)^{-1},$$

this formula implies calculating two inverse matrices and one matrix product, which has a computational cost of $\mathcal{O}(n^3)$ that is very expensive [59].

The last algorithm is improved to reduce the computational cost for a matrix of the form $A + uv^\top$ with u and v vectors using the following formula [90]

$$(A + uv^\top)^{-1} = A^{-1} - \frac{A^{-1}uv^\top A^{-1}}{1 + v^\top A^{-1}u}.$$

This recursive least-square formula is applied to P_{k+1} and A_{k+1}, generating the following expressions for the online EDMD algorithm [59]

$$P_{k+1} = (P_k^{-1} + x_{k+1}x_{k+1}^\top)^{-1} = P_k - \gamma_{k+1}P_k x_{k+1}x_{k+1}^\top P_k,$$

where

$$\gamma_{k+1} = \frac{1}{1 + x_{k+1}^\top P_k x_{k+1}}.$$

$$A_{k+1} = A_k + \gamma_{k+1}(y_{k+1} - A_k x_{k+1})x_{k+1}^\top P_k.$$

This improved algorithm has a total computational cost of $4n^2$ and does not need to store the complete set of data generated [59]. It is possible to initialize the algorithm using an already stored set of measurements or by starting with an identity matrix.

4.2.2 Online EDMD and Microgrid Design

The control of the MG is based on the EDMD representation of the system. It requires updating the Koopman-based representation of the system by using measurements for a defined period. Following the hierarchical control scheme for MGs and the droop-control low-pass filter time, which is in the order of cents of milliseconds, the system matrix should be updated at least at the time constant given by the low-pass filter.

The online design for the MG should consider the following aspects

— The MGs start with a set of matrices A_0, B_0, and C_0, determined by previous

experiments, matrix A_0 can be set as the identity matrix $c\mathbb{I}_n$ where $c \in \mathbb{R}^+$

— The update of the matrix should be done when there is a perturbation in the system or each updating period T_u

— Matrices B_0 and C_0 are kept unchanged

— Data signals are filtered and saturated to avoid those values that affect the calculation of the EDMD matrix.

The next MPC controller is proposed using an online structure with the state matrix $\mathcal{A}_0$

$$\min_u \sum_{k=1}^{H_p} ||v^{ref} - y_k||_Q^2 + ||u_k||_R^2 \tag{4.2}$$

$$\text{s.t.} \quad z_{k+1} = \mathcal{A}z_k + \mathcal{B}u_k \tag{4.3}$$

$$y_k = \mathcal{C}z_k \tag{4.4}$$

with matrix $\mathcal{A}$ updated each multiple of the sampling time

$$A_{k+1} = A_k + \gamma_{k+1}(y_{k+1} - A_k x_{k+1})x_{k+1}^\top P_k, \tag{4.5}$$

$$\gamma_{k+1} = \frac{1}{1 + x_{k+1}^\top P_k x_{k+1}}. \tag{4.6}$$

Assumption 6 *[91] The set of observables $\psi(x)$ is Lipschitz continuous or local Lipschitz continuous.*

From the set of observables (4.1), functions V_i, and V_j are Lipschitz with Lipschitz constant $L = 1$. The functions V_i^2, and V_j^2 are local Lipschitz on the bounded interval $[V_{min}, V_{max}]$, and the product $V_i V_j$ with bounded functions V_i, and V_j is bounded too and Lipschitz continuous.

Assumption 7 *The matrices that define the initial dynamics of the system $\mathcal{A}_0$, $\mathcal{B}_0$, $\mathcal{C}_0$ are available.*

Assumption 8 *The system defined by the Koopman matrices is controllable for $(\mathcal{A}_0, \mathcal{B}_0)$, and observable for $(\mathcal{A}_0, \mathcal{C}_0)$.*

This condition can be checked directly by determining the observability and controllability of the calculated matrices $\mathcal{A}_0$, $\mathcal{B}_0$, $\mathcal{C}_0$.

Assumption 9 *Each agent has access to local measurements to update the system matrices and that they do not have a delay. The data from states X and input U are sampled from a normal probability distribution.*

Assumption 10 *At each matrix update, the matrix keeps its full rank condition.*

Proposition 1 *The predictive controller with restriction defined by the linear predictor in the lifted-space given by $\mathcal{A}_0$, $\mathcal{B}_0$, $\mathcal{C}_0$ as shown in (4.2), and following the assumptions (7)- (10) converges as $k \to \infty$.*

Proof 4.2.1 *The convex objective function is rewritten as $V = \frac{1}{2}z^\top zQ + \frac{1}{2}u^2R$ with restrictions $z_{k+1} = \mathcal{A}z_k + Bu_k$, and gain matrices $Q > 0$, and $R \geq 0$. Then, we have a sequence of costs function of the form*

$$V_{k+1} = V_k - \frac{1}{2}(z^\top zQ + u^2R),$$

this cost function is bounded and non-increasing along the close-loop trajectory if the system is controllable, then the state cost $z^\top zQ \to 0$, and the input cost $u^2R \to 0$ as $k \to \infty$. The pair $\mathcal{A}_0$, and $\mathcal{B}_0$ is controllable if the matrix $C_o = [B \quad AB \quad \cdots \quad A^{n-1}B]$ is of full rank (n). Matrices $\mathcal{A}_0$, and $\mathcal{B}_0$ has full rank, and the update of matrix $\mathcal{A}_0$ with P_k is also full rank due to $P_k = (X_k X_k^\top)^{-1}$, which guarantee the convergence of the algorithm.

4.2.3 Online EDMD Simulations

Using the MG model from Figure 2.8 whose parameters are shown in Table 4.1, we obtain the next simulation results.

The MG starts with no loads connected at $t = 0$s, then at $t = 5$s, five loads of $2k$var are connected to the nodes 3, 5, 6, 9, 14. Each inverter has stored an initial set of measurements, whose size guarantees that the matrix P_k is invertible. In this case, each inverter has to store a set of 1000 samples of values V_i and V_j, the initial state matrix $A_0 = \mathbb{I}_n$, and matrices $\mathcal{B}_0$, and $\mathcal{C}_0$ are kept constant as calculated by a previous simulation.

The RMS voltage measured at each inverter is shown in Figure 4.2, the state matrix is updated each 0.1s at each inverter, and the data used to determine the Koopman operator is also rewritten at each time sample by the voltage measured. The voltages converge around the reference value with a very small deviation. The reactive power does

Algorithm 5 Online DMD microgrid control

procedure INPUT: u^i, V^i, V^j, matrices $\mathcal{A}_0$, $\mathcal{B}_0$, $\mathcal{C}_0$ ▷ Initial values and matrices
  System Initialization
  Read the value
  Set V^{ref} ▷ Voltage of reference
  Set H_p, T ▷ Prediction Horizon and Sampling Time
  Set Q, R ▷ MPC Gains
  while $k \leq H_p$ **do**
    $\arg\min_u \sum_{k=1}^{H_p} ||y_k^i - V^{ref}||_Q^2 + ||u_k||_R^2$
    s.t.
    $x_{k+1}^i \leftarrow x_k^i + T(\mathcal{A}^i x_k^i + \mathcal{B}^i u_k^i)$
    $y_k^i \leftarrow \mathcal{C}^i x_k^i$
    Select $u^i(1)$ ▷ Select the first optimal value from vector u
    Set x_k and y_k using V^i
    if $kT = T_u$ **then**
      Calculate $P_k = (X_k X_k^\top)^{-1}$
      Calculate $\gamma_{k+1} = \frac{1}{1+x_{k+1}^\top P_k x_{k+1}}$
      Calculate $A_{k+1} = A_k + \gamma_{k+1}(y_{k+1} - A_k x_{k+1}) x_{k+1}^\top P_k$
      Update $\mathcal{A} \leftarrow A$

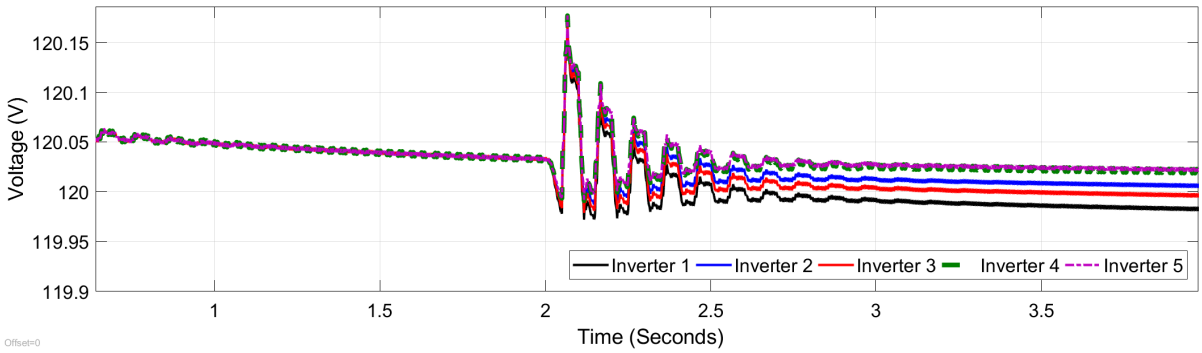

Figure 4.2 – RMS voltage at each inverter after connecting a load at $t = 5$s by using the online EDMM algorithm.

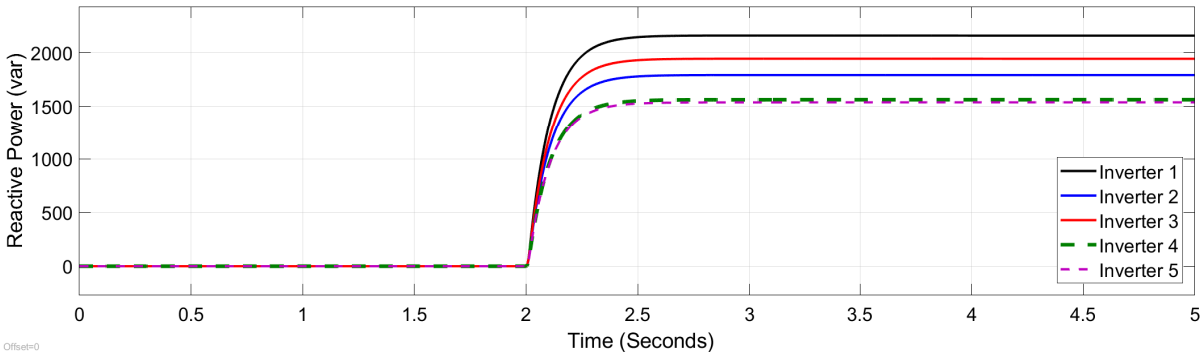

Figure 4.3 – Reactive power at each inverter after connecting the loads at $t = 5$s. There is no variation or oscillations after the load connection.

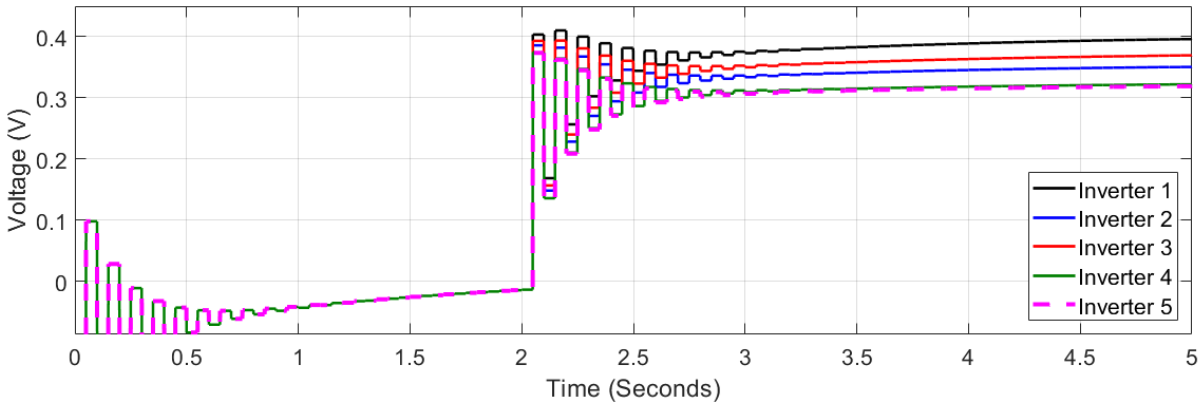

Figure 4.4 – Online control signal at each inverter after the load connection at $t = 5$s using EDMM.

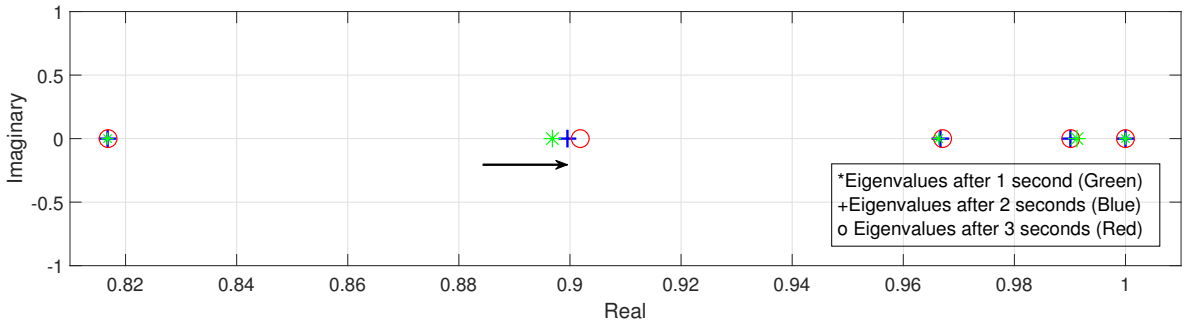

Figure 4.5 – Evolution of the eigenvalues of the state matrix $\mathcal{A}$, starting with an identity matrix, the eigenvalues approximate to the limit value represented by red circles in the picture.

Inverter	1	2	3	4	5
P^{ref} (kW)	10	10	10	10	10
Q^{ref} (kVar)	5	5	5	8	8
m_i (1×10^{-4})	1	1	1	1	1
n_i (1×10^{-4})	0.200	0.200	0.200	0.125	0.125
$V^{ref}(RMS)$	120	120	120	120	120
τ^i(s)	0.10	0.15	0.20	0.25	0.28
f(Hz)	60	60	60	60	60
Updating time t(s)	0.1	0.1	0.1	0.1	0.1

Table 4.1 – Microgrid parameters for control with online EDMD.

not show a particular change as shown in Figure 4.3. The control signals show some small oscillations at $t = 0$s as shown in Figure 4.4. Finally, the evolution of the eigenvalue is shown in Figure 4.5, at the beginning, the eigenvalues are at zero and one due to the initial identity matrix, then they move toward a limit value.

4.3 MPC Algorithm with Disturbances and Noise Rejection

The proper dynamic system identification and control using data techniques have to deal with noise and disturbances. During the identification process, noise measurements affect the correct determination of the system eigenvalues. For the control process, noisy environments affect the controller performance, requiring proper design for noise rejection. For online data-driven control, measurements from the system should be processed before being used by the controller. Regularization is one option to calculate the representation of a system using measurements, and MPC with noise rejection is an option for control design.

4.3.1 Determination of the Koopman Operator with Noisy Measurements

Noisy measurements affect the calculation of the Koopman matrix by shifting the eigenvalues of $\mathcal{A}$ or introducing oscillatory terms. This problem is analyzed by using data techniques such as regularization with improved algorithms to calculate the pseudo-inverse

of data-vectors, such as Lasso [92].

From a set of data points x_k with perturbation norm-bounded deterministic perturbation,

$$\delta x_k = x_k + \delta, \quad \delta \in \Delta,$$

where Δ is the uncertainty set.

The algorithm proposed by [60] to find the proper Koopman representation of a system with noisy measurements needs two sets of dictionary functions denoted by f, g, and n available measurements of the system. First, three vectors of measurements are constructed as follows

$$X = [f(0) \quad f(1) \quad \dots \quad f(n-1)], \tag{4.7}$$

$$Y = [f(1) \quad f(2) \quad \dots \quad f(n)], \tag{4.8}$$

$$Z = [g(0) \quad g(1) \quad \dots \quad g(n-1)]. \tag{4.9}$$

Second, the next matrices are calculated

$$G_0 = \frac{1}{n} X Z^\top \qquad G_1 = \frac{1}{n} Y Z^\top, \tag{4.10}$$

and finally the A matrix is given by

$$A = G_1 G_0^\dagger. \tag{4.11}$$

The approximate representation of matrix A is found using the algorithm proposed by [60]. Next, we proposed a set of observables to calculate the Koopman representation for the MG with noisy measurements.

4.3.2 Effect of Noise Over the Voltage Measurements

To check the effect of noisy measurements on the eigenvalues of the Koopman representation of the MG, we applied (4.10) and (4.11). The two sets of basis that we proposed to find the Koopman approximation are

$$f = [V_i \quad V_j \quad V_i^2 \quad V_j^2 \quad V_i V_j]^\top, \tag{4.12}$$

$$g = [e^{\alpha_1(V_i - V_j)} \quad e^{\alpha_2(V_i - V_j)} \quad e^{\alpha_3(V_i - V_j)} \quad e^{\alpha_4(V_i - V_j)} \quad e^{\alpha_5(V_i - V_j)}]^\top, \tag{4.13}$$

where $\alpha_1 = 0.02$, $\alpha_2 = 0.04$, $\alpha_3 = 0.08$, $\alpha_4 = 0.12$, $\alpha_5 = 0.15$, are weight factors.

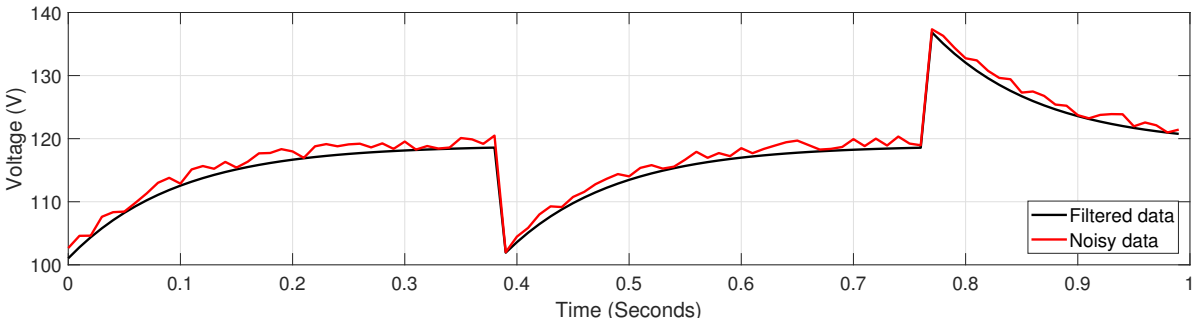

Figure 4.6 – Voltage signal for inverter one for simulation with ideal waveform and added noise.

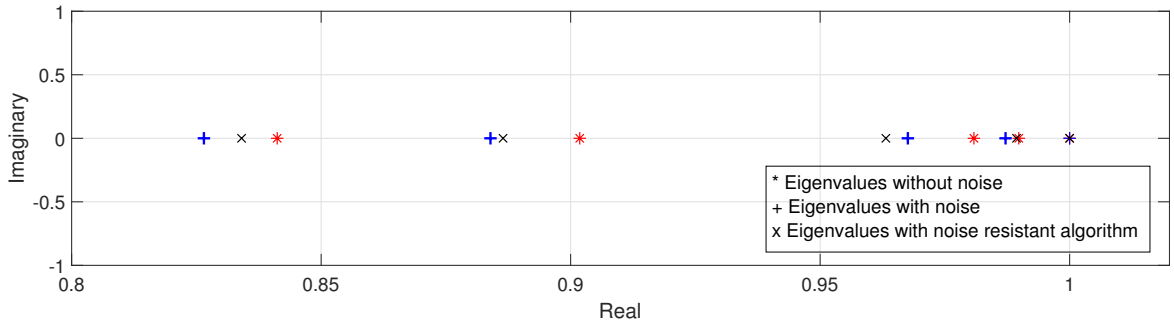

Figure 4.7 – Eigenvalues for data without noise (red), with noise added (blue), and using the algorithm for noisy measurements.

We used a set of voltage measurements without noise and a second set by adding noise with uniform distribution with a magnitude between the interval $[-1, 1]$ to compare the effect of noise over the determination of the state matrix $\mathcal{A}$ and its eigenvalues. Both signals are shown in Figure 4.6. The eigenvalues for the three cases: pure signal, signal with noise, and those found using the algorithm are shown in Figure 4.7. The algorithm approximates better the smaller eigenvalues of the system without noise. Finding a good approximation of the real eigenvalues relies on using a proper set of different basis functions, which is not always easy.

4.3.3 Algorithm with Noise Rejection

A nonlinear system with noise or perturbation represented by w can be written in the linear form using a Koopman-based linear predictor as follows

$$z_{k+1} = \mathcal{A}z_k + \mathcal{B}u_k + \mathcal{D}w_k \qquad y_{k+1} = \mathcal{C}z_k, \tag{4.14}$$

where $\mathcal{D}$ is the Koopman matrix for the perturbation term [57]. The Koopman matrices are obtained using EDMD by solving the least-squares problem

$$\min_{\mathcal{A},\mathcal{B},\mathcal{D}} ||\mathbf{Y} - \mathcal{A}\mathbf{x} - \mathcal{B}\mathbf{U} - \mathcal{D}\mathbf{w}||_F,$$

where $\mathbf{x}$ is the vector of measurements evaluated in the dictionary of functions, and $\mathbf{w}$ is the vector of perturbations measured.

Control Designing for the MG with Perturbations We set the dynamic model of the MG for 1000 seconds, varying the initial conditions to generate the set of trajectories. Then, we use a perturbation signal with uniform perturbation over the range $[-1, 1]$.

$$\min_u ||v^{ref} - y_k||_Q^2 + ||u_k||_R^2 \tag{4.15}$$

$$\text{s.t.} \quad z_{k+1} = \mathcal{A}z_k + \mathcal{B}u_k + \mathcal{D}w_k \tag{4.16}$$

$$y_k = \mathcal{C}z_k \tag{4.17}$$

$$u_{min} \leq u_k \leq u_{max} \tag{4.18}$$

The diagram of Figure 4.8 shows the steps to deal with systems with noise. For systems whose measurements are affected by noise, the algorithm proposed by [60] allows identifying a better approximation of the system eigenvalues by running several experiments using a different set of functions or shifted data using a Hankel matrix. Also, the Koopman identification allows including the matrix $\mathcal{D}$ that models the effect of perturbations or noise over the system.

4.3.4 Simulation of the System with Perturbations

We simulate a MG with 14 nodes with perturbations at inverters 1, 3, and 4 at 1s, 2s, and 3s, respectively. The MG starts with a 5kvar load connected, the sampling time is $T_s = 0.1$s, and the gains $Q = 10$, and $R = 1$, with a prediction horizon of $H_p = 10$.

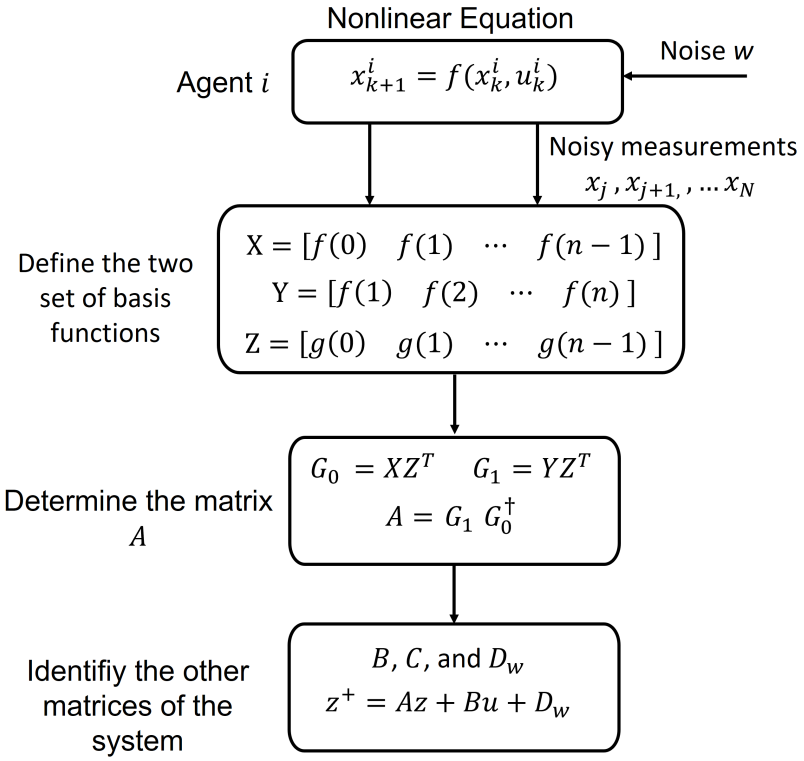

Figure 4.8 – Identification of systems with noisy measurements using EDMD and determination of the perturbation term for the linear predictor.

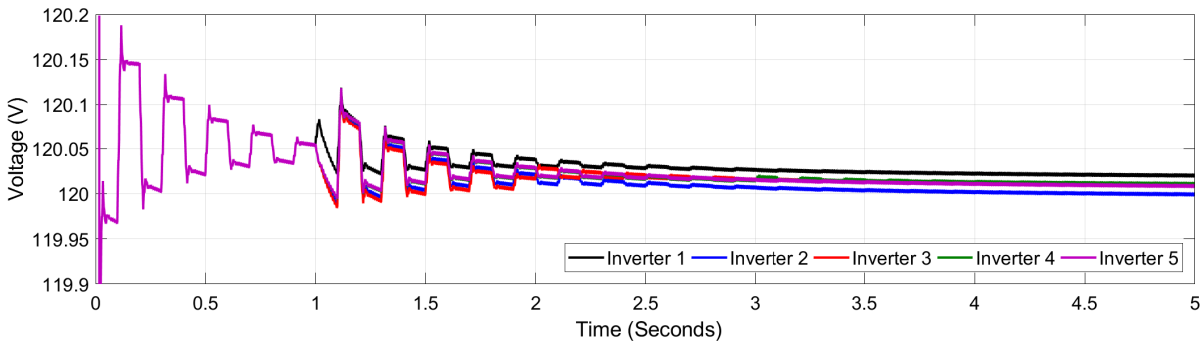

Figure 4.9 – RMS voltage at each inverter when there are disturbances.

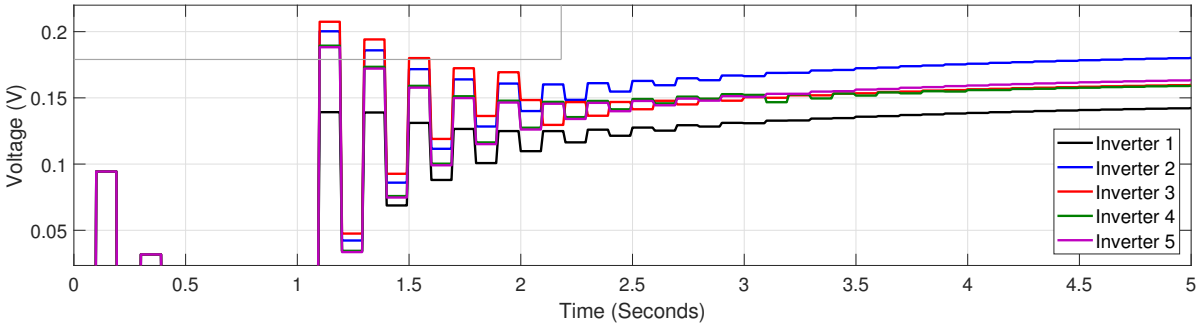

Figure 4.10 – Control signals from each inverter when there are disturbances.

Figure 4.9 shows the voltage measured at each inverter, where can be seen the three perturbations at inverters 1, 3, and 4. The control overcomes this constant perturbation while following the reference value. The control signals for each inverter are shown in Figure 4.10.

4.4 Conclusions of the Chapter

In this chapter, we presented a Koopman-based data-driven online controller for the MG, and an algorithm to deal with noise in the identification and control using the Koopman operator. The online problem updates the state matrix using the measurements available at each sampling time and EDMM. The convergence of the algorithm is determined by the full rank matrix condition.

For noise rejection, the problem is divided into two stages: the first stage uses an additional set of observables to determine the true eigenvalues of a system whose measurements are affected by noise. The second stage introduces a linear term in the state representation to include the effect of noise and perturbations over the system. The algorithm controls the system even when it is affected by noise, where the paramount problem is the detection of the true eigenvalues of the system. This problem requires several experiments when working with data from sensors or real experiments; in the case of online Koopman identification, the data measurements should be treated by filtering to avoid high-frequency components.

Conclusions and perspectives

This document focuses on the identification and control of voltage in electrical microgrids. Proper identification of power systems can be a challenging problem due to their networked structure, heterogeneity, different time scales, and uncertainties. The availability of large amounts of data from the system enables the use of data-driven techniques to improve the identification and control of interconnected systems. We have considered different approaches to the MG control view as a distributed system. This document presents several contributions to the control of networked systems, and in particular, of MGs.

The first chapter presented the state-of-the-art of data-driven techniques and the problem of controlling the MG, focused on voltage regulation and reactive power compensation by using a high-inductive transmission line at the inverter's output. Among several techniques, that exploits data for identification and control of dynamic systems, we focus on the Koopman operator and their data-driven approximations due to their capacity to generate a linear representation of nonlinear systems, and to define a set of observables to approximate systems.

In the second chapter, we present a decentralized data-driven control using Koopman operator theory to regulate the voltage of the MG using optimization techniques. The proposed algorithm requires that each generator uses local online measurement to generate the control actions. We design a local linear predictor whose eigenvalues are inside the unit circle that guarantee the algorithm convergence. The linear predictor also reduces the computational time compared with other controllers that proposed a nonlinear MPC. The proposed algorithm is susceptible to changes in the transmission lines, especially in the local inductance, which also requires recalculating the Koopman representation. Also, the algorithm omits some networked systems behaviors, such as the effect of the control actions among agents.

In the third chapter, we presented a set of algorithms to deal with distributed systems using the Koopman-based linear predictor and MPC: non-cooperative, cooperative, and distributed ADMM. The non-cooperative algorithm uses only voltage measurements from other generators to regulate its voltage by using a linear consensus term. We demon-

strated that the system converges to the reference value. For the cooperative MPC, we proposed an offline algorithm to obtain the input matrix of the generators by varying the control signal and measuring its effect on the other generators. We proposed a third algorithm that uses the representation of the error in the Koopman space to design an MPC that is solved using a distributed ADMM. It reduces the computation time and improves the computational cost. The three algorithms converge to the reference values, depending on the weight matrices and parameter selection. We also used a Koopman-based linear predictor that includes a disturbance matrix. The simulation shows that the MPC controller, using the Koopman representation, regulates the voltage in the MG for load changes, transmission line changes, and communication graph changes. The proper identification of the dynamic of each generator depends on the correct treatment of the data available in this first case, rapid variations such as high-frequency components might affect the correct calculation of the Koopman representation. A low-pass filter was used to avoid fast variations in the voltage.

In the fourth chapter, we use a proposed algorithm to deal with the problem of noisy measurements to determine the Koopman representation of the system. These tools are a key factor for the online Koopman algorithm design that we proposed to regulate the voltage in the MG. The problem of updating the Koopman matrix is solved by using an algorithm that uses only a pair of measurements at each iteration. We have shown the proposed algorithm converges by following a set of restrictions related to the data measured, and with the time set to do the actualization. Finally, an algorithm that uses a term for perturbations identified by EDMD is presented and used to design a MPC capable of rejecting voltage perturbations.

The future perspective of this work is related to the improvement of the online algorithm and to the problem of working with noisy measurements. Another aspect is dealing with non-lossless transmission lines, which should include the phase angle between voltage and current. The work with the Koopman operator in distributed systems can also address a proper identification of the nodes with more interaction with other agents in a coupled network, such as the power system.

BIBLIOGRAPHY

[1] Jean and D. Pitteloud, "Global Wind Power Statistics – Wind Energy International Platform," 2019.

[2] International Energy Agency, "Solar PV – Analysis - IEA," 2020.

[3] F. Milano, F. Dorfler, G. Hug, D. J. Hill, and G. Verbič, "Foundations and challenges of low-inertia systems (Invited Paper)," *20th Power Systems Computation Conference, PSCC 2018*, 2018.

[4] S. Akkari, E. Prieto-Araujo, J. Dai, O. Gomis-Bellmunt, and X. Guillaud, "Impact of the dc cable models on the svd analysis of a multi-terminal hvdc system," in *2016 Power Systems Computation Conference (PSCC)*, pp. 1–6, IEEE, 2016.

[5] M. Budišić, R. Mohr, and I. Mezić, "Applied koopmanism," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 22, no. 4, p. 047510, 2012.

[6] Y. Susuki and A. Chakrabortty, "Introduction to Koopman Mode Decomposition for Data-Based Technology of Power System Nonlinear Dynamics," *IFAC-PapersOnLine*, vol. 51, no. 28, pp. 327–332, 2018.

[7] V. T. Priyanga, M. Chandni, N. Mohan, and K. P. Soman, "Data-driven analysis for low frequency oscillation identification in smart grid using FB-DMD and T-DMD methods," *Proceedings of the 2019 9th International Conference on Advances in Computing and Communication, ICACC 2019*, pp. 51–57, 2019.

[8] F. Raak, Y. Susuki, and T. Hikihara, "Multi-Way Partitioning of Power Networks via Koopman Mode Analysis," *IFAC-PapersOnLine*, vol. 48, no. 30, pp. 421–426, 2015.

[9] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Sparse identification of nonlinear dynamics with control (sindyc)," *IFAC-PapersOnLine*, vol. 49, no. 18, pp. 710–715, 2016.

[10] M. Korda and I. Mezić, "Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control," *Automatica*, vol. 93, pp. 149–160, 2018.

[11] M. Korda, Y. Susuki, and I. Mezić, "Power grid transient stabilization using koopman model predictive control," *IFAC-PapersOnLine*, vol. 51, no. 28, pp. 297–302, 2018.

[12] T. Yang, X. Yi, J. Wu, Y. Yuan, D. Wu, Z. Meng, Y. Hong, H. Wang, Z. Lin, and K. H. Johansson, "A survey of distributed optimization," *Annual Reviews in Control*, vol. 47, no. xxxx, pp. 278–305, 2019.

[13] R. H. Lasseter, "MicroGrids," *IEEE Power Engineering Society winter Meeting*, pp. 305–308, 2002.

[14] J. J. Justo, F. Mwasilu, J. Lee, and J. W. Jung, "AC-microgrids versus DC-microgrids with distributed energy resources: A review," *Renewable and Sustainable Energy Reviews*, vol. 24, pp. 387–405, 2013.

[15] A. Tayyebi, F. Dörfler, F. Kupzog, Z. Miletic, and W. Hribernik, "Grid-Forming Converters - Inevitability , Control Strategies and Challenges in Future Grid Applications and Challenges in Future Grids Application," *CIRED Workshop 2018*, no. 0236, pp. 1–5, 2018.

[16] S. Sen and V. Kumar, "Microgrid control: A comprehensive survey," *Annual Reviews in Control*, vol. 45, no. March, pp. 118–151, 2018.

[17] J. Schiffer, D. Zonetti, R. Ortega, A. M. Stanković, T. Sezi, and J. Raisch, "A survey on modeling of microgrids—From fundamental physics to phasors and voltage sources," *Automatica*, vol. 74, pp. 135–150, 2016.

[18] S. Moayedi and A. Davoudi, "Distributed Tertiary Control of DC Microgrid Clusters," *IEEE Transactions on Power Electronics*, vol. 31, no. 2, pp. 1717–1733, 2016.

[19] F. Dörfler, J. W. Simpson-Porco, and F. Bullo, "Breaking the hierarchy: Distributed control and economic optimality in microgrids," *IEEE Transactions on Control of Network Systems*, vol. 3, no. 3, pp. 241–253, 2015.

[20] Z. S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Information Sciences*, vol. 235, pp. 3–35, 2013.

[21] S. Yin, X. Li, H. Gao, and O. Kaynak, "Data-based techniques focused on modern industry: An overview," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 1, pp. 657–667, 2015.

[22] Z. Hou and S. Xiong, "On model-free adaptive control and its stability analysis," *IEEE Transactions on Automatic Control*, vol. 64, no. 11, pp. 4555–4569, 2019.

[23] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022.

[24] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits and Systems Magazine*, vol. 9, no. 3, pp. 32–50, 2009.

[25] L. Buşoniu, T. de Bruin, D. Tolić, J. Kober, and I. Palunko, "Reinforcement learning for control: Performance, stability, and deep approximators," *Annual Reviews in Control*, vol. 46, pp. 8–28, 2018.

[26] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis, "Optimal and Autonomous Control Using Reinforcement Learning: A Survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2042–2062, 2018.

[27] L. Buşoniu, R. Babuška, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 38, no. 2, pp. 156–172, 2008.

[28] M. L. Li, S. Chen, and J. Chen, "Adaptive Learning: A New Decentralized Reinforcement Learning Approach for Cooperative Multiagent Systems," *IEEE Access*, vol. 8, pp. 99404–99421, 2020.

[29] F. Dietrich, T. N. Thiem, and I. G. Kevrekidis, "On the koopman operator of algorithms," *SIAM Journal on Applied Dynamical Systems*, vol. 19, no. 2, pp. 860–885, 2020.

[30] Z. Ping, Z. Yin, X. Li, Y. Liu, and T. Yang, "Deep Koopman model predictive control for enhancing transient stability in power grids," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 6, pp. 1964–1978, 2021.

[31] C. W. Rowley, I. Mezi, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," *Journal of Fluid Mechanics*, vol. 641, pp. 115–127, 2009.

[32] M. Haseli and J. Cortés, "Approximating the Koopman operator using noisy data: Noise-resilient extended dynamic mode decomposition," *Proceedings of the American Control Conference*, vol. 2019-July, pp. 5499–5504, 2019.

[33] S. Peitz and S. Klus, "Koopman operator-based model reduction for switched-system control of PDEs," *Automatica*, vol. 106, pp. 184–191, 2019.

[34] S. Klus, F. Nüske, P. Koltai, H. Wu, I. Kevrekidis, C. Schütte, and F. Noé, "Data-Driven Model Reduction and Transfer Operator Approximation," *Journal of Nonlinear Science*, vol. 28, no. 3, pp. 985–1010, 2018.

[35] J. N. Kutz, X. Fu, and S. L. Brunton, "Multiresolution dynamic mode decomposition," *SIAM Journal on Applied Dynamical Systems*, vol. 15, no. 2, pp. 713–735, 2016.

[36] A. Mauroy, Y. Susuki, and I. Mezić, "Introduction to the koopman operator in dynamical systems and control theory," in *The Koopman Operator in Systems and Control*, vol. 484, pp. 3–33, Springer, 2020.

[37] Y. Susuki and I. Mezić, "A prony approximation of Koopman Mode Decomposition," *Proceedings of the IEEE Conference on Decision and Control*, vol. 54rd IEEE Conference on Decision and Control,CDC 2015, no. Cdc, pp. 7022–7027, 2015.

[38] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A Data–Driven Approximation of the Koopman Operator: Extending Dynamic Mode Decomposition," *Journal of Nonlinear Science*, vol. 25, no. 6, pp. 1307–1346, 2015.

[39] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *Journal of Fluid Mechanics*, vol. 656, pp. 5–28, 2010.

[40] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz, "On dynamic mode decomposition: Theory and applications," *Journal of Computational Dynamics*, vol. 1, no. 2, pp. 391–421, 2014.

[41] I. Mezic, "Spectral properties of dynamical systems, model reduction and decompositions," *Nonlinear Dynamics*, vol. 41, pp. 309–325, 2005.

[42] J. L. Proctor, S. L. Brunton, and J. N. Kutz, "Dynamic mode decomposition with control," *SIAM Journal on Applied Dynamical Systems*, vol. 15, no. 1, pp. 142–161, 2016.

[43] J. Mann and J. N. Kutz, "Dynamic mode decomposition for financial trading strategies," *Quantitative Finance*, vol. 16, no. 11, pp. 1643–1655, 2016.

[44] N. B. Erichson, S. L. Brunton, and J. N. Kutz, "Compressed dynamic mode decomposition for background modeling," *Journal of Real-Time Image Processing*, vol. 16, no. 5, pp. 1479–1492, 2019.

[45] A. E. Saldana, E. Barocio, A. R. Messina, J. J. Ramos, R. J. Segundo, and G. A. Tinajero, "Monitoring harmonic distortion in microgrids using dynamic mode decomposition," *IEEE Power and Energy Society General Meeting*, vol. 2018-Janua, pp. 1–5, 2018.

[46] E. Kaiser, J. N. Kutz, and S. Brunton, "Data-driven discovery of Koopman eigenfunctions for control," *Machine Learning: Science and Technology*, vol. 98195, pp. 1–40, 2021.

[47] M. Korda and I. Mezic, "Optimal Construction of Koopman Eigenfunctions for Prediction and Control," *IEEE Transactions on Automatic Control*, vol. 65, no. 12, pp. 5114–5129, 2020.

[48] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[49] D. P. Bertsekas and J. N. Tsitsiklis, "Comments on "Coordination of groups of mobile autonomous agents using nearest neighbor rules"," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 968–969, 2007.

[50] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative Control of Multi-Agent Systems*. Communications and Control Engineering, London: Springer London, 2014.

[51] A. Mohammed, S. S. Refaat, S. Bayhan, and H. Abu-Rub, "Ac microgrid control and management strategies: Evaluation and review," *IEEE Power Electronics Magazine*, vol. 6, no. 2, pp. 18–31, 2019.

[52] D. K. Molzahn, F. Dörfler, H. Sandberg, S. H. Low, S. Chakrabarti, R. Baldick, and J. Lavaei, "A Survey of Distributed Optimization and Control Algorithms for Electric Power Systems," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 2941–2962, 2017.

[53] R. Olfati-Saber, J. A. Fax, and R. M. Murray, "Consensus and cooperation in networked multi-agent systems," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 215–233, 2007.

[54] Dörfler, Florian, "Distributed consensus-based optimization review of optimization theory." `https://people.ee.ethz.ch/~floriand/docs/Teaching/ATIC_2017/Optimization_Lecture.pdf`, 2017. Accessed: 2021-02-06.

[55] T. Tatarenko and B. Touri, "Non-Convex Distributed Optimization," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3744–3757, 2017.

[56] W. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality: Second Edition*. United States: Wiley-Blackwell, Sept. 2011.

[57] M. Korda, Y. Susuki, and I. Mezić, "Power grid transient stabilization using Koopman model predictive control," *IFAC-PapersOnLine*, vol. 51, no. 28, pp. 297–302, 2018.

[58] T. Goldstein, B. O'Donoghue, S. Setzep, and R. Baraniuk, "Fast alternating direction optimization methods," *SIAM Journal on Imaging Sciences*, vol. 7, no. 3, pp. 1588–1623, 2014.

[59] H. Zhang, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, "Online dynamic mode decomposition for time-varying systems," *SIAM Journal on Applied Dynamical Systems*, vol. 18, no. 3, pp. 1586–1609, 2019.

[60] M. Wanner and I. Mezić, "Robust approximation of the stochastic koopman operator," *SIAM Journal on Applied Dynamical Systems*, vol. 21, no. 3, pp. 1930–1951, 2022.

[61] S. Sinha, B. Huang, and U. Vaidya, "On Robust Computation of Koopman Operator and Prediction in Random Dynamical Systems," *Journal of Nonlinear Science*, vol. 30, no. 5, pp. 2057–2090, 2020.

[62] P. Tielens and D. Van Hertem, "The relevance of inertia in power systems," *Renewable and Sustainable Energy Reviews*, vol. 55, no. 2016, pp. 999–1009, 2016.

[63] N. Pogaku, M. Prodanović, and T. C. Green, "Modeling, analysis and testing of autonomous operation of an inverter-based microgrid," *IEEE Transactions on Power Electronics*, vol. 22, no. 2, pp. 613–625, 2007.

[64] J. Rocabert, A. Luna, F. Blaabjerg, and P. Rodríguez, "Control of power converters in ac microgrids," *IEEE Transactions on Power Electronics*, vol. 27, no. 11, pp. 4734–4749, 2012.

[65] S. Anderson, P. Hidalgo-Gonzalez, R. Dobbe, and C. J. Tomlin, "Distributed Model Predictive Control for Autonomous Droop-Controlled Inverter-Based Microgrids," *Proceedings of the IEEE Conference on Decision and Control*, vol. 2019-Decem, no. Cdc, pp. 6242–6248, 2019.

[66] Q.-C. Zhong and Y. Zeng, "Universal droop control of inverters with different types of output impedance," *IEEE access*, vol. 4, pp. 702–712, 2016.

[67] Q. Liu, Y. Tao, X. Liu, Y. Deng, and X. He, "Voltage unbalance and harmonics compensation for islanded microgrid inverters," *IET Power Electronics*, vol. 7, no. 5, pp. 1055–1063, 2014.

[68] J. Schiffer, R. Ortega, A. Astolfi, J. Raisch, and T. Sezi, "Conditions for stability of droop-controlled inverter-based microgrids," *Automatica*, vol. 50, no. 10, pp. 2457–2469, 2014.

[69] J. M. Guerrero, J. C. Vasquez, J. Matas, L. G. De Vicuña, and M. Castilla, "Hierarchical control of droop-controlled AC and DC microgrids - A general approach toward standardization," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 1, pp. 158–172, 2011.

[70] J. M. Guerrero, M. Chandorkar, T. Lee, and P. C. Loh, "Advanced Control Architectures for Intelligent Microgrids; Part I: Decentralized and Hierarchical Control," *Industrial Electronics, IEEE Transactions on*, vol. 60, no. 4, pp. 1254–1262, 2013.

[71] J. Hu, Y. Shan, J. M. Guerrero, A. Ioinovici, K. W. Chan, and J. Rodriguez, "Model predictive control of microgrids – An overview," *Renewable and Sustainable Energy Reviews*, vol. 136, no. September 2020, p. 110422, 2021.

[72] R. P. Aguilera and D. E. Quevedo, "Predictive control of power converters: Designs with guaranteed performance," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 1, pp. 53–63, 2015.

[73] G. Qu and N. Li, "Optimal Distributed Feedback Voltage Control under Limited Reactive Power," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 315–331, 2020.

[74] L. M. Leon, A. S. Bretas, and S. Rivera, "Quadratically constrained quadratic programming formulation of contingency constrained optimal power flow with photovoltaic generation," *Energies*, vol. 13, no. 13, 2020.

[75] J. Boudreaux, "Design, Simulation, and Construction of an IEEE 14-Bus Power System," *LSU Master's Theses*, sep 2018.

[76] J. Lofberg, "Yalmip : a toolbox for modeling and optimization in matlab," in *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508)*, pp. 284–289, 2004.

[77] P. D. Christofides, R. Scattolini, D. Muñoz de la Peña, and J. Liu, "Distributed model predictive control: A tutorial review and future research directions," *Computers and Chemical Engineering*, vol. 51, pp. 21–41, 2013.

[78] J. Maestre and R. Negenborn, *Distributed Model Predictive Control Made Easy*, vol. 69 of *Intelligent Systems, Control and Automation: Science and Engineering; Distributed Model Predictive Control Made Easy*. Springer Netherlands, 2014.

[79] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.

[80] P. Šulc, S. Backhaus, and M. Chertkov, "Optimal distributed control of reactive power via the alternating direction method of multipliers," *IEEE Transactions on Energy Conversion*, vol. 29, no. 4, pp. 968–977, 2014.

[81] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal control*. John Wiley & Sons, 2012.

[82] C. Conte, C. N. Jones, M. Morari, and M. N. Zeilinger, "Distributed synthesis and stability of cooperative distributed model predictive control for linear systems," *Automatica*, vol. 69, pp. 117–125, 2016.

[83] M. Chen, J. Zhao, Z. Xu, Y. Liu, Y. Zhu, and Z. Shao, "Cooperative distributed model predictive control based on topological hierarchy decomposition," *Control Engineering Practice*, vol. 103, no. August, 2020.

[84] D. Fu, H.-T. Zhang, A. Dutta, and G. Chen, "A cooperative distributed model predictive control approach to supply chain management," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 12, pp. 4894–4904, 2020.

[85] S. P. Nandanoori, S. Pal, S. Sinha, S. Kundu, K. Agarwal, and S. Choudhury, "Data-driven Distributed Learning of Multi-agent Systems: A Koopman Operator Approach," *Proceedings of the IEEE Conference on Decision and Control*, vol. 2021-December, no. Cdc, pp. 5059–5066, 2021.

[86] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia, "Cooperative distributed model predictive control," *Systems & Control Letters*, vol. 59, no. 8, pp. 460–469, 2010.

[87] N. Bastianello, A. Simonetto, and R. Carli, "Distributed prediction-correction admm for time-varying convex optimization," in *2020 54th Asilomar Conference on Signals, Systems, and Computers*, pp. 47–52, 2020.

[88] E. Wei and A. Ozdaglar, "Distributed alternating direction method of multipliers," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 5445–5450, 2012.

[89] H. Arbabi and I. Mezić, "Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the Koopman operator," *SIAM Journal on Applied Dynamical Systems*, vol. 16, no. 4, pp. 2096–2126, 2017.

[90] W. W. Hager, "Updating the inverse of a matrix," *SIAM Review*, vol. 31, no. 2, pp. 221–239, 1989.

[91] X. Zhang, W. Pan, R. Scattolini, S. Yu, and X. Xu, "Robust tube-based model predictive control with Koopman operators," *Automatica*, vol. 137, p. 110114, 2022.

[92] M. Akinkunmi, *Introduction to Statistical Analysis*, pp. 1–13. Cham: Springer International Publishing, 2019.

APPENDIX

Adjacency matrix for the 14-nodes power system

$$Y_{ij} = \begin{bmatrix}
4.04 & -3.20 & 0.00 & 0.00 & -0.85 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
-3.20 & 4.15 & -0.95 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & -0.95 & 2.06 & -1.11 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & -1.11 & 6.86 & -4.50 & 0.00 & -0.91 & 0.00 & -0.34 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
-0.85 & 0.00 & 0.00 & -4.50 & 6.09 & -0.75 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & -0.75 & 3.90 & 0.00 & 0.00 & 0.00 & 0.00 & -0.95 & -0.74 & -1.46 & 0.00 \\
0.00 & 0.00 & 0.00 & -0.91 & 0.00 & 0.00 & 3.71 & -1.07 & -1.72 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -1.07 & 1.07 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & -0.34 & 0.00 & 0.00 & -1.72 & 0.00 & 5.01 & -2.25 & 0.00 & 0.00 & 0.00 & -0.70 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -2.25 & 3.23 & -0.99 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.95 & 0.00 & 0.00 & 0.00 & -0.99 & 1.94 & 0.00 & 0.00 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.74 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 1.69 & -0.95 & 0.00 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -1.46 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.95 & 2.95 & -0.54 \\
0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.70 & 0.00 & 0.00 & 0.00 & -0.54 & 1.24
\end{bmatrix}$$

State-space matrices

$$A_1 = \begin{bmatrix}
1.0337 & 0.0618 & -0.0004 & -0.0001 & -0.0004 \\
-0.0045 & 1.0155 & 0.0001 & -0.0000 & -0.0002 \\
22.1154 & 0.7293 & 0.8084 & -0.0014 & 0.0025 \\
-2.6088 & 5.2852 & 0.0264 & 0.9843 & -0.0331 \\
-1.8765 & 14.6171 & 0.0224 & -0.0005 & 0.8717
\end{bmatrix} \qquad B_1 = \begin{bmatrix}
0.0732 \\ 0.0085 \\ 17.5105 \\ 2.0848 \\ 9.7445
\end{bmatrix}$$

$$C_1 = \begin{bmatrix}
1.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\
0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000
\end{bmatrix}$$

Coupling B_{ij} matrices

$$B_{12} = \begin{bmatrix}
0.0000 \\ 0.0002 \\ -0.0005 \\ 0.0490 \\ 0.0243
\end{bmatrix} \quad B_{14} = \begin{bmatrix}
0.0000 \\ 0.0001 \\ -0.0006 \\ 0.0124 \\ 0.0059
\end{bmatrix} \quad B_{21} = \begin{bmatrix}
0.0000 \\ 0.0003 \\ -0.0007 \\ 0.0720 \\ 0.0357
\end{bmatrix} \quad B_{23} = \begin{bmatrix}
0.0000 \\ 0.0003 \\ -0.0008 \\ 0.0669 \\ 0.0331
\end{bmatrix}$$

$$B_{32} = \begin{bmatrix} 0.0000 \\ 0.0002 \\ -0.0003 \\ 0.0527 \\ 0.0262 \end{bmatrix} \quad B_{34} = \begin{bmatrix} 0.0000 \\ 0.0001 \\ -0.0003 \\ 0.0119 \\ 0.0058 \end{bmatrix} \quad B_{35} = \begin{bmatrix} 0.0000 \\ 0.0000 \\ -0.0002 \\ 0.0092 \\ 0.0045 \end{bmatrix} \quad B_{41} = \begin{bmatrix} 0.0000 \\ 0.0001 \\ 0.0002 \\ 0.0337 \\ 0.0170 \end{bmatrix}$$

$$B_{43} = \begin{bmatrix} 0.0000 \\ 0.0001 \\ 0.0002 \\ 0.0278 \\ 0.0140 \end{bmatrix} \quad B_{53} = \begin{bmatrix} -0.0000 \\ 0.0000 \\ -0.0000 \\ 0.0119 \\ 0.0060 \end{bmatrix}$$

Error matrices

$$A_1 = \begin{bmatrix} 0.8997 & 0.0034 & -0.0002 & -0.0000 & 0.0000 \\ -0.0211 & 1.0103 & 0.0000 & -0.0001 & 0.0002 \\ 1.2191 & 0.0570 & 0.7852 & -0.0002 & -0.0103 \\ -5.3249 & 2.4602 & 0.0081 & 0.9796 & 0.0464 \\ 0.3297 & 0.3800 & -0.0216 & -0.0030 & 0.8990 \end{bmatrix} \quad B_1 = \begin{bmatrix} -0.0732 \\ 0.0087 \\ -0.0827 \\ 2.0937 \\ -8.7254 \end{bmatrix}$$

$$C_1 = \begin{bmatrix} 1.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

State-space and perturbation matrices

$$A_1 = \begin{bmatrix} 1.0337 & 0.0618 & -0.0004 & -0.0001 & -0.0004 \\ -0.0045 & 1.0155 & 0.0001 & -0.0000 & -0.0002 \\ 22.1154 & 0.7293 & 0.8084 & -0.0014 & 0.0025 \\ -2.6088 & 5.2852 & 0.0264 & 0.9843 & -0.0331 \\ -1.8765 & 14.6171 & 0.0224 & -0.0005 & 0.8717 \end{bmatrix} \quad B_1 = \begin{bmatrix} 0.0732 \\ 0.0085 \\ 17.5105 \\ 2.0848 \\ 9.7445 \end{bmatrix}$$

$$D_1 = \begin{bmatrix} 1.0337 \\ 1.0337 \\ 1.0337 \\ 1.0337 \\ 1.0337 \end{bmatrix} \qquad C_1 = \begin{bmatrix} 1.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}$$

DOCTORAT BRETAGNE LOIRE MATHSTIC

IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Titre : Contrôle piloté par les données des réseaux énergétiques interdépendants

Mot clés : Opérateur de Koopman, micro-réseaux, commande distribuée, prédicteur linéaire, commande prédictive de modèle

Résumé : Cette recherche a proposé plusieurs algorithmes pour l'identification et le commande des micro-réseaux basés sur l'opérateur Koopman. Les contributions présentées dans ce manuscrit se concentrent sur le comman de de la tension et de la puissance réactive. Nous avons considéré cinq scénarios de commande basés sur l'opérateur de Koopman : (i) un algorithme centralisé qui régule la tension du micro-réseau sans partager l'information en utilisant MPC. (ii) une commande distribuée non coopérative, avec un terme de consensus dans les restrictions, qui régule la tension en fonction du modèle de Koopman des onduleurs. (iii) un MPC distribué coopératif qui utilise le modèle Koopman du micro-réseau, où les agents partagent leurs entrées de commande pour générer les signaux de commandes. Ici, nous identifions les matrices d'entrée en utilisant des données. (iv) un commande distribuée qui utilise les données pour identifier l'erreur du système afin de concevoir un algorithme ADMM. (v) une commande en ligne pilotée par les données qui régule la tension du micro-réseau et une analyse des valeurs propres du système et des effets des mesures bruitées.

Title: Data-Driven Control of Interdependent Energy Networks

Keywords: Koopman operator, microgrid, distributed control, linear predictor, model predictive control

Abstract: This research proposed several algorithms for the identification and control of microgrids based on the Koopman operator. The contributions presented in this manuscript are focused on the control of voltage and reactive power. We have considered five control scenarios based on the Koopman operator: (i) a centralized algorithm that regulates the microgrid voltage without sharing information using MPC. (ii) a non-cooperative distributed control, with a consensus term in the restrictions, that regulates the voltage based on the Koopman model of the inverters. (iii) a cooperative distributed MPC that uses the microgrid Koopman model, where the agents share their control inputs to generate the control signals. Here, we identify the input matrices by using data. (iv) a distributed control that uses data to identify the system error to design an ADMM algorithm. (v) an online data-driven controller that regulates the microgrid voltage and an analysis of the eigenvalues of the system and the effects of noisy measurements.