University of Alberta

Coordination Techniques for Distributed Model Predictive Control

by

Padideh Ghafoor Mohseni

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Chemical Engineering

Department of Chemical and Materials Engineering

Fall 2013
Edmonton, Alberta

This thesis is dedicated to ...

my parents,
Simin & Hassan,
and
my husband,
Amir

Abstract

Industrial chemical plants are complex, highly integrated systems composed of geographically distributed processing units, linked together by material and energy streams. To ensure efficient operation in such integrated plants, multivariable optimal control methods like MPC are required.

Although centralized MPC may provide the best achievable control performance, issues such as lack of flexibility and maintainability make this approach impractical. The general industrial practice to plant-wide MPC is to recognize the distributed structure of the processing units to design a network of decentralized MPCs. Decentralized controllers avoid the disadvantages associated with centralized control at the expense of poorer plant-wide control performance. To improve the performance of decentralized controllers, Distributed MPC (DMPC) methods have become centre of attention in the plant-wide optimal control research community. DMPC methods are divided into two general classes of non-coordinated and coordinated approaches. Coordinated Distributed MPC (CDMPC) networks, which consist of distributed controllers and a coordinator, are able to yield optimal centralized solution under a wide range of conditions.

This work addresses systematic development of CDMPC networks for plant-wide MPC of interconnected dynamical processes, by modifying the existing decentralized MPC network and designing coordinator. Goal Coordination, Interaction Prediction Coordination and Modified-Pseudo Model Coordination are the three coordination methods studied in this thesis to alter the network of decentralized linear constrained MPCs into CDMPC network. Convergence accuracy studies are provided for the proposed coordination algorithms. CDMPC networks are also developed to study the impacts of uncertainty on the CDMPC and coordinator design using an individual

chance-constrained approach. By modifying the CDMPC and coordinator in the Goal Coordination method, it is shown that choosing efficient numerical strategies can improve convergence performance of the coordination algorithm. A novel linear CDMPC network, which has performance of centralized nonlinear MPC, is presented to address the plant-wide nonlinear MPC problem. Numerical simulations are provided to test performance of the proposed CDMPC networks.

Acknowledgements

There are no proper words to convey my heartfelt gratitude and respect for my thesis supervisor, Dr. J. Fraser Forbes, for his support throughout these years, for keeping me going when times were tough, for the insightful discussions, and for offering invaluable advice. He has taught me to believe in myself and my abilities, to conduct original independent research and to realize the power of critical reasoning.

My special appreciation goes to members of my PhD examination committee, Dr. Ali Elkamel, Dr. John Doucette, Dr. Jinfeng Liu and Dr. Vinay Prasad for reviewing and evaluating my research and Dr. Tony Yeung for accepting to chair my PhD Final Oral.

I am deeply grateful to the professors of process control program in the Department of Chemical and Petroleum Engineering at Sharif University of Technology, Dr. Mohammad Shahrokhi, Dr. Ramin Bozorgmehri and Dr. Mahmoud Reza Pishvaie, for all that I have learnt from them and for being the inspiration to pursue my PhD degree.

I want to thank Natalia I. Marcos for being a good friend and for the helpful technical discussions.

I would like to offer my special thanks to my awesome best friend, Leily Mohammadi, for all that we have shared and for always being there.

I would like to express appreciation to my grand parents, to my in-laws and to all my family, for their love and encouragement. I would particularly like to thank my lovely aunts, Noushin and Hermien, and my dear uncle, Ali.

Most importantly of all, I want to thank the three people that are my world: My endless appreciation goes to my amazing parents, Simin and Hassan, for everything they have done for me, for always believing in me and for not letting me settle for less.

I could not have made it so far in my life without their unlimited love, support and dedication. I owe my deepest gratitude to my wonderful husband, Amir, for his pure love and support, for believing in me, for always being by my side and accompanying me through all the ups and downs of this unforgettable journey. The completion of this thesis would not have been possible without him.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Chemical plants typically consist of multiple geographically distributed interacting process units. Today's increased competitiveness in the chemical processing industry and the effort to achieve safe, environmentally responsible and profitable plant operation have led to plants that are more complex and highly integrated. The added complexities and intensified interactions are the result of using new equipment and applying novel approaches for improving economics, safety, energy recovery, reusing unused materials and reducing environmental impacts of the involved processes.

Efficient operation of such complex integrated plants, can be achieved by using optimal control strategies. *Centralized*, *decentralized* and *distributed* control schemes, which will be further discussed in detail, are three structures that can be adopted for plant-wide optimal control. Distributed control arose to exploit the advantages of both centralized and decentralized control structures, while avoiding their drawbacks. Various distributed control schemes are available, among which *coordinated distributed* control is a powerful approach that is capable of yielding maximum achievable control performance. Significant advantages that coordinated distributed control can bring into the realm of plant-wide optimal control, and the lack of comprehensive studies in this context, have inspired this research work.

Centralized Control Structure

In the centralized scheme, as shown in Figure 1.1, a monolithic controller is designed to control the overall plant. In the synthesis of centralized controllers, the complete plant model is used.

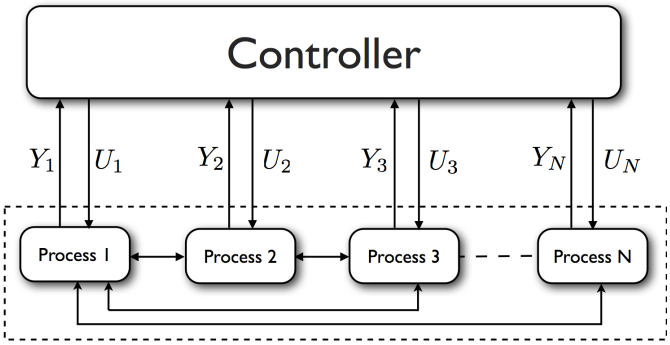


Figure 1.1: Centralized control structure

By using the complete plant model, the effects of interactions between the process units, are taken into account in the optimization problem. Therefore, centralized MPC produces optimal plant-wide control performance; however, despite the control performance benefit, industrial practitioners generally consider centralized control an unrealistic approach. In the past, one of the issues that made centralized MPC be impractical, was the limited power of computers in solving large optimization problems. With the advances in computer technologies and efficient optimization algorithms, the computational issues resulting from the large size of centralized convex optimization problems have been resolved to some extend; however, the availability of powerful computational tools does not imply that there is no computational limitation on the size and complexity of optimization problems. Another major drawback of centralized structure is related to maintainability and flexibility issues. That is, if a maintenance or a repair is required for one of the process units, then the entire control system should be shut down. Also, tuning, managing and improving the control performance of local processes, become difficult tasks under the centralized control structure.

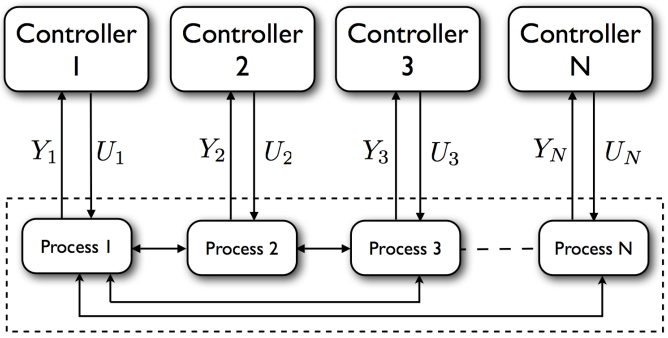


Figure 1.2: Decentralized control structure

Decentralized Control Structure

Decentralized control structure is the industrially practiced control architecture. As is illustrated in Figure 1.2, in this scheme, the distributed structure of the process units has been exploited to design independent controllers for each unit. In the decentralized scheme, the interactions between the processes are not considered in designing the independent (local) controllers. Since instead of one monolithic controller, N smaller controllers are used, the aforementioned issues of the centralized scheme, are avoided in the decentralized architecture; however, not explicitly incorporating interaction models in the controller synthesis, degrades control performance with respect to the centralized control. In other words, the solution produced by decentralized controllers are sub-optimal solutions. Only when the process units are truly decoupled will the decentralized controllers produce optimal plant-wide performance. In addition to the inherent performance loss, if strong interactions exist between local processes, stabilizing the system with decentralized controllers may become quite challenging (Venkat (2006), Sun and El-Farra (2008), Stewart (2010),Christofides *et al.* (2013)).

Distributed Control Structure

The issues with centralized and decentralized control structures have motivated the use of an alternative control structure, known as *distributed control*. In distributed control, the goal is to improve control performance of the decentralized control

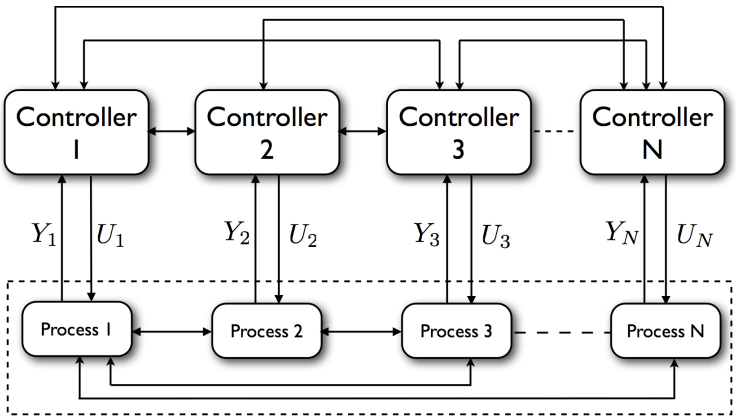


Figure 1.3: Non-coordinated distributed control structure

network and bring it closer to the performance of centralized control, while still benefitting from the characteristics of decentralized control structure. Appropriate information exchange within the control network, is one of the key factors in distributed control that leads to performance improvement. Distributed control, is divided into two main categories of non-coordinated and coordinated distributed control. Since the focus of this thesis is on coordinated distributed MPC, the following discussions on the two classes of distributed control are directed towards distributed model predictive control (DMPC).

Non-coordinated Distributed Control Structure

In the context of non-coordinated DMPC, the main idea is to improve control performance of decentralized MPC network and bring it closer to that of centralized MPC, by exchanging some information among local controllers and explicitly using interaction models in the controller's formulation. Also, depending on the distributed method, objective function of the distributed controllers may be modified. A general schematic representation of this control architecture is shown in Figure 1.3. The topology of the communication network and the cost function considered in the local MPC optimization problems have significant impact on the achieved amount of performance enhancement. Scattolini (2009) and Christofides *et al.* (2013) classified and reviewed various DMPC design algorithms. According to these two review papers, four general classifications can be made for DMPC algorithms: 1) *Fully* or *Partially*

connected algorithms, depending on whether any local MPC communicates with all other local controllers or just with a given subset of other local controllers; 2) *Non-cooperative* (*Independent*) or *Cooperative*, if each local MPC optimizes a local or global cost function; 3) *Non-iterative* or *Iterative*, according to whether within each sampling time, local controllers exchange information once or many times; and finally 4) *Sequential* or *Parallel*, depending on whether local MPC optimizations are executed in sequence or at the same time.

DMPC algorithms are designed based on *Game Theory* concepts. From game theory point of view, the iterative non-cooperative DMPC networks reach a *Nash Equilibrium* [1]. When iterative cooperative algorithms are used in a fully connected convex DMPC network and a sufficient number of iterations have been concluded, the control network can reach the centralized performance (Venkat (2006) and Rawlings and Stewart (2008)). Performance of other flavours of DMPC without using a coordinator, is always sub-optimal.

In the past decade, extensive research has been done on DMPC approaches. Scattolini (2009) and Christofides *et al.* (2013) provided a comprehensive list of references for various DMPC design methods. Pertaining to iterative cooperative DMPC, the first contribution was made by Venkat (2006), and further developed by Stewart (2010). As required by cooperative control, the interaction models were explicitly used in the local prediction models and a centralized objective function was used for each local controller. Venkat and Stewart proposed cooperative linear DMPC algorithms that were able to converge to the centralized solution, after sufficient number of iterations. In their proposed methods, the control based on any intermediate termination of the algorithm was guaranteed to be feasible and provide nominal closed-loop stable system.

Stewart (2010; 2011) extended the developed distributed convex DMPC strategies in Venkat (2006) and Stewart (2010), to non-convex DMPC problems, without

[1] The Nash Equilibrium is a solution concept of a non-cooperative game involving multiple players, in which each player is assumed to know the equilibrium strategies of the other players, and no player has anything to gain by changing only his own strategy unilaterally. If each player has chosen a strategy and no player can benefit by changing strategies while the other players keep theirs unchanged, then the current set of strategy choices and the corresponding payoffs constitute a Nash Equilibrium (WIKIPEDIA (2013)).

guaranteeing convergence of the distributed algorithm to the centralized performance. It was assumed that nonlinear dynamics lead to non-convex MPC problems. Another featured research in cooperation-based nonlinear DMPC was done by Liu *et al.* (2010*b*). They proposed two cooperative distributed strategies, where local controllers were designed by Lyapunov MPC techniques (Mhaskar *et al.* (2006)) to ensure closed-loop stability through explicit characterization of stability regions. In the first strategy, the distributed Lyapunov-based MPC (LMPC) communicated in one direction and were evaluated sequentially and non-iteratively. In the second approach, each distributed LMPC employed a two-way communication strategy, the LMPCs were evaluated in parallel and iteratively. Because of the non-convexity of the optimization problems, the aforementioned cooperative iterative nonlinear DMPC approaches cannot guarantee achieving the centralized nonlinear MPC solution; however, the intermediate solutions produced, were feasible and resulted in closed-loop stable plant.

Two other lines of work are also available for DMPC algorithms. The first one is based on negotiation between agents for systems that are coupled only through the inputs (Maestre *et al.* (2011)). The agents were DMPCs with local cost functions that did not know about the dynamics of their neighbours, but could communicate unrestrictedly with them in order to reach an agreement. Each agent communicated with its neighbours by exchanging their calculated inputs, based on a given protocol. At each sampling time, agents made proposals to improve an initial feasible solution on behalf of their local cost function, state and model. The proposals were accepted if the global cost function improved using the current solution. Closed-loop stability was guaranteed by including terminal region obtained from the centralized problem. The second line of work is based on using optimization concepts such as sensitivity information and duality, to modify local MPC formulations. The DMPCs iteratively communicate with their neighbours until overall (centralized) optimality is achieved. Scheu and Marquardt (2011) proposed a sensitivity-based algorithm for DMPC of linear time-invariant systems. Overall optimality was achieved by means of linear approximation of the objective functions of neighbouring controllers within the objective function of each local controller. Scheu and Marquardt showed convergence

of the proposed algorithm to the centralized solution and claimed that the method could be extended to cover nonlinear systems; however, feasibility and closed-loop stability properties of the distributed method were not studied. Doan *et al.* (2011) presented a decomposition method for DMPC of dynamically coupled linear systems, based on Fenchel's duality. Using the proposed algorithm, the network of distributed controllers were able to arrive at the centralized MPC solution. The application of the method proposed by Doan *et al.* was limited to quadratic programming problems only. Also, properties such as feasibility and guaranteed closed-loop stability of intermediate iterations were remained as open.

Coordinated Distributed Control Structure

In coordinated distributed control, the goal is to reproduce the centralized control performance, by proper coordination of local controllers. Mathematical theory of hierarchical multilevel systems provides the design foundation for this type of control structure.

Three basic classes of hierarchical systems are *Multistrata*, *Multilayer* and *Multiechelon* (Mesarovic *et al.* (1970)). The Multistrata structure considers modelling of various aspects and phenomena involved in complex systems, that each are viewed from a different level of abstraction. Levels of this structure are referred to as *Strata*. In the Multilayer structure, the solution to complex decision-making systems is sought in a hierarchical approach. A family of decision problems whose solution is attempted in a sequential manner, is defined. The solution of any problem in the sequence determines and fixes some parameters in the subsequent problem, so that the latter is completely specified and its solution can be found. The solution of the overall problem is obtained when each layer has successfully solved its own problem. Levels of this hierarchy, which each describes a family of decision problems, are known as *Layers*.

A system with Multiechelon or organizational hierarchy, has three characteristics: 1) the system involves a group of interacting subsystems; 2) the subsystems are decision making units and 3) some of the decision units are influenced by other decision units in a hierarchical manner. Each layer of this type of hierarchy is known

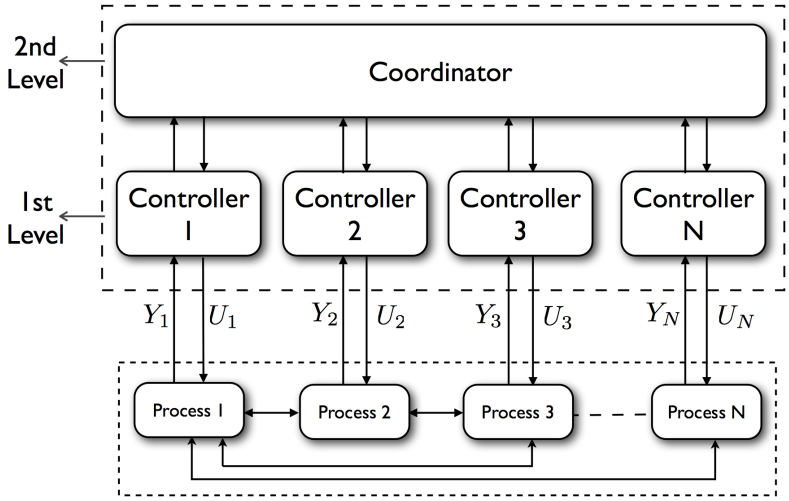


Figure 1.4: Coordinated distributed control structure

as *echelon*. Higher echelons are responsible for relaxing the interactions among the lower echelons, by properly accounting for the couplings and constraints, handling uncertainties and resolving conflicting goals (Mahmoud *et al.* (1977)). An important feature of this hierarchy is that, the higher echelons condition the lower level units, but they do not completely control the decision making processes of the lower levels. Therefore, lower level units have the freedom of selecting and calculating their own decision variables (Mesarovic *et al.* (1970)).

These three hierarchical structures may be imbedded within one another. Comprehensive discussions on various aspects of hierarchical multilevel systems can be found in the fundamental book by Mesarovic *et al.* (1970).

Optimal control of interconnected dynamical processes by synchronous single rate controllers, falls in the class of hierarchical multiechelon structure, where generally two echelons are involved. Local controllers are in the first level and the *supremal control system* (Mesarovic *et al.* (1970)) or the *coordinator*, is located at the second level. Figure 1.4 shows the elements and communication links of these two-level structures. In coordinated distributed control, local controllers do not use the centralized objective function and/or centralized model. Also, instead of communicating with each other, the local controllers in coordinated distributed systems only communicate with the coordinator. Therefore, the communication strategy in this type of structure

is standard, and can be regarded as one of the priorities of this structure over the distributed control structures that do not use a coordinator.

The role of the coordinator is to manage the behaviour of local controllers by manipulating their interactions, resolving the conflicts and adjusting the goal and model interventions, so that the centralized control performance is achieved. The coordinator influences local controllers through *intervention parameters* (Mahmoud *et al.* (1977)). Another advantage of coordinated structure over those without a coordinator, is that the former is designed to reconstruct the centralized performance, while the latter is designed to improve the performance of decentralized control network and guarantee feasibility and closed-loop stability properties of any solution produced by the distributed algorithm; however, only under certain communication topologies and conditions, the non-coordinated distributed schemes can yield the centralized performance.

In DMPC, despite the advantages of coordinated distributed structure and the rich literature on mathematical theory of hierarchical multilevel systems, the vast majority of research is focused on non-coordinated DMPC methods. The first study on coordinated DMPC was conducted by Cheng (2007), where the price-driven coordination method was applied to solve plant-wide MPC target calculation problem. Newton's method was used for the price-adjustment algorithm. Marcos (2012) extended the application of the price-driven coordination method in Cheng (2007), to plant-wide linear MPC of interconnected dynamical processes. Marcos (2012) also extended the prediction-driven coordination method originally employed by (Cohen (1977)) for Linear Quadratic (LQ) control of linear continuous-time systems based on the Interaction Prediction Coordination Principle (Mesarovic *et al.* (1970)), to the plant-wide unconstrained linear MPC of interconnected discrete-time linear dynamic systems. Marcos (2012) also addressed the problem of dual-rate DMPC, coordinated by price-driven coordination method.

1.1 Research Scope and Outline

Despite the ability of Coordinated DMPC (CDMPC) to enhance performance of decentralized control system to the highest level, its capabilities have been overshadowed by misconceptions about issues such as difficulties of coordinator synthesis, the need to re-design the control system, feasibility and stability considerations. This thesis intends to establish CDMPC as a promising approach to optimal plant-wide control of interconnected dynamical processes. This work contributes to the systematic development of coordinated distributed control structures, to elevate the performance of decentralized MPC network to the performance of centralized MPC. In particular, three coordination methods are used in conjunction with decentralized linear constrained MPC, to transform the existing decentralized control system into a CDMPC network. This thesis uses multilevel optimization-based coordination methods to develop the two essential elements in coordinated distributed networks, namely, the CDMPC and coordinator, for various MPC applications.

Converting any system of decentralized MPC into CDMPC network, involves two major tasks. First, the required modifications in the objective functions and constraints for the optimization problem of the decentralized MPC, should be determined. The modifications are performed by appropriate relaxation of the so-called interaction equations. The alterations done in the decentralized MPC, establish the connection between the CDMPC and the coordinator. The coordinator is then synthesized to properly compensate for the relaxed interactions in the overall control system. Mathematically, the compensation for the interaction effects is achieved by incorporating a proper numerical strategy to solve a system of algebraic equations.

The coordination methods investigated in this thesis, are based on duality. Local (decentralized/distributed) MPC problems addressed in this work, have convex structures, are synchronous and single-rate, and use state-space prediction models. It is assumed that all the states are measured. Also, the centralized MPC as the performance benchmark, and plant's existing decentralized constrained MPC network are assumed to yield stable plant operation by properly tuning the prediction

horizon, and without explicit characterization of stability regions. As required by any distributed algorithm, whether with or without a coordinator, it is assumed that interactions models are available.

Chapter 2 is a review of the prevailing coordination methods with the aim of carrying them over into developing CDMPC networks. The first coordination method studied in chapter 2, is the Goal Coordination method. Based on this method, the existing network of linear constrained decentralized MPC are altered into CDMPC network. The coordinator is designed by the algorithm involved in the numerical method chosen to solve an appropriate dual optimization problem. It is shown that the price-driven coordination method discussed in Cheng (2007) and Marcos (2012), is a special version of the Goal Coordination approach. Interaction Prediction and Modified Pseudo-Model Coordination methods are the other coordination approaches studied in Chapter 2. The coordinators are designed by the numerical algorithms used to solve portions of optimality conditions resulting from the overall control system. Two numerical strategies, a fixed-point iteration approach and a gradient-based algorithm, are employed to synthesize two different coordinators. The proposed CDMPC network developed by using the Interaction Prediction method, is the exhaustive version of the coordinated distributed scheme proposed by Marcos (2012), that has resolved several design issues. Another highlight of chapter 2 is the first introduction of the Modified Pseudo-Model Coordination approach in the context of CDMPC. The three proposed CDMPC networks guarantee arriving at optimal centralized MPC solution, upon convergence of the coordinator. Convergence accuracy studies are provided for the proposed coordination algorithms.

To illustrate the capabilities of coordination methods in various MPC applications, chapter 3, deals with the problem of uncertainties and their impacts on developing CDMPC networks. To this end, the three coordination approaches are successfully applied to a network of decentralized single chance-constrained MPC, to explicitly account for the effects of uncertain disturbances throughout the plant. In the Goal Coordination method, by using the method of Separated Augmented Lagrangian Algorithms (SALA) to numerically solve the dual optimization problem, a new coordinator with improved convergence behaviour, is proposed to address convergence

issues of the coordination algorithm, specially in the presence of active local inequality constraints. The use of SALA also leads to slight modifications in the objective functions of distributed controllers.

To address the plant-wide nonlinear MPC problem, a novel linear CDMPC network that reproduces the performance of centralized nonlinear MPC, is presented in chapter 4. In this new scheme, the Modified Pseudo-Model Coordination method and a special linearization technique are used to convert the system of linear decentralized MPC into linear CDMPC network. Portions of the optimality conditions for the overall control system, which result in a system of algebraic (nonlinear) equations, are solved numerically in the coordination layer. Thus, the coordinator is designed based on the algorithm involved in the numerical method adopted to solve the system of equations.

Chapter 5, summarizes and concludes the thesis, and provides suggestions for addressing the remaining challenges, further theoretical developments and new applications.

1.2 Terms and Definitions

To avoid misunderstandings, the meanings that are assigned to some of the key terms used throughout this thesis, are explained below.

In this work, the term **control network** is used to refer to a group of controllers, responsible for controlling the plant. Depending on the context, the term **plant-wide control** refers to controlling the plant by a network of controllers or a centralized controller. **Optimal plant-wide solution** and **optimal centralized solution** are used interchangeably and correspond to the optimal solution obtained from a centralized controller. Similarly, **optimal plant-wide performance** and **centralized performance**, both refer to optimal performance of a centralized controller.

Depending on the context, the term **local controller** refers to individual MPC in either decentralized or (coordinated) distributed control networks. Also, **subsystem** is used interchangeably with **local controller**.

Local process model may interchangeably be used with **Local prediction**

model, and both refer to the process model used for state prediction in the MPC problem.

MPC stands for Model Predictive Control/Controller. The term **DMPC** is the short form for Distributed MPC, and **CDMPC** stands for Coordinated Distributed MPC. **CDMPC network/system** corresponds to a distributed network in which DMPCs are connected to/coordinated by a **coordinator**. **GC-DMPC**, **IPC-DMPC** and **MPMC-DMPC** refer to CDMPC network, obtained by using Goal Coordination, Interaction Prediction Coordination and Modified Pseudo-Model Coordination methods, respectively.

In the context of CDMPC, **communication cycle** denotes a two-way information transmission between the coordinator and CDMPCs. **Iteration** also describes the process of transmitting information between the coordinator and CDMPCs. In the context of non-coordinated DMPC, communication cycle and iteration refer to the information exchange between the distributed controllers. Thus, **communication cycles** and **iterations** convey the same sense; the former is more of a network terminology and the latter is indicative of a successive computational procedure.

Chapter 2

Coordination Methods for Linear Distributed MPC

In this chapter, three CDMPC schemes are proposed for optimal control of interconnected dynamical systems. This is the first comprehensive study of coordination methods in the context of plant-wide MPC problem. The goal of using CDMPC is to achieve plant-wide optimal performance by appropriately coordinating the CDMPCs. In each of the three proposed CDMPC networks, the plant's already available network of decentralized MPCs is converted into a network of coordinated distributed MPCs by performing appropriate modifications and introducing a coordinator. The required modifications to the distributed controllers and formulations for the coordinators are presented in detail. Performance of the proposed CDMPC approaches is studied via case studies.

The inherent capability of model predictive control in dealing with multivariable and constrained systems along with its many other attributes have turned this advanced control method into an industry standard. Within a processing plant, rather than designing a plant-wide predictive controller, MPC is most commonly applied to individual process units or a single unit operations apparatus which, results in a network of decentralized controllers. As discussed in Chapter 1, the flexibility and maintainability issues, as well as the significant mismatch in complexities, time scales

and nonlinearities of local process units in a plant, make solving plant-wide MPC as a monolithic optimization problem, impractical. Thus, it would be valuable to have methods that improve performance of existing decentralized MPC networks and bring MPC technology to its full potential. To this end, the current chapter focuses on the formulation and analysis of three coordination methods for distributed linear constrained model predictive control of interconnected dynamical systems.

Since MPC has an optimization problem at its core, the toolset for solving the plant-wide MPC problem in the context of CDMPC would be the theories and algorithms developed for mathematical programming and control of multilevel systems. The literature on multi-level systems abounds and several coordination algorithms have been proposed for solving optimization problems arising in these systems (Mesarovic *et al.* (1970), Lasdon (1970), Pearson (1971), Mahmoud (1977), Haimes (1977), Singh and Titli (1978), Singh (1980), Jamshidi (1983)); however, as is discussed in Sorenson and Koble (1984), the complicated notation, specialized jargon and the *ad hoc* nature of many of the developed algorithms, make them difficult to understand and to apply. Sorenson and Koble (1984) employed a general taxonomic scheme originally proposed by Geoffrion (1970) and created a useful framework, in which a representative portion of the literature on algorithms for solving optimization problems in multilevel systems was unified into four general coordination methods with applications in optimal control of interconnected dynamical systems.

The four general coordination methods are the Goal Coordination (GC), Interaction Prediction Coordination (IPC), Co-State Coordination (CSC) and Pseudo-Model Coordination (PMC). The fundamental idea underlying these approaches is to have independent subproblems that each, in addition to local decision variables, contain certain parameters (original problem variables, pseudo-variables, Lagrange multipliers, co-state variables, or a combination of the above) called *coordinating variables*. Subproblem solutions are achieved by temporarily fixing the values of the coordinating variables. Based on the local solutions, a second level of control (a coordinator) is used to update the values of coordinating variables in an iterative manner, until a solution to the composite system is achieved (Sorenson and Koble (1984)). Coordinating variables are defined through the process of relaxing the

interactions between the subproblems. In all the subproblems, the set of local equality constraints with interaction variables is re-stated in terms of one set of local and one set of interaction constraints[1]. The main differences between various coordination methods lie in how the interaction constraints are treated and how the coordinating variables are calculated.

In the Goal Coordination method, duality theory is used to construct an equivalent two-level problem to the primal (centralized) optimization problem of the interconnected system. For each subsystem, the interaction constraints (complicating constraints) are adjoined to the cost function using Lagrange multipliers. In other words, a dual problem with respect to the complicating constraints is formed and local constraints remain as explicit constraints in the local units. The values of estimated Lagrange multipliers (dual variables) are iteratively improved by numerically solving the unconstrained dual optimization problem. A characteristic feature of the dual optimization problem is that its gradient is readily available and thus, gradient search procedures can be used. In the context of multi-level systems optimization, the GC approach is the most widely investigated method in the literature (e.g., Bauman (1966), Lasdon (1970), Pearson (1971), Singh and Titli (1978), Haimes (1977), Bazaraa and Shetty (1979), Singh (1980), Jamshidi (1983), Leunberger (1984), Sorenson and Koble (1984)). In most of these references, numerical unconstrained optimization methods, such as steepest ascent, in which only the gradient information of the dual optimization problem is needed, are used to update estimations of Lagrange multipliers[2]. The use of Newton's method, where the Hessian information is also employed, can be traced to the work of Bauman (1966).

Similar to the GC method, in the Interaction Prediction method, the duality principle[3] is applied to augment the cost function with the interaction constraints; however, for updating the dual variables, rather than solving the unconstrained dual

[1]The interaction constraints are also referred to as complicating constraints, as they contain variables from other subsystems.

[2]Throughout this work Lagrange multipliers estimates are referred to as prices.

[3]In the IPC, CSC and MPMC methods, the *Duality Principle* refers to *Relaxation* of the problem by penalizing certain constraints; however, the term *Duality* emphasizes that the desired constraints are linearly penalized by associating penalty parameters that are *dual variables*, with them. In other words, if the values of these dual variables equal the corresponding Lagrange multipliers values in the original problem, the *relaxed* problem is equivalent to the original problem.

optimization problem, an explicit satisfaction of portions of first-order optimality conditions of the composite system is used. Through the use of predictive type equations obtained from numerically solving optimality conditions, coordinating variables, including the price vector, are iteratively corrected with much less computational burden (Sorenson and Koble (1984)). The IPC method was originally proposed by Takahara (1965) for continuous linear interconnected local systems having quadratic performance measures. Convergence properties of the method were investigated by Cohen *et al.* (1974).

The duality principle is also applied in the Co-State coordination method, but here the dual problem is formed with respect to the local constraints and therefore the interaction constraints remain as explicit constraints in the primal optimization problem. This method was first proposed by Mahmoud *et al.* (1977). Later Cohen (1978) criticized the work stating that unlike what had been claimed in the original paper, the method would not produce separable local problems. Following that, Mahmoud corrected the CSC algorithm in Mahmoud (1978). In the CSC method three levels are used. In the first level, the subproblems are solved. The coordination task is done based on a gradient type routine and a prediction type update in the second and third levels, respectively. The major disadvantage of this method is that the separability of the local problems is lost as they contain the coupling equations. Therefore, the CSC approach cannot be regarded as a hierarchical method (Sorenson and Koble (1984)).

In the Pseudo-Model Coordination method, pseudo-variables are substituted for the interaction variables. The transformed interaction equations[4] are then included in the local objective functions using quadratic penalty functions and a penalty parameter. Numerically solving the appropriate optimality conditions of the integrated system, provides predictive update equations for the coordinating variables including pseudo-variables. The PM Coordination approach was first proposed by Pearson (1971). Based on his work, Singh (1975) developed a two-level algorithm for continuous time optimal control of interconnected systems with general local nonlinear models and linear interconnection dynamics. He also specialized the

[4]The equations used to define pseudo-variables are called the transformed interaction equations.

algorithm for linear quadratic (LQ) control problems. Simmons (1976) modified Singh's proposed update algorithm as some terms from the optimality conditions were missing. Also, Simmons (1976) showed that in Singh (1975), analysis of the problem and the statement about the independence of coordination from the choice of penalty parameter were not correct. As was discussed in Simmons (1976), only when the penalty parameter is large enough, will the aggregate of the local problems produce the original optimal solution; otherwise, the two-level system will produce an approximation of the plant-wide optimization problem. Sorenson and Koble (1984) overcame this major drawback by using *convexification*; wherein, the transformed interaction equations are quadratically penalized in the local objective functions and at the same time kept as explicit constraints. They then employed the duality principle to form the dual problem by relaxing the linear transformed equations. In other words, the transformed equations were appended to the the local cost functions by associating multipliers with them. In this thesis, the method proposed by Sorenson and Koble (1984) is referred to as the Modified Pseudo-Model Coordination (MPMC) method.

The literature has focused on developing computational algorithms for solving general multilevel optimization problems. As an important and common application, many of the developed algorithms have been tailored to solve optimal control problems for interconnected dynamical systems; however, the research on solving plant-wide MPC problems through coordination methods has not been given enough attention. The first attempt to do so was made by Cheng *et al.* (2007). They extended the price-driven coordination method for solving resource distribution or auction problems, proposed by Jose and Ungar (1998) and Jose and Ungar (2000), to large-scale quadratic programming problems and in particular, to the steady-state target calculation MPC problem. To efficiently adjust the prices, based on Newton's method, they proposed a price update scheme for the coordinator, in which sensitivity analysis and active-set change identification techniques were employed. Following their work, Marcos *et al.* (2008; 2009) extended the price-driven coordination method to solve the plant-wide MPC problem for interconnected dynamical systems. As was briefly mentioned above, in Cheng (2007) and Cheng *et al.* (2007), rather than incorporating

the multilevel optimization theories and concepts, an auction-based viewpoint was used to formulate the coordinated distributed problem. Consequently, understanding the elements of the price-driven coordination method and its extensions to other CDMPC applications have become very difficult. The main reason for such difficulty is that the formulations, as well as the interpretations on the impact of the variables and the mechanism of coordination were expressed in a different jargon than what is common in multilevel optimization and optimal control context. From the multilevel (hierarchical) systems viewpoint, the price-driven coordination method used in Cheng (2007) and Marcos (2012), is a special case of the Goal Coordination approach, where the coordinator uses Newton's method to update the prices.

Another application of coordination-based optimization methods in solving the plant-wide MPC problem was studied in Marcos (2012), where the IPC method in Cohen (1977) was adopted to design a coordinated distributed unconstrained linear MPC network. Convergence properties of the coordination algorithm and stability of the closed-loop system were investigated. The coordinator was designed such that the optimality conditions of the centralized problem were solved for the calculated local control variables and predicted states. Therefore, in the coordinator, at each iteration, by using the calculated manipulated variable changes from the local controllers, the system of linear equations resulting from the centralized plant model was solved to give the predicted states. The local control variable changes and the predicted states were then used in the first-order optimality conditions of the centralized problem to obtain the full price vector by solving the resulting system of linear equations. Then, the obtained price vector was localized by substituting the components of the full price vector corresponding to the i_{th} subsystem by zeros. Although the proposed formulation for the coordinator was correct, it contained redundant steps. Also, since the coordinated distributed scheme was formulated for unconstrained MPC, the effects of the limits on the local process outputs and control inputs were not taken into account in the coordinator's update equations.

In this chapter, according to the classification used in Sorenson and Koble (1984), the GC, IPC and MPMC methods are employed to construct coordinated distributed linearly constrained MPC networks. As the CDMPC will be formed by modifying

the already available decentralized MPC configuration in the plant, and because the Co-State Coordination method uses non-separable subproblems, this coordination method is not considered in this thesis.

The main contributions of this chapter can be summarized as follows:

- The formulations of local MPC and coordinator for constrained linear CDMPC network, are derived using the Goal Coordination Method. The required modifications for converting the decentralized MPC into CDMPC are performed by forming an appropriate dual optimization problem. The coordinator is formulated by the numerical algorithm used to solve the resulting dual unconstrained optimization problem. A general structure is proposed for the coordinator, in which Newton's method distinguishes one special version of this general form.

- Constrained linear CDMPC and the corresponding coordinator, are formulated using the Interaction Prediction Coordination method. The modifications in the local controllers are the same as those in the GC method. The coordinator's equations are derived based on the numerical algorithm employed to solve the appropriate portions of optimality conditions of the composite CDMPC. A fixed-point iteration technique and gradient-based method are the two numerical approaches used in the coordinator design. The coordinator involves very simple update equations. Based on the proposed coordinator formulation, unlike the method discussed by Marcos (2012), there is no need to solve the system of equations resulting from the centralized model. Also, the distributed controllers use the same price vector and thus, the coordinator does not need to localize the price vector. The effects of the local inequalities on the update process have been accounted for in the coordinator design phase.

- For the first time, the Modified Pseudo-Model Coordination method is applied to plant-wide linear constrained MPC problem. The decentralized controllers are converted into distributed controllers by replacing the interaction variables with the defined pseudo-variables and forming a relaxed problem. The coordinator is designed using the algorithm involved in the numerical method

chosen to solve the appropriate portions of the optimality conditions for the aggregate of CDMPC. The two numerical methods used in constructing the coordinator are the fixed-point iteration and gradient-based methods. The obtained coordinator equations are very simple. Also, the effects of local inequalities on the coordinating variables have explicitly been taken into account in the coordinator's update equations.

It is hoped that the comprehensive study and analysis of the application of coordination methods to linearly constrained decentralized MPC, presented in this chapter, will be helpful in the development of new coordinated distributed schemes for various plant-wide MPC problems including robust, stochastic, nonlinear and asynchronous cases.

2.1 Background

Before proceeding with the main topic of this chapter, some mathematical background and notation are presented, which will be used later in the development and analysis of CDMPC.

2.1.1 Plant Model

In this chapter the following linearized discrete-time state-space representation of the entire plant is used:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \tag{2.1a}$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \tag{2.1b}$$

where $\mathbf{x}(k) \in \mathbb{R}^{n_x}$ is the state vector, $\mathbf{u}(k) \in \mathbb{R}^{n_u}$ is the control input vector and $\mathbf{y}(k) \in \mathbb{R}^{n_y}$ is the output vector, all in deviation variable form and at sampling instant k. It is assumed that full state information is available.

Considering that the plant contains N interconnected process units, the $\mathbf{A}$ and $\mathbf{B}$ matrices in the plant model (2.1a) can be written in the following block-wise form:

$$\mathbf{A} = \left[\begin{array}{c|c|c|c|c} \mathbf{A}_{11} & \dots & \mathbf{A}_{1j} & \dots & \mathbf{A}_{1N} \\ \hline \vdots & & \vdots & & \vdots \\ \hline \mathbf{A}_{i1} & \dots & \mathbf{A}_{ii} & \dots & \mathbf{A}_{iN} \\ \hline \vdots & & \vdots & & \vdots \\ \hline \mathbf{A}_{N1} & \dots & \mathbf{A}_{Nj} & \dots & \mathbf{A}_{NN} \end{array}\right] \quad \text{and} \quad \mathbf{B} = \left[\begin{array}{c|c|c|c|c} \mathbf{B}_{11} & \dots & \mathbf{B}_{1j} & \dots & \mathbf{B}_{1N} \\ \hline \vdots & & \vdots & & \vdots \\ \hline \mathbf{B}_{i1} & \dots & \mathbf{B}_{ii} & \dots & \mathbf{B}_{iN} \\ \hline \vdots & & \vdots & & \vdots \\ \hline \mathbf{B}_{N1} & \dots & \mathbf{B}_{Nj} & \dots & \mathbf{B}_{NN} \end{array}\right] \tag{2.2}$$

where the pair $(\mathbf{A}_{ii}, \mathbf{B}_{ii})$ along the diagonal of $\mathbf{A}$ and $\mathbf{B}$, respectively, represents the dynamics of the local process unit i. The off-diagonal pairs $(\mathbf{A}_{ij}, \mathbf{B}_{ij})$ models the interaction between the local process unit i and j.

Using this block-wise decomposition of the state space matrices, the plant model (2.1) can be re-written as:

$$\mathbf{x}_i(k+1) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{u}_i(k) + \sum_{\substack{j=1 \\ j\neq i}}^{N} [\mathbf{A}_{ij}\mathbf{x}_j(k) + \mathbf{B}_{ij}\mathbf{u}_j(k)] \tag{2.3a}$$

$$\mathbf{y}_i(k) = \mathbf{C}_{ii}\mathbf{x}_i(k) \tag{2.3b}$$

where $i = 1, \dots, N$. The vectors $\mathbf{x}_i(k) \in \mathbb{R}^{n_{xi}}$, $\mathbf{u}_i(k) \in \mathbb{R}^{n_{ui}}$ and $\mathbf{y}_i(k) \in \mathbb{R}^{n_{yi}}$ contain the states, inputs and outputs of the local unit i at time k in deviation form, respectively. Also, $\sum_{i=1}^{N} n_{xi} = n_x$, $\sum_{i=1}^{N} n_{ui} = n_u$ and $\sum_{i=1}^{N} n_{yi} = n_y$ that is the sum of the local variables equals the total number of plant variables, which implies that states and control inputs are not shared between the process units.

It is assumed that the plant is controlled by a decentralized MPC network. It should be emphasized that in this work, it is not of concern how the plant is partitioned into local processes. The three important implications of this assumption are: 1) the local pair $(\mathbf{A}_{ii}, \mathbf{B}_{ii})$ corresponds to the dynamics of local unit i; 2) the local pairs $(\mathbf{A}_{ii}, \mathbf{B}_{ii})$, are controllable; 3) the existing decentralized controllers will be converted into distributed controllers by modifying their formulations.

Also, without loss of generality and throughout this thesis, the state-space model is used for MPC; however, the coordination methods discussed here are not limited to MPC with state-space prediction models. Since MPC with any type of prediction model and objective function is an optimization problem, it is expected that the same

concepts and principles of coordination methods discussed in this work, can be used for formulating CDMPC with other available forms of process models, such as Finite Impulse Response (FIR), step response and transfer function models. The successful extension of the price-driven method to coordinated distributed MPC, in which finite step-response models were used in Marcos *et al.* (2008), is a good example to show the applicability of the coordination methods to MPC with prediction models other than state-space.

2.1.2 Plant-Wide Model Predictive Control Problem

In order to assess the accuracy of various coordination approaches in distributed MPC, a performance benchmark is required. Although it may not be physically realizable for the reasons previously discussed, the best achievable performance would be obtained by solving the MPC problem for the entire plant, by using an accurate and complete model for the entire plant. In this thesis, the solution to such a plant-wide (centralized) MPC problem represents the maximum achievable plant performance and is one of the performance benchmarks that is used throughout this work.

Model predictive control poses the control problem as an optimization problem, and refers to a class of advanced model-based multivariable control algorithms that calculate manipulated variables profiles by using a process model to optimize a performance objective subject to constraints over a future time horizon. The first move of the calculated optimal manipulated variables profiles are implemented in the process and the rest are discarded. At the next control interval, based on the receding horizon principle, the calculations are repeated using the same time horizon and updated process measurements.

MPC is commonly formulated as a constrained quadratic programming problem. In this work, the MPC formulation in Maciejowski (2000), in which the quadratic objective function penalizes deviations of the predicted controlled outputs from a reference trajectory and the control input changes[5], has been used. The constraints

[5]Deviations of the input vector from some ideal resting values, i.e, $||\mathbf{U}_i - \mathbf{U}_i^*||^2_{\mathbf{S}}$ can also be used in the objective function (Maciejowski (2000)). In this work, this term has not been included in the MPC formulation. If this term is also present in the CDMPC schemes the *coordinator's formulation* and the *modified terms* in the local decentralized controllers will change to include its

include the plant model (2.3) and linear inequality constraints, which account for process limitations on the outputs, control inputs and control input changes. Such a plant-wide linear constrained MPC can mathematically be described as:

$$\min_{\mathbf{X},\boldsymbol{\Delta}\mathbf{U}} \quad \sum_{i=1}^{N} \left[\left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i \right] \tag{2.4a}$$

subject to

$$\mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] + \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\alpha \mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1)\right] + \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\beta \mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_j(k+a|k)\right] \tag{2.4b}$$

$$\begin{aligned} &\mathbf{y}_i^{min}(k+l+1) \leq \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \leq \mathbf{y}_i^{max}(k+l+1) \\ &\mathbf{u}_i^{min}(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}_i^{max}(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \leq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \leq \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b) \end{aligned} \tag{2.4c}$$

where

$$\begin{cases} l = 0, \ldots, H_p - 1 \\ b = 0, \ldots, H_u - 1 \\ \text{for} \begin{cases} l = 0, \quad \alpha = 1, \quad \beta = 0 \\ l \neq 0 \quad \alpha = 0, \quad \beta = 1 \end{cases} \\ i = 1, \ldots, N \end{cases}$$

where, H_p and H_u are prediction and control horizons, respectively. The terms $\mathbb{C}_{ii}\mathbf{X}_i$ in the objective function (2.4a) and $\mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k)$ in the inequality constraints (2.4c) represent the predicted output $\mathbf{Y}_i$ where the vector $\mathbf{Y}_i \triangleq \left[\mathbf{y}_i(k+1|k)^T, \ldots, \mathbf{y}_i(k+H_p|k)^T\right]^T$ contains the predicted outputs of local unit i over the entire prediction horizon. Similarly, $\mathbf{X}_i \triangleq \left[\mathbf{x}_i(k+1|k)^T, \ldots, \mathbf{x}_i(k+H_p|k)^T\right]^T$ is the local predicted state vector over the prediction horizon H_p. The matrix $\mathbb{C}_{ii} = blkdiag(\overbrace{\mathbf{C}_{ii}, \ldots, \mathbf{C}_{ii}}^{H_p \quad times})$ is a block diagonal matrix. The vector $\boldsymbol{\Delta}\mathbf{U}_i \triangleq \left[\boldsymbol{\Delta}\mathbf{u}_i(k|k)^T, \ldots, \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k)^T\right]^T$ includes the manipulated variable

effects; however, the required changes will automatically be obtained if the *same procedures* discussed in this work for the coordination algorithms, are followed.

changes for the subsystem i over the entire control horizon. The vector $\mathbf{r}_i \triangleq \left[\mathbf{r}_i(k+1|k)^T, \ldots, \mathbf{r}_i(k+H_p|k)^T\right]^T$ involves the pre-specified set-point trajectory. $\mathbb{Q}_{ii}$ and $\mathbb{R}_{ii}$ are block-wise matrices containing the weighting matrices $\mathbf{Q}_{ii}$ and $\mathbf{R}_{ii}$ along their diagonal. In the equality constraint (2.4b), the control input $\mathbf{u}(k+l|k)$ has been expressed in terms of the control input change $\boldsymbol{\Delta}\mathbf{u}(k+1|k)$ using the following relation between the control input and its change (Maciejowski (2000)):

$$\mathbf{u}(k+l|k) = \sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}(k+a|k) + \mathbf{u}(k-1) \tag{2.5}$$

where for $l = H_u, \ldots, H_p - 1$, the control input vector remains constant (i.e, $\boldsymbol{\Delta}\mathbf{u}(k+a|k) = 0$).

Remark 2.1.1 *The local process model (2.4b) is written based on three terms. These terms distinguish the centralized, decentralized and distributed MPC. They also, play an important role in developing different coordinated distributed methods.*

The terms containing states at time k and past control inputs are referred to as ***known*** *terms, as the current states and past control inputs are available. The terms involving the predicted states and predicted control input changes are designated as* ***unknown*** *terms because the predicted states and predicted manipulated variable changes are decision variables in the optimization problem.*

Based on this classification and from the view-point of local unit i, the first term in (2.4b), $\mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right]$ *includes both* ***known*** *and* ***unknown local*** *information; The second term,* $\sum_{\substack{j=1 \\ j\neq i}}^{N} [\alpha\mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1)]$ *contains* ***known interaction*** *information and the third portion,* $\sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\beta\mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_j(k+a|k)\right]$, *contains* ***unknown interaction*** *information.*

Based on the availability of interaction information, the local process model (2.4b) can equivalently be described by the following two systems of equations:

$$\mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] + \sum_{\substack{j=1\\ j\neq i}}^{N}\left[\alpha\mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1)\right] + \mathbf{v}_i(k+l|k) \tag{2.6a}$$

$$\mathbf{v}_i(k+l|k) \triangleq \sum_{\substack{j=1\\ j\neq i}}^{N}\left[\beta\mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_j(k+a|k)\right] \tag{2.6b}$$

where the vector $\mathbf{v}_i$ *is referred to as* ***interaction variable****, since it contains the unknown interaction information; The set of equations (2.6a) is regarded as* ***local dynamic equations****, since all terms have either index i or is a known interaction term. The system of equations (2.6b) describes the* ***(local) interaction equations****. In the context of distributed optimal control, interaction equations are referred to as the complicating constraints because they contain variables from more than one process unit (Pearson (1971), Sorenson and Koble (1984)). The manner in which the interaction equations are treated provides the basis of different coordinated distributed formulations.*

2.1.3 Decentralized Model Predictive Control

By performing some modifications, the available decentralized MPC can be converted into CDMPC. This implies that the decentralized controllers also provide a performance benchmark for assessing the performance of the distributed controllers. In the synthesis of decentralized MPC, interaction effects are not taken into account. That is, instead of solving the monolithic optimization problem (2.4), N smaller, fully decoupled optimization problems are solved. Decentralized MPC is formulated as:

$$\min_{\mathbf{X}_i,\boldsymbol{\Delta}\mathbf{U}_i} \quad \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i \tag{2.7a}$$

subject to

$$\mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] \tag{2.7b}$$

$$\begin{aligned}
&\mathbf{y}_i^{min}(k+l+1) \leq \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \leq \mathbf{y}_i^{max}(k+l+1) \\
&\mathbf{u}_i^{min}(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}_i^{max}(k+b) \\
&\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \leq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \leq \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b)
\end{aligned} \tag{2.7c}$$

Unless the process units in the plant are truly decoupled, when (2.7) is solved for all the decentralized controllers, the aggregate of the solutions will not produce the optimal solution of the plant-wide problem (2.4). Thus, neglecting the interactions between the local units in the decentralized scheme yields sub-optimal plant-wide performance. Nevertheless, the decentralized MPC approach is commonly used in industry and it is important to investigate possible solutions for its performance improvement.

2.2 CDMPC Problem Statement

In this thesis, the coordinated distributed model predictive controllers have the following general form:

$$\min_{\mathbf{X}_i, \boldsymbol{\Delta}\mathbf{U}_i} \quad \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i + \{CoT\}_i \tag{2.8a}$$

subject to

$$\begin{aligned}
&\mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] + \\
&\sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\alpha\mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1)\right] + \mathbf{v}_i(k+l|k)
\end{aligned} \tag{2.8b}$$

$$\begin{aligned}
&\mathbf{y}_i^{min}(k+l+1) \leq \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \leq \mathbf{y}_i^{max}(k+l+1) \\
&\mathbf{u}_i^{min}(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}_i^{max}(k+b) \\
&\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \leq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \leq \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b)
\end{aligned} \tag{2.8c}$$

where $\{CoT\}$ stands for *Coordinating Term(s)* and the index i indicates that each local controller has its own coordinating term(s). $\{CoT\}_i$ links the local controller to the coordinator and therefore, it will contain some form of *coordinating variables* that are to be updated by the coordinator. The interaction variables vector $\mathbf{v}_i$, depending on the coordination method, is calculated either by the distributed controllers or the coordinator.

If $\{CoT\}_i$ in the objective function (2.8a) is set to zero, the aggregate of the distributed controllers (2.8) along with the interaction equations (2.6b) will form the plant-wide MPC problem (2.4). This relation between the distributed and plant-wide controllers has two implications: 1) it implies that the local interaction equations in the CDMPCs are compensated for by $\{CoT\}_i$; 2) and if the correct values of $\{CoT\}_i$ are found, the plant-wide performance will be achieved by the distributed controllers.

Comparing the decentralized formulation (2.7) and the distributed formulation (2.8) shows that two modifications have been made in the decentralized controllers: 1) the coordinating terms are added to their objective functions; 2) and the **known interaction** information and interaction variables $\mathbf{v}_i$, have been added to the local process models[6].

The above comparisons between distributed, decentralized and plant-wide MPC indicate that these three schemes can be converted into each other by full or partial inclusion and/or exclusion of the interaction equations. Thus, knowing the *interaction models* is the major requirement in converting the decentralized controllers into their distributed counterparts. If the interaction models are available, the CDMPC will provide the advantage of having independent controllers and reaching optimal plant-wide performance. In all the coordination methods discussed in this thesis, it is assumed that the coordinator knows the interaction models $(\mathbf{A}_{ij}, \mathbf{B}_{ij})$, where $\{\forall i, j | i, j = 1, \ldots, N \ \& \ j \neq i\}$.

Treatment of interaction equations plays an important role in determining the coordinating terms. The numerical strategy chosen to calculate the coordinating variables involved in the $\{CoT\}_i$, dictates the coordinator equations. The coordination process is inherently an iterative scheme because of the computational techniques used in calculation of the coordinating variables (Mahmoud (1977)).

In designing every CDMPC network, the following two questions should be answered:

1. How are the coordinating terms $\{CoT\}_i$, defined?

2. How are the coordinating variables in the $\{CoT\}_i$, calculated?

[6]In other words, the local prediction models (2.7b) have been replaced by the dynamic equations (2.6a).

The answer to the first question dictates the modifications required in the existing decentralized MPC, in order to convert them into CDMPC. The answer to the second question provides the structure of the coordinator.

2.3 Goal Coordinated Distributed MPC

In this section, the GC method is applied to define the coordinating terms in linear CDMPC and construct the coordinator. The use of the GC approach for coordinating the DMPC, where the linking (complicating) constraints are of equality type will result in an unconstrained dual optimization problem, which can be numerically solved using any numerical optimization approaches. The update equations resulting from the adopted numerical method forms the coordinator.

2.3.1 Distributed Controllers in the GC Method

To determine the $\{CoT\}_i$ in the distributed controllers and construct the coordinator, first the interaction equations (2.6b) are used to form the local *interaction error vector* $\mathfrak{E}_i$,

$$\mathfrak{E}_i(k+l|k) \triangleq \mathbf{v}_i(k+l|k) - \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\beta \mathbf{A}_{ij} \mathbf{x}_j(k+l) + \mathbf{B}_{ij} \sum_{a=0}^{l} \mathbf{\Delta u}_j(k+a) \right] \tag{2.9}$$

and then $\mathfrak{E}_i$ is written over the entire prediction horizon H_p for all of the local systems to form the *overall interaction error vector* $\mathfrak{E}$ as follows:

$$\mathfrak{E} = \begin{bmatrix} \mathfrak{E}_1(k|k) \\ \vdots \\ \mathfrak{E}_1(k+H_p-1|k) \\ \hdashline \vdots \\ \hdashline \mathfrak{E}_N(k|k) \\ \vdots \\ \mathfrak{E}_N(k+H_p-1|k) \end{bmatrix} =$$

$$\begin{bmatrix} \mathbf{v}_1(k|k) - \sum_{j=2}^{N} [\mathbf{B}_{1j}\Delta\mathbf{u}_j(k|k)] \\ \mathbf{v}_1(k+1|k) - \sum_{j=2}^{N} \left[\mathbf{A}_{1j}\mathbf{x}_j(k+1|k) + \mathbf{B}_{1j}\sum_{a=0}^{1}\Delta\mathbf{u}_j(k+a|k)\right] \\ \vdots \\ \mathbf{v}_1(k+H_p-1|k) - \sum_{j=2}^{N} \left[\mathbf{A}_{1j}\mathbf{x}_j(k+H_p-1|k) + \mathbf{B}_{1j}\sum_{a=0}^{H_u-1}\Delta\mathbf{u}_j(k+a|k)\right] \\ \hdashline \vdots \\ \hdashline \mathbf{v}_N(k|k) - \sum_{j=1}^{N-1} [\mathbf{B}_{Nj}\Delta\mathbf{u}_j(k|k)] \\ \mathbf{v}_N(k+1|k) - \sum_{j=1}^{N-1} \left[\mathbf{A}_{Nj}\mathbf{x}_j(k+1|k) + \mathbf{B}_{Nj}\sum_{a=0}^{1}\Delta\mathbf{u}_j(k+a|k)\right] \\ \vdots \\ \mathbf{v}_N(k+H_p-1|k) - \sum_{j=1}^{N-1} \left[\mathbf{A}_{Nj}\mathbf{x}_j(k+H_p-1|k) + \mathbf{B}_{Nj}\sum_{a=0}^{H_u-1}\Delta\mathbf{u}_j(k+a|k)\right] \end{bmatrix} \tag{2.10}$$

Next, (2.10) is re-arranged such that it is converted into the following additive separable form:

$$\mathfrak{E} = \underbrace{\begin{bmatrix} \mathbf{v}_1(k|k) \\ \mathbf{v}_1(k+1|k) \\ \vdots \\ \mathbf{v}_1(k+H_p-1|k) \\ \hdashline \vdots \\ \hdashline -\left[\mathbf{B}_{N1}\Delta\mathbf{u}_1(k|k)\right] \\ -\left[\mathbf{A}_{N1}\mathbf{x}_1(k+1|k) + \mathbf{B}_{N1}\sum_{a=0}^{1}\Delta\mathbf{u}_1(k|k)\right] \\ \vdots \\ -\left[\mathbf{A}_{N1}\mathbf{x}_1(k+H_p-1|k) + \mathbf{B}_{N1}\sum_{a=0}^{H_u-1}\Delta\mathbf{u}_1(k+a|k)\right] \end{bmatrix}}_{\boldsymbol{\Theta}_1\begin{bmatrix}\mathbf{X}_1 \\ \boldsymbol{\Delta}\mathbf{U}_1 \\ \mathbf{V}_1\end{bmatrix}}$$

$$+ \ldots + \tag{2.11a}$$

$$\underbrace{\begin{bmatrix} -\left[\mathbf{B}_{1N}\Delta\mathbf{u}_N(k|k)\right] \\ -\left[\mathbf{A}_{1N}\mathbf{x}_N(k+1|k) + \mathbf{B}_{1N}\sum_{a=0}^{1}\Delta\mathbf{u}_N(k+a|k)\right] \\ \vdots \\ -\left[\mathbf{A}_{1N}\mathbf{x}_1(k+H_p-1|k) + \mathbf{B}_{1N}\sum_{a=0}^{H_u-1}\Delta\mathbf{u}_N(k+a|k)\right] \\ \hdashline \vdots \\ \hdashline \mathbf{v}_N(k|k) \\ \mathbf{v}_N(k+1|k) \\ \vdots \\ \mathbf{v}_N(k+H_p-1|k) \end{bmatrix}}_{\boldsymbol{\Theta}_N\begin{bmatrix}\mathbf{X}_N \\ \boldsymbol{\Delta}\mathbf{U}_N \\ \mathbf{V}_N\end{bmatrix}}$$

$$\implies \mathfrak{E} = \sum_{i=1}^{N} \boldsymbol{\Theta}_i \begin{bmatrix}\mathbf{X}_i \\ \boldsymbol{\Delta}\mathbf{U}_i \\ \mathbf{V}_i\end{bmatrix} \tag{2.11b}$$

In (2.11b), the overall interaction error vector $\mathfrak{E}$ is expressed as summation of *local*

matrices and variables. For each subsystem, the matrix $\mathbf{\Theta}_i$ contains the interaction models $\mathbf{A}_{ji}$ and $\mathbf{B}_{ji}$ and the Identity matrix $\mathbf{I}$ corresponding to $\mathbf{v}_i$. The dimension of $\mathbf{\Theta}_i$ would be $(\sum_{i=1}^{N} n_{xi}) \times (2n_{xi} + n_{ui})$. More details on the elements and structure of this matrix for a simple example, can be found in Appendix B.

Remark 2.3.1 *It should be noted that, since the overall interaction error vector* $\mathfrak{E}$ *is defined as the difference between the left- and right-hand sides of the interaction equations (2.6b) of all local units over the entire prediction horizon, the system of equations*

$$\mathfrak{E} = \sum_{i=1}^{N} \mathbf{\Theta}_i \begin{bmatrix} \mathbf{X}_i \\ \mathbf{\Delta U}_i \\ \mathbf{V}_i \end{bmatrix} = \mathbf{0} \tag{2.12}$$

is the additive separable form of the interaction dynamic constraints (2.6b).

Now the plant-wide MPC problem (2.4)[7] is re-written by replacing the equality constraints (2.4b) with (2.6a) and the additive separable form of (2.6b) as :

$$\min_{\mathbf{X},\mathbf{\Delta U},\mathbf{V}} \quad \sum_{i=1}^{N} \left[\left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right) + \mathbf{\Delta U}_i^T \mathbb{R}_{ii} \mathbf{\Delta U}_i \right] \tag{2.13a}$$

subject to

$$\mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l} \mathbf{\Delta u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] + \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\alpha \mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1)\right] + \mathbf{v}_i(k+l|k) \tag{2.13b}$$

[7]Since it is assumed that the interactions models are known, the plant-wide MPC problem (2.4) can be formed by performing the following simple modifications on the *aggregate* of the decentralized controllers (2.7):

1. The local process models (2.7b) are replaced by (2.6a).
2. The interaction equations (2.6b) are included in the constraints of the decentralized controllers.

Thus, the use of plant-wide MPC problem in defining CoT_i for the distributed controllers does not mean that in order to create the distributed controllers, a physical centralized MPC should exist.

$$\sum_{i=1}^{N} \mathbf{\Theta}_i \begin{bmatrix} \mathbf{X}_i \\ \mathbf{\Delta U}_i \\ \mathbf{V}_i \end{bmatrix} = \mathbf{0} \tag{2.13c}$$

$$\begin{aligned} &\mathbf{y}_i^{min}(k+l+1) \le \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \le \mathbf{y}_i^{max}(k+l+1) \\ &\mathbf{u}_i^{min}(k+b) \le \mathbf{u}_i(k+b|k) \le \mathbf{u}_i^{max}(k+b) \\ &\mathbf{\Delta u}_i^{min}(k+b) \le \mathbf{\Delta u}_i(k+b|k) \le \mathbf{\Delta u}_i^{max}(k+b) \\ &i = 1, \ldots, N \end{aligned} \tag{2.13d}$$

In order to maintain the separable structure of the decentralized controllers, the additively separable equality constraints (2.13c) are relaxed by associating the vector of Lagrange multipliers $\mathbf{p}$ having the dimension $(\sum_{i=1}^{N} n_{xi} \times 1)$, with them. This relaxation results in the following problem:

$$\min_{\mathbf{X},\mathbf{\Delta U},\mathbf{V}} \quad \sum_{i=1}^{N} \left[\left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right) + \mathbf{\Delta U}_i^T \mathbb{R}_{ii} \mathbf{\Delta U}_i \right] + \mathbf{p}^T \sum_{i=1}^{N} \mathbf{\Theta}_i \begin{bmatrix} \mathbf{X}_i \\ \mathbf{\Delta U}_i \\ \mathbf{V}_i \end{bmatrix} \tag{2.14a}$$

subject to

$$\begin{aligned} \mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii} \left[\sum_{a=0}^{l} \mathbf{\Delta u}_i(k+a|k) + \mathbf{u}_i(k-1) \right] + \\ \sum_{\substack{j=1 \\ j \ne i}}^{N} \left[\alpha \mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1) \right] + \mathbf{v}_i(k+l|k) \end{aligned} \tag{2.14b}$$

$$\begin{aligned} &\mathbf{y}_i^{min}(k+l+1) \le \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \le \mathbf{y}_i^{max}(k+l+1) \\ &\mathbf{u}_i^{min}(k+b) \le \mathbf{u}_i(k+b|k) \le \mathbf{u}_i^{max}(k+b) \\ &\mathbf{\Delta u}_i^{min}(k+b) \le \mathbf{\Delta u}_i(k+b|k) \le \mathbf{\Delta u}_i^{max}(k+b) \\ &i = 1, \ldots, N \end{aligned} \tag{2.14c}$$

The optimization problem (2.14) is separable, which means that its solution can be found by solving N smaller optimization problems. This implies that in the GC method, the $\{CoT\}_i = \mathbf{p}^T \mathbf{\Theta}_i \begin{bmatrix} \mathbf{X}_i \\ \mathbf{\Delta U}_i \\ \mathbf{V}_i \end{bmatrix}$ and thus, the distributed controllers (2.8)

become:

$$\min_{\mathbf{X}_i,\boldsymbol{\Delta}\mathbf{U}_i,\mathbf{V}_i} \quad \left(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i\right)^T\mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i\right)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i+\mathbf{p}^T\boldsymbol{\Theta}_i\begin{bmatrix}\mathbf{X}_i\\ \boldsymbol{\Delta}\mathbf{U}_i\\ \mathbf{V}_i\end{bmatrix} \tag{2.15a}$$

subject to

$$\begin{aligned}\mathbf{x}_i(k+l+1|k)=&\mathbf{A}_{ii}\mathbf{x}_i(k+l|k)+\mathbf{B}_{ii}\left[\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k)+\mathbf{u}_i(k-1)\right]\\&+\sum_{\substack{j=1\\j\neq i}}^{N}\left[\alpha\mathbf{A}_{ij}\mathbf{x}_j(k+l|k)+\mathbf{B}_{ij}\mathbf{u}_j(k-1)\right]+\mathbf{v}_i(k+l|k)\end{aligned} \tag{2.15b}$$

$$\begin{aligned}&\mathbf{y}_i^{min}(k+l+1)\leq\mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k)\leq\mathbf{y}_i^{max}(k+l+1)\\&\mathbf{u}_i^{min}(k+b)\leq\mathbf{u}_i(k+b|k)\leq\mathbf{u}_i^{max}(k+b)\\&\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b)\leq\boldsymbol{\Delta}\mathbf{u}_i(k+b|k)\leq\boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b)\end{aligned} \tag{2.15c}$$

where the coordinating variable vector $\mathbf{p}$, is the so-called price vector (Cheng (2007), Marcos (2012)) which is determined by the coordinator and transmitted to the local controllers. The local vector $\mathbf{V}_i \triangleq \left[\mathbf{v}_i(k|k)^T,\mathbf{v}_i(k+1|k)^T,\ldots,\mathbf{v}_i(k+H_p-1|k)^T\right]^T$ contains the estimated interaction effects and is determined by the local controllers, as part of their decision variables. The constant matrix $\boldsymbol{\Theta}_i$ involves the interaction models. The coordinator calculates these constant matrices according to (2.11a) and provides them to the local controllers[8].

The way $\{CoT\}_i$ is defined dictates that same price vector should be provided to each distributed controllers. In each controller, the price vector acts as the penalty for violating the local part of all the interaction dynamic constraints. In other words, the price vector penalizes the local part of the overall interaction error vector.

Now, the important question is how the correct price vector can be calculated such that the aggregate of distributed controllers (2.14) provides the plant-wide MPC solution (2.13). The answer to this question ultimately leads to the appropriate coordinator structure.

[8]Since the process models used in this work are assumed to be time-invariant, $\boldsymbol{\Theta}_i$ is constant. Thus, the coordinator needs to just calculate $\boldsymbol{\Theta}_i$, once.

2.3.2 Coordinator Design in the GC method

In the GC method, the solution strategy to solve (2.14) is developed by numerically solving the *dual optimization problem*[9]. Before introducing the dual optimization problem, in order to work with simpler notation, let the objective function of the aggregate problem (2.14) be represented by $\mathcal{J}$:

$$\begin{aligned}&\mathcal{J}(\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\mathbf{V},\mathbf{p})=\sum_{i=1}^{N}\mathcal{J}_i(\mathbf{X}_i,\boldsymbol{\Delta}\mathbf{U}_i,\mathbf{V}_i,\mathbf{p})=\\&\sum_{i=1}^{N}\left[\left(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i\right)^T\mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i\right)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\right]+\mathbf{p}^T\sum_{i=1}^{N}\boldsymbol{\Theta}_i\begin{bmatrix}\mathbf{X}_i\\\boldsymbol{\Delta}\mathbf{U}_i\\\mathbf{V}_i\end{bmatrix}\end{aligned}\tag{2.16}$$

Based on (2.14) and (2.16), the *Lagrangian dual function* (Pearson (1971), Bazaraa and Shetty (1979), Leunberger (1984), Boyd and Vandenberghe (2004)) or *perturbation function* (Geoffrion (1971)) $\varphi(\mathbf{p})$ is defined as:

$$\varphi(\mathbf{p})\triangleq\inf_{\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\mathbf{V}}\quad\{\mathcal{J}(\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\mathbf{V},\mathbf{p})|(2.14b)\quad\&\quad(2.14c)\quad,i=1,\ldots,N\}\tag{2.17}$$

In (2.13), let the optimal value be denoted by $\mathcal{P}^*$. For every value of the vector $\mathbf{p}$, the dual function $\varphi(\mathbf{p})$ yields a lower bound on $\mathcal{P}^*$[10]:

[9]Details on *Duality* can be found in many optimization textbooks (e.g., Bertsekas (1999), Boyd and Vandenberghe (2004)). In this section, the tutorial by Vert (2006) has been used to adapt the concepts to the problem at hand.

[10]Let $\mathbf{Z}_i\triangleq\begin{bmatrix}\mathbf{X}_i\\\boldsymbol{\Delta}\mathbf{U}_i\\\mathbf{V}_i\end{bmatrix}$ and (2.13) be described by the following compact form:

$$\begin{aligned}&\min_{\mathbf{Z}_1,\ldots,\mathbf{Z}_N}J_p(\mathbf{Z}_1,\ldots,\mathbf{Z}_N)\\&\text{subject to}\\&\mathbf{f}_i(\mathbf{Z}_i)=\mathbf{0}\text{ and }\sum_{i=1}^{N}\boldsymbol{\Theta}_i\mathbf{Z}_i=\mathbf{0}\text{ and }\mathbf{g}_i(\mathbf{Z}_i)\leq\mathbf{0}\text{ where i=1,...,N.}\end{aligned}$$

Let $\bar{\mathbf{Z}}_i$ be any feasible point, i.e, $\mathbf{f}_i(\bar{\mathbf{Z}}_i)=\mathbf{0}$, $\sum_{i=1}^{N}\boldsymbol{\Theta}_i\bar{\mathbf{Z}}_i=\mathbf{0}$, $\mathbf{g}_i(\bar{\mathbf{Z}}_i)\leq\mathbf{0}$ and $\boldsymbol{\mu}_i\geq\mathbf{0}$. Thus, for all $\bar{\mathbf{Z}}$ the following inequality holds:

$$L(\bar{\mathbf{Z}},\boldsymbol{\lambda},\mathbf{p},\boldsymbol{\mu})=J_p(\bar{\mathbf{Z}})+\sum_{i=1}^{N}\boldsymbol{\lambda}_i^T\mathbf{f}_i(\bar{\mathbf{Z}}_i)+\mathbf{p}^T\sum_{i=1}^{N}\boldsymbol{\Theta}_i\bar{\mathbf{Z}}_i+\sum_{i=1}^{N}\boldsymbol{\mu}_i\mathbf{g}_i(\bar{\mathbf{Z}}_i)\leq J_p(\bar{\mathbf{Z}})$$

where L is the Lagrangian. Also, $\varphi(\mathbf{p})=\inf_{\mathbf{Z}}\quad L(\mathbf{Z},\boldsymbol{\lambda},\mathbf{p},\boldsymbol{\mu})\leq L(\bar{\mathbf{Z}},\boldsymbol{\lambda},\mathbf{p},\boldsymbol{\mu})$.

$$\varphi(\mathbf{p}) \leq \mathcal{P}^* \tag{2.18}$$

The definition of the dual function (2.17), implies that the lower bound depends on the vector $\mathbf{p}$. In order to find the value of $\mathbf{p}$ that yields the greatest lower bound the following optimization problem should be solved:

$$\max_{\mathbf{p}} \quad \varphi(\mathbf{p}) \tag{2.19}$$

where because the nature of the vector $\mathbf{p}$ is of Lagrange multiplier associated with equality constraints, there is no sign restriction for vector $\mathbf{p}$ and thus, (2.19) is an unconstrained optimization.

Problem (2.19) is called the *dual optimization problem*. Since the dual function can be any real value i.e., $\{\varphi(\mathbf{p}) \in \mathbb{R} : -\infty \leq \varphi(\mathbf{p}) \leq \infty\}$, the *infimum* is found by performing *minimization*. Thus, (2.19) can be written as:

$$\begin{aligned} \max_{\mathbf{p}} \quad & \min_{\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\mathbf{V}} \quad \mathcal{J}(\mathbf{X}, \boldsymbol{\Delta}\mathbf{U}, \mathbf{V}, \mathbf{p}) \\ & \text{subject to} \\ & (2.14b) \quad \& \quad (2.14c) \\ & i = 1, \ldots, N \end{aligned} \tag{2.20}$$

Let the optimal value of the dual problem (2.20) be denoted by $\mathcal{D}^*$. From (2.18) it is immediately concluded that:

$$\mathcal{D}^* \leq \mathcal{P}^* \tag{2.21}$$

Inequality (2.21) is known as *Weak Duality* and the *Duality Gap*, which is defined as the difference between the optimal objective function value of the dual and primal

---

$$\implies \varphi(\mathbf{p}) \leq J_p(\bar{\mathbf{Z}})$$

Thus, for the optimal solution $\mathbf{Z}^*$ that corresponds to the optimal value $J_p(\mathbf{Z}^*) = \mathcal{P}^*$, the above inequality becomes $\varphi(\mathbf{p}) \leq \mathcal{P}^*$.

optimization problems, is positive, i.e., $\mathcal{D}^* - \mathcal{P}^* \geq 0$. The equality $\mathcal{D}^* = \mathcal{P}^*$ in which the optimal value of the dual optimization problem equals the optimal value of the original problem (before relaxation) is recognized as *Strong Duality* and in this case the duality gap is zero. Unlike weak duality that always holds for any optimization problem, strong duality can only be guaranteed to be held for convex optimization problems that satisfy Slater's constraint qualification condition. This condition requires the original optimization problem be strictly feasible or in other words, has at least one feasible point.

The above discussion indicates that in order to reach the optimal plant-wide solution through solving the dual optimization problem, strong duality should hold. The distributed controllers have quadratic objective functions and linear constraints and thus, the aggregate problem (2.14) is a convex optimization problem. Also, as the plant-wide problem (2.13) is the performance benchmark, it is assumed that it has at least one feasible point. Therefore, strong duality holds and the optimal solution of the plant-wide MPC problem (2.13) can equivalently be obtained by solving the CDMPC problem (2.20).

So far, it has been shown that the dual problem (2.20) produces the optimal plant-wide solution. The next issue is determining how solving (2.20) is related to the coordinator design?

Solving the dual problem (2.20) may seem awkward, since its cost function is a constrained minimization problem. A valuable characteristic of (2.20) is that its gradient can be easily evaluated as:

$$\mathbf{J} = \frac{dL(\mathbf{X}, \boldsymbol{\Delta}\mathbf{U}, \mathbf{V}, \mathbf{p})}{d\mathbf{p}} = \mathfrak{E} = \sum_{i=1}^{N} \boldsymbol{\Theta}_i \begin{bmatrix} \mathbf{X}_i \\ \boldsymbol{\Delta}\mathbf{U}_i \\ \mathbf{V}_i \end{bmatrix} \tag{2.22}$$

where L is the Lagrangian of the aggregate of distributed controllers. Equation (2.22) indicates that $\mathbf{J}$ depends on the local controllers solutions. Thus, if $\mathbf{X}_i$, $\boldsymbol{\Delta}\mathbf{U}_i$ and $\mathbf{V}_i$ are available, the gradient $\mathbf{J}$ is easily calculated.

Similarly, the Hessian of the dual problem can be calculated as:

$$\mathbf{H} = \frac{d\mathbf{J}}{d\mathbf{p}} = \frac{d\mathfrak{E}}{d\mathbf{p}} = \sum_{i=1}^{N} \mathbf{\Theta}_i \frac{d \begin{bmatrix} \mathbf{X}_i \\ \mathbf{\Delta U}_i \\ \mathbf{V}_i \end{bmatrix}}{d\mathbf{p}} \tag{2.23}$$

According to (2.23), the Hessian depends on the sensitivity information of the local solutions. Therefore, if the variations of the local solutions with respect to the price vector are known, the Hessian can be calculated using (2.23). Details for calculating local sensitivity matrices can be found in Cheng (2007) and Marcos (2012).

Considering that the gradient information is readily available, the dual optimization problem (2.20) can be solved using gradient methods[11] (e.g., steepest ascent and Newton's method). Thus, the solution strategy for solving the dual problem (2.20) involves the following general iterative scheme:

$$\mathbf{p}^{q+1} = \mathbf{p}^q + \epsilon^q \mathbf{S}^q \tag{2.24}$$

where q is the iteration counter, ϵ is the optimal step length and $\mathbf{S}$ is the search direction that includes the gradient information $\mathbf{J}$ and the Hessian $\mathbf{H}$. The set of equations (2.24) provides updates for the price vector $\mathbf{p}$. On the other hand, the coordinator is responsible for updating the coordinating variable vector $\mathbf{p}$. Consequently, (2.24) is the coordinator in the GC method.

The general coordinator structure (2.24) includes the two main groups of gradient methods. One category is the first-order gradient approaches such as steepest ascent and conjugate gradient methods. These approaches, at each iteration, use gradient information $\mathbf{J}^q$, in the search direction vector $\mathbf{S}^q$. In solving the dual optimization problem, first-order gradient methods show very slow convergence rates, particularly when active local inequality constraints are present. Therefore, in most cases, a significant number of communication cycles is required, which makes these approaches impractical for on-line implementation. The problem worsens when, rather than using a fixed step-size, the optimal ϵ^q is required to be calculated at each iteration to improve the convergence behaviour. In the next chapter, a first-order gradient approach with

[11] Gradient or indirect search methods are referred to the methods that require derivatives of the objective function to numerically solve unconstrained optimization problems (Rao (1996)).

improved convergence performance is proposed, which converges in a finite number iterations and does not use optimal step-size calculations.

The second category of gradient methods includes techniques that use first and second order information in (2.24). If a Negative Definite (ND) Hessian is available, Newton's method will be the most efficient method for updating the price vector. In Newton's method, the update equation (2.24) becomes:

$$\mathbf{p}^{q+1} = \mathbf{p}^q - \epsilon^q \left[\mathbf{H}^{-1}\right]^q \mathbf{J}^q \tag{2.25}$$

The Hessian matrix can be obtained from the local sensitivity information. For CDMPC without local inequalities, the Hessian is usually ND and thus, the first choice for solving the dual optimization problem would be Newton's method. In the presence of local inequality constraints, due to the change of local active sets, the resulting Hessian may not be ND and a Newton-based coordinator (2.25) can encounter convergence problems. Quasi-Newton methods (i.e., Davidon-Flethcher-Powell (DFP) and Broyden-Fletcher-Goldfarb-Shanno (BFGS)), in which the Hessian is estimated (Rao (1996)), are other alternatives for numerically solving the dual optimization problem. The issues with using these methods are that the procedure for the Hessian estimation may complicate coordinator's calculations and similar to the Newton's method, the estimated Hessian may cause convergence problems for the coordination algorithm.

The iterative nature of the coordinator establishes a hierarchical two-level scheme in which, during each sampling instant k, the coordinator and CDMPC can communicate and exchange information until the coordinator converges. Convergence is achieved when the optimal prices have been obtained (i.e., $\mathbf{p} = \mathbf{p}^*$) and has the following implications:

- The value of the price vector[12] becomes equal to value of the Lagrange multipliers associated with the interaction equations in the plant-wide MPC problem.

[12]The prices are estimates of the Lagrange multiplier vector associated with the equality constraints resulting from the plant model or equivalently the Lagrange multipliers associated with interaction equations.

- The interaction equations are satisfied.
- The overall interaction error vector $\mathfrak{E}$ becomes zero.
- The vector $\mathbf{v}_i$, provides exact estimates of local interactions.

Once the coordinator converges, the first sets of calculated optimal control moves (i.e, $\mathbf{u}_i(k|k)$) are implemented at the local process unit level.

In the GC method, communication and information exchange between CDMPC and the coordinator is performed in a systematic manner shown in Algorithm 1. Also, the information flow in the GC method expressed by block diagrams, can be found in Appendix A.

Algorithm 1 : Implementation of GC-DMPC network

1. Coordinator: Iteration counter q is set to 1.
2. Coordinator: Price vector $\mathbf{p}$ is arbitrarily initialized.
3. Coordinator: Step-size ϵ is chosen/calculated.
4. Coordinator: Price vector $\mathbf{p}$ is sent to the local controllers.
5. Local Controllers: Local optimization problems (2.15) are solved.
6. Local Controllers: Local optimal solutions $\mathbf{X}_i$, $\mathbf{\Delta U}_i$ and $\mathbf{V}_i$ are sent to the coordinator [13].
7. Coordinator: If $||\mathfrak{E}|| \leq \mathfrak{e}$, algorithm stops. Otherwise, next step is taken.
8. Coordinator: The gradient matrix $\mathbf{J}$ (and the Hessian $\mathbf{H}$) is calculated using (2.22) (and (2.23)).
9. Coordinator and Local Controllers: If required, optimal step-size ϵ is calculated.
10. Coordinator: Price vector $\mathbf{p}$ is updated using (2.24).
11. Coordinator: Iteration counter q is increased by 1.
12. Steps 3-7 are repeated.

[13] If Newton's method (2.25) is used, local sensitivity information should also be sent.

2.3.2.1 Coordinator Convergence Accuracy

Convergence of the coordination algorithm in the GC method can be proved by defining the following Lyapunov function[14]:

$$\mathcal{L}(\mathbf{p}) \triangleq \mathcal{D}^* - \varphi(\mathbf{p}) \tag{2.26}$$

where $\mathcal{D}^*$ is the optimal value of the dual optimization problem (2.19) and its objective function $\varphi(\mathbf{p})$ is the dual function defined in (2.17). The optimization problem (2.19) implies that when $\mathbf{p} = \mathbf{p}^*$, the function $\mathcal{L}$ in (2.26) is zero and when $\mathbf{p} \neq \mathbf{p}^*$, $\mathcal{L}$ is positive definite, i.e., $\mathcal{L} \geq 0$. Thus, (2.26) can be a candidate Lyapunov function.

Changes of $\mathcal{L}$ during the iterations (communication cycles) is expressed by:

$$\dot{\mathcal{L}} = \frac{d\mathcal{L}}{dq} = \left(\frac{d\mathcal{L}}{d\mathbf{p}}\right)^T \left(\frac{d\mathbf{p}}{dq}\right) \tag{2.27}$$

From (2.26), taking the derivative of the Lyapunov function with respect to the price vector yields:

$$\frac{d\mathcal{L}}{d\mathbf{p}} = -\frac{d\varphi(\mathbf{p})}{d\mathbf{p}} = -\mathbf{J} \tag{2.28}$$

where $\mathbf{J}$ is the gradient vector (2.22).

Also, as previously discussed, because gradient-based numerical optimization methods are used to solve the *maximization* problem (2.19), in general, the coordinator is described by the set of iterative equations (2.24). The system of equation (2.24) implies that the rate of change of the price vector over the iterations is *in the direction of the gradient vector* or *is proportional to the gradient vector*, that is:

$$\frac{d\mathbf{p}}{dq} \propto \mathbf{J} = \zeta\mathbf{J} \tag{2.29}$$

where ζ is a positive constant.

Replacing (2.28) and (2.29) in (2.26) yield:

$$\dot{\mathcal{L}} = -\mathbf{J}^T\mathbf{J} \tag{2.30}$$

[14]Convergence proof of the coordination algorithm provided in this section, is inspired by the convergence analysis of unfeasible coordination methods in Singh and Titli (1978).

According to (2.30), it is clear that $\dot{\mathcal{L}} \leq 0$. Therefore, the price-update scheme expressed by the ordinary differential equations (2.29) is stable, which implies that the coordination algorithm is convergent.

2.4 Interaction Prediction Coordinated Distributed MPC

In this section, details on determining the coordinating terms for the linear CDMPC and the coordinator design in the IPC method are provided. The coordinator is constructed by numerically solving portions of first-order optimality conditions of the aggregate of the CDMPC.

2.4.1 Distributed Controllers in the IPC Approach

In the IPC method, $\{CoT\}_i$ is determined as in the GC method. The only difference is that in the IPC method, $\mathbf{v}_i$ is among the coordinating variables and consequently is determined by the coordinator. Following the procedures of section 2.3, the linearly constrained CDMPC becomes:

$$\min_{\mathbf{X}_i, \mathbf{\Delta U}_i} \quad (\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i)^T \mathbb{Q}_{ii} (\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i) + \mathbf{\Delta U}_i^T \mathbb{R}_{ii} \mathbf{\Delta U}_i + \mathbf{p}^T \mathbf{\Phi}_i \begin{bmatrix} \mathbf{X}_i \\ \mathbf{\Delta U}_i \end{bmatrix} \tag{2.31a}$$

subject to

$$\begin{aligned} \mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii} \left[\sum_{a=0}^{l} \mathbf{\Delta u}_i(k+a|k) + \mathbf{u}_i(k-1) \right] \\ + \sum_{\substack{j=1 \\ j \neq i}}^{N} [\alpha \mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1)] + \mathbf{v}_i(k+l|k) \end{aligned} \tag{2.31b}$$

$$\begin{aligned} &\mathbf{y}_i^{min}(k+l+1) \leq \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \leq \mathbf{y}_i^{max}(k+l+1) \\ &\mathbf{u}_i^{min}(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}_i^{max}(k+b) \\ &\mathbf{\Delta u}_i^{min}(k+b) \leq \mathbf{\Delta u}_i(k+b|k) \leq \mathbf{\Delta u}_i^{max}(k+b) \\ &i = 1, \ldots, N \end{aligned} \tag{2.31c}$$

where the price vector $\mathbf{p}$ comes from the coordinator and is shared by all the local controllers; the vector $\mathbf{v}_i$ contains *predictions* of the interactions and is updated and provided for the local controllers by the coordinator; the constant matrix $\mathbf{\Phi}_i$ contains

the information of the interaction models $\mathbf{A}_{ji}$ and $\mathbf{B}_{ji}$ and has the dimension of $(\sum_{i=1}^{N} n_{xi}) \times (n_{xi} + n_{ui})$. In Appendix B, the $\mathbf{\Phi}_i$ matrix is built for a simple example.

It can be seen that the distributed controllers (2.31) are very similar to the distributed controller formulation in the GC method (2.15). The two main differences are:

1. In (2.15), the vector $\mathbf{v}_i$ is estimated by the distributed controllers, while in (2.31) $\mathbf{v}_i$ is predicted by the coordinator.

2. For ***implementation*** purposes, instead of $\mathbf{\Theta}_i$, the lower dimensional matrix $\mathbf{\Phi}_i$ is used in the $\{CoT\}_i$. This is because, at each communication cycle, the vector $\mathbf{V}_i$ is calculated by the coordinator and thus, is temporarily fixed in value when the CDMPCs are solving their optimization problems. This implies that, in performing local quadratic optimizations (2.31), multiplication of the relevant parts of $\mathbf{p}^T\mathbf{\Theta}_i$ by $\mathbf{V}_i$ produces a constant term in the local objective functions that does not have an impact on determining the decision variables. In other words, in the objective function (2.31a), the priced term $\mathbf{p}^T\mathbf{\Theta}_i \begin{bmatrix} \mathbf{X}_i \\ \mathbf{\Delta U}_i \\ \mathbf{V}_i \end{bmatrix}$ has the same effect as $\mathbf{p}^T\mathbf{\Phi}_i \begin{bmatrix} \mathbf{X}_i \\ \mathbf{\Delta U}_i \end{bmatrix}$; however, in the ***coordinator design phase***, the aggregate of the distributed controllers with the priced term $\mathbf{p}^T\mathbf{\Theta}_i \begin{bmatrix} \mathbf{X}_i \\ \mathbf{\Delta U}_i \\ \mathbf{V}_i \end{bmatrix}$ (i.e., problem (2.14)) is used so that the optimality conditions written for the aggregate of CDMPCs be the same as the optimality conditions for the centralized problem (2.13).

2.4.2 Coordinator Design in the IPC Method

After determining $\{CoT\}_i$ in the distributed controllers, the coordinator is designed to calculate the coordinating variables $\mathbf{p}$ and $\mathbf{V}_i$, based on the numerical solution strategy chosen for solving portions of first-order optimality conditions resulting from the optimization problem of aggregate of the CDMPC.

Before proceeding with synthesizing the coordinator, to work with simpler notation,

the aggregate of the CDMPC (2.14) is written over the entire control and prediction horizons, so that it can be re-stated in the following compact form:

$$\min_{\mathbf{X}_i,\boldsymbol{\Delta}\mathbf{U}_i} \quad \sum_{i=1}^{N}\Big[(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)^T\,\mathbb{Q}_{ii}\,(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\Big]+\mathbf{p}^T\sum_{i=1}^{N}\boldsymbol{\Theta}_i\begin{bmatrix}\mathbf{X}_i\\ \boldsymbol{\Delta}\mathbf{U}_i\\ \mathbf{V}_i\end{bmatrix} \tag{2.32a}$$

subject to

$$\hat{\mathbb{A}}_{ii}\mathbf{X}_i+\hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i-\mathbf{V}_i=\overbrace{\bar{\mathbb{A}}_{ii}\mathbf{x}_i(k)+\bar{\mathbb{B}}_{ii}\mathbf{u}_i(k-1)+\sum_{\substack{j=1\\ j\neq i}}^{N}\big[\bar{\mathbb{A}}_{ij}\mathbf{x}_j(k)+\bar{\mathbb{B}}_{ij}\mathbf{u}_j(k-1)\big]}^{\mathcal{Z}_i} \tag{2.32b}$$

$$\begin{aligned}&\mathbf{Y}_i^{min}\le\mathbb{C}_{ii}\mathbf{X}_i\le\mathbf{Y}_i^{max}\\ &\mathfrak{U}_i^{min}\le\mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\le\mathfrak{U}_i^{max}\\ &i=1,\ldots,N\end{aligned} \tag{2.32c}$$

where the matrices $\hat{\mathbb{A}}_{ii}$,$\hat{\mathbb{B}}_{ii}$, $\bar{\mathbb{A}}_{ii}$, $\bar{\mathbb{B}}_{ii}$, $\bar{\mathbb{A}}_{ij}$ and $\bar{\mathbb{B}}_{ij}$, are obtained by writing the local process models over the entire prediction horizon; on the right-hand side of the equality constraints (2.32b), the vector $\mathcal{Z}_i$ contains known (available) information; the inequality constraints imposed on the manipulated variable $\mathbf{u}_i(k+b|k)$ and its change $\boldsymbol{\Delta}\mathbf{u}(k+b|k)_i$ have been written for the whole control horizon and the vectors $\mathfrak{U}_i^{min}$ and $\mathfrak{U}_i^{max}$ and the matrix $\mathbb{U}_i$ have been obtained by combining these inequalities. Details of constructing (2.32) have been provided in the Appendix B.

It should be emphasized that re-stating the aggregate problem (2.14) as the compact form (2.32) is done to facilitate derivation of the coordinator formulation; however, for ***implementation purposes***, it suffices to solve the optimization problem of the distributed controllers (2.31) along with the coordinator's update equations, which will be provided shortly.

To design the coordinator in the IPC method, rather than estimating the interaction effects by the local controller and finding the best price vector by numerically solving the dual optimization problem, portions of optimality conditions of the aggregate problem, are used to calculate the correct price vector $\mathbf{p}$ and interaction effects $\mathbf{v}_i$. To do so, first the Lagrangian of the aggregate problem (2.32)

is formed as:

$$
\begin{aligned}
&L(\mathbf{X}, \boldsymbol{\Delta}\mathbf{U}, \mathbf{V}, \boldsymbol{\Lambda}, \boldsymbol{\Omega}, \boldsymbol{\Upsilon}, \mathbf{p}) = \sum_{i=1}^{N} L_i(\mathbf{X}_i, \boldsymbol{\Delta}\mathbf{U}_i, \mathbf{V}_i, \boldsymbol{\Lambda}_i, \boldsymbol{\Omega}_i, \boldsymbol{\Upsilon}_i, \mathbf{p}) = \\
&\sum_{i=1}^{N} \left[\left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i \right] + \mathbf{p}^T \sum_{i=1}^{N} \boldsymbol{\Theta}_i \begin{bmatrix} \mathbf{X}_i \\ \boldsymbol{\Delta}\mathbf{U}_i \\ \mathbf{V}_i \end{bmatrix} + \\
&\sum_{i=1}^{N} \boldsymbol{\Lambda}_i^T \left(\hat{\mathbb{A}}_{ii}\mathbf{X}_i + \hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i - \mathbf{V}_i - \mathcal{Z}_i \right) + \\
&\sum_{i=1}^{N} \left[\boldsymbol{\Omega}_{i,min}^T \left(\mathbf{Y}_i^{min} - \mathbb{C}_{ii}\mathbf{X}_i \right) + \boldsymbol{\Omega}_{i,max}^T \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{Y}_i^{max} \right) \right] + \\
&\sum_{i=1}^{N} \left[\boldsymbol{\Upsilon}_{i,min}^T \left(\mathfrak{U}_i^{min} - \mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i \right) + \boldsymbol{\Upsilon}_{i,max}^T \left(\mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i - \mathfrak{U}_i^{max} \right) \right]
\end{aligned}
\tag{2.33}
$$

where $\boldsymbol{\Lambda}_i$, $\boldsymbol{\Omega}_i$ and $\boldsymbol{\Upsilon}_i$ are Lagrange multipliers associated with the local dynamic equations and inequality constraints. The vectors $\boldsymbol{\Lambda}$, $\boldsymbol{\Omega}$ and $\boldsymbol{\Upsilon}$ contain the relevant multipliers for all the local subsystems. Optimal solution of the aggregate optimization problem (2.32) must satisfy the following optimality conditions[15]:

$$
\begin{cases} \nabla_{\eta} L \left(\mathbf{X}, \boldsymbol{\Delta}\mathbf{U}, \mathbf{V}, \boldsymbol{\Lambda}, \boldsymbol{\Omega}, \boldsymbol{\Upsilon}, \mathbf{p} \right) = \mathbf{0} \\ \text{for } \eta = \{ \mathbf{X}, \boldsymbol{\Delta}\mathbf{U}, \mathbf{V}, \boldsymbol{\Lambda}, \mathbf{p} \} \end{cases}
\tag{2.34}
$$

The conditions $\nabla_{\mathbf{X}} L = \mathbf{0}$, $\nabla_{\boldsymbol{\Delta}\mathbf{U}} L = \mathbf{0}$, $\nabla_{\boldsymbol{\Lambda}} L = \mathbf{0}$ along with the feasibility, complementary slackness and non-negativity conditions arising from the local

[15] The precise statement of the necessary KKT conditions for the aggregate problem (2.32) is:

Optimality: $\nabla_{\mathbf{X}} L = \mathbf{0}$, $\nabla_{\boldsymbol{\Delta}\mathbf{U}} L = \mathbf{0}$ and $\boxed{\nabla_{\mathbf{V}} L = \mathbf{0}}$

Feasibilty: $\begin{cases} \nabla_{\boldsymbol{\Lambda}} L = \mathbf{0} \text{ and } \boxed{\nabla_{\mathbf{p}} L = \mathbf{0}} \\ \begin{bmatrix} \mathbf{Y}^{min} - \mathbb{C}\mathbf{X} \\ \mathbb{C}\mathbf{X} - \mathbf{Y}^{max} \\ \mathfrak{U}^{min} - \mathbb{U}\Delta\mathbf{U} \\ \mathbb{U}\Delta\mathbf{U} - \mathfrak{U}^{max} \end{bmatrix} \leq \mathbf{0} \end{cases}$

Complementary slackness: $\begin{bmatrix} \boldsymbol{\Omega} \\ \boldsymbol{\Upsilon} \end{bmatrix}^T \begin{bmatrix} \mathbf{Y}^{min} - \mathbb{C}\mathbf{X} \\ \mathbb{C}\mathbf{X} - \mathbf{Y}^{max} \\ \mathfrak{U}^{min} - \mathbb{U}\Delta\mathbf{U} \\ \mathbb{U}\Delta\mathbf{U} - \mathfrak{U}^{max} \end{bmatrix} = \mathbf{0}$

Non-negativity: $\begin{bmatrix} \boldsymbol{\Omega} \\ \boldsymbol{\Upsilon} \end{bmatrix} \geq \mathbf{0}$

Throughout this thesis, in order to use simpler language, wherever the term *optimality conditions* is used for the coordinator design in the IPC method, it refers to the above Optimality and Feasibility conditions included in boxes.

inequalities are handled by the distributed controllers (2.31). The system of equations resulting from $\nabla_{\mathbf{p}} L = \mathbf{0}$ and $\nabla_{\mathbf{V}} L = \mathbf{0}$ are used to design the coordinator. The former yields prediction equations for the coordinating variables $\mathbf{V}_i$ and the latter provides a relationship between the multipliers $\mathbf{\Lambda}$, $\mathbf{\Omega}$ and $\mathbf{p}$.

A key idea in the IPC method is applying numerical techniques to portions of the optimality conditions, to obtain update equations for the coordinating variables (Sorenson and Koble (1984), Cohen and Miara (1990)). Fixed-point iteration technique is one of the common numerical strategies used in the coordinator design that converts the optimality conditions (2.34) into the following prediction type update form:

$$\nabla_{\eta} L \left(\mathbf{X}^q, \mathbf{\Delta U}^q, \mathbf{V}^{q+1}, \mathbf{\Lambda}^q, \mathbf{\Omega}^q, \mathbf{\Upsilon}^q, \mathbf{p}^{q+1} \right) = \mathbf{0} \tag{2.35}$$

According to (2.35), at each iteration, the coordinating variables are updated using the current local information. In other words, the price vector and interaction predictions ($\eta = \mathbf{p}, \mathbf{V}$) are simultaneously updated using the current local controllers optimal solutions. When $\eta = \mathbf{p}$, taking derivative of the Lagrangian (2.33) with respect to the price vector and using the predictive form (2.35), gives:

$$\nabla_{\mathbf{p}} L = \sum_{i=1}^{N} \mathbf{\Theta}_i \begin{bmatrix} \mathbf{X}_i^q \\ \mathbf{\Delta U}_i^q \\ \mathbf{V}_i^{q+1} \end{bmatrix} = \mathbf{0} \tag{2.36}$$

Considering (2.12), which describes the relation between the overall interaction error vector $\mathfrak{E}$ (2.11b) and the interaction equations (2.6b), it is clear that (2.36) implies:

$$\mathbf{v}_i^{q+1}(k+l|k) = \sum_{\substack{j=1 \\ j \neq i}}^{N} \left[\beta \mathbf{A}_{ij} \mathbf{x}_j^q(k+l|k) + \mathbf{B}_{ij} \sum_{a=0}^{l} \mathbf{\Delta u}_j^q(k+a|k) \right] \tag{2.37}$$

for $i = 1, \ldots, N$

Taking derivative of the Lagrangian (2.33) with respect to $\mathbf{V}$ and using the iteration

index q based on (2.35), leads to the following set of equations:

$$\nabla_{\mathbf{V}} L = \sum_{i=1}^{N} \nabla_{\mathbf{V}_i} L = 0 \implies$$
$$\mathbf{p}^{q+1} \begin{bmatrix} \mathbf{I}_1 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{I}_N \end{bmatrix} - \begin{bmatrix} \mathbf{\Lambda}_1^q \\ \vdots \\ \mathbf{\Lambda}_N^q \end{bmatrix} - \begin{bmatrix} (\mathbb{C}_{11} \frac{d\mathbf{X}_1}{d\mathbf{V}_1})^T \mathbf{\Omega}_{1,min}^q \\ \vdots \\ (\mathbb{C}_{NN} \frac{d\mathbf{X}_N}{d\mathbf{V}_N})^T \mathbf{\Omega}_{N,min}^q \end{bmatrix} + \begin{bmatrix} (\mathbb{C}_{11} \frac{d\mathbf{X}_1}{d\mathbf{V}_1})^T \mathbf{\Omega}_{1,max}^q \\ \vdots \\ (\mathbb{C}_{NN} \frac{d\mathbf{X}_N}{d\mathbf{V}_N})^T \mathbf{\Omega}_{N,max}^q \end{bmatrix} = \mathbf{0} \tag{2.38}$$

where $\mathbf{I}_i$ is the identity matrix with appropriate dimension.

The derivative term $\frac{d\mathbf{X}_i}{d\mathbf{V}_i}$ is calculated using the equalities in (2.32b):

$$\frac{d\mathbf{X}_i}{d\mathbf{V}_i} = \hat{\mathbb{A}}_{ii}^{-1} \tag{2.39}$$

Substituting (2.39) in (2.38) and re-arranging the terms give:

$$\mathbf{p}^{q+1} = \begin{bmatrix} \mathbf{\Lambda}_1^q \\ \vdots \\ \mathbf{\Lambda}_N^q \end{bmatrix} + \begin{bmatrix} (\mathbb{C}_{11} \hat{\mathbb{A}}_{11}^{-1})^T \mathbf{\Omega}_{1,min}^q \\ \vdots \\ (\mathbb{C}_{NN} \hat{\mathbb{A}}_{NN}^{-1})^T \mathbf{\Omega}_{N,min}^q \end{bmatrix} - \begin{bmatrix} (\mathbb{C}_{11} \hat{\mathbb{A}}_{11}^{-1})^T \mathbf{\Omega}_{1,max}^q \\ \vdots \\ (\mathbb{C}_{NN} \hat{\mathbb{A}}_{NN}^{-1})^T \mathbf{\Omega}_{N,max}^q \end{bmatrix} \tag{2.40}$$

The two sets of update equations in (2.37) and (2.40) construct the coordinator in the IPC method.

Remark 2.4.1 *One approach to numerically solve a system of algebraic equations is to use a fixed-point iteration technique. Another approach is to use gradient-based methods. Note that there is a close relationship between numerically solving a set of equations and unconstrained optimization. Thus, gradient-based approaches can also be used to design the coordinator in the IPC method. In doing so, for solving* $\nabla_{\mathbf{p}} L = \mathbf{0}$ *and* $\nabla_{\mathbf{V}} L = \mathbf{0}$ *the steepest ascent and steepest descent*[16] *methods can be*

[16]From duality point of view, elements of the price vector $\mathbf{p}$ are the so-called *dual variables*, as they are Lagrange multipliers. Dual variables are decision variables of the dual optimization problem. In the context of CDMPC discussed in this work, the primal optimization (plant-wide MPC) is a minimization problem and thus, the corresponding dual optimization is a maximization problem. This implies that if the price vector is calculated by numerically solving the dual optimization problem using a gradient-based method, the search direction will be in the direction of the gradient vector. Consequently, in solving the set of algebraic equations $\nabla_{\mathbf{p}} L = \mathbf{0}$ by a gradient-based method, the same search direction is used.
In the context of duality, elements of the coordinating variable vector $\mathbf{V}_i$ are the so-called *primal variables*, as they are the decision variables of the primal optimization problem. In this work, the primal problem is a minimization problem. In numerically solving minimization problems by a gradient-based method, the search direction is in the negative direction of the gradient. Likewise, for solving $\nabla_{\mathbf{V}} L = \mathbf{0}$ using a gradient-based method, the same search direction (i.e. negative direction of the gradient vector) is used.

used, respectively, to obtain the following coordinator:

$$\mathbf{p}^{q+1} = \mathbf{p}^q + \epsilon_1 (\overbrace{\sum_{i=1}^{N} \mathbf{\Theta}_i \begin{bmatrix} \mathbf{X}_i^q \\ \mathbf{\Delta U}_i^q \\ \mathbf{V}_i^q \end{bmatrix}}^{(\nabla_{\mathbf{p}} L)^q}) \tag{2.41}$$

$$\mathbf{V}^{q+1} = \mathbf{V}^q - \epsilon_2 (\overbrace{\mathbf{p}^q - \begin{bmatrix} \mathbf{\Lambda}_1^q \\ \vdots \\ \mathbf{\Lambda}_N^q \end{bmatrix} - \begin{bmatrix} (\mathbb{C}_{11}\hat{\mathbb{A}}_{11}^{-1})^T \mathbf{\Omega}_{1,min}^q \\ \vdots \\ (\mathbb{C}_{NN}\hat{\mathbb{A}}_{NN}^{-1})^T \mathbf{\Omega}_{N,min}^q \end{bmatrix} + \begin{bmatrix} (\mathbb{C}_{11}\hat{\mathbb{A}}_{11}^{-1})^T \mathbf{\Omega}_{1,max}^q \\ \vdots \\ (\mathbb{C}_{NN}\hat{\mathbb{A}}_{NN}^{-1})^T \mathbf{\Omega}_{N,max}^q \end{bmatrix}}^{(\nabla_{\mathbf{V}} L)^q}) \tag{2.42}$$

where $i = 1, \ldots, N$; and ϵ_1 and ϵ_2 are tuning parameters (Singh and Titli (1978) and Cohen and Miara (1990)).

Similar to the GC approach, the iterative scheme in the coordinator results in a hierarchical structure. At every sampling interval, the coordinator (2.37 and 2.40) and distributed controllers (2.31) exchange information until the coordinator converges. The convergence criterion used to stop the coordination algorithm is $||\mathbf{V}^{q+1} - \mathbf{V}^q|| \leq \mathfrak{e}$[17], in which $\mathfrak{e}$ is the pre-specified error tolerance. Coordinator convergence corresponds to the case where:

- The value of the price vector becomes equal to the value of Lagrange multipliers associated with the interaction equations in the plant-wide MPC problem.
- The interaction equations are satisfied.
- The vector $\mathbf{v}_i$, yields exact values of the local interactions.

Communication between the two levels in the Interaction Prediction Coordination method is systematically done according to Algorithm 2. Also, for more clarification on the information flow between the coordinator and distributed controllers, a flowchart has been provided in Appendix A.

[17]The criterion $||\mathbf{p}^{q+1} - \mathbf{p}^q||$ can also be used along with or instead of $||\mathbf{V}^{q+1} - \mathbf{V}^q|| \leq \mathfrak{e}$; however, since both these criteria correspond to the same solution, using one of them suffices.

Algorithm 2 : Implementation of IPC-DMPC network

1. Coordinator: Iteration counter q is set to 1.
2. Coordinator: Coordinating variables $\mathbf{p}$ and $\mathbf{V}_i$ are arbitrarily initialized.
3. Coordinator: Coordinating variables are sent to the local controllers.
4. Local Controllers: Local optimization problems (2.31) are solved.
5. Local Controllers: Local optimal solutions $\mathbf{X}_i$ and $\boldsymbol{\Delta}\mathbf{U}_i$ and Lagrange multipliers $\boldsymbol{\Lambda}_i$ and/or $\boldsymbol{\Omega}_i$ are sent to the coordinator.
6. Coordinator: Coordinating variables are updated using (2.37) and (2.40).
7. Coordinator: If $||\mathbf{V}^{q+1} - \mathbf{V}^{q}|| \leq \mathfrak{e}$, algorithm stops. Else, next step is taken.
8. Coordinator: Iteration counter is increased by 1.
9. Steps 3-7 are repeated.

2.4.2.1 Coordinator Convergence Accuracy

Convergence of Coordinator Designed by Fixed-Point Iterations

In applying the IPC method in which the coordinator is obtained by fixed-point iteration technique, to linear quadratic (LQ) control problem, Cohen (1977) showed that the coordinator will converge, if the value of the objective function of the aggregate problem decreases monotonously over iterations. The proof is valid only for quadratic objective functions which fits with the MPC formulations used in this thesis. Marcos (2012) extended the proof to unconstrained linear CDMPC and showed that under certain conditions, the monotone decrease of the objective function is achieved. These conditions depend on the dynamic models and weighting matrices that are chosen off-line.

In this work, the same approach is adopted for constrained linear CDMPC, to see if the presence of local inequalities will affect the convergence behaviour of the coordinator. The proof is carried out by investigating if the overall objective function decreases over any two consecutive iterations.

In the plant-wide MPC (2.4), if the predicted states in the objective function and inequality constraints are replaced by $\mathbf{X} = \mathcal{S}(\Delta\mathbf{U})$ obtained from the equality

constraint (2.4b), then problem (2.4) can be described in the following reduced space:

$$\begin{aligned} &\underset{\Delta\mathbf{U}}{min} \quad J_{Plantwide}(\mathcal{S}(\Delta\mathbf{U}), \Delta\mathbf{U}) \\ &\text{subject to} \\ &\mathbb{E}\Delta\mathbf{U} \leq \mathbf{0} \end{aligned} \tag{2.43}$$

where $\mathbb{E}$ is a constant matrix and contains the coefficients resulting from concatenating inequality constraints.

For the aggregate of the distributed controllers (2.32), let the predicted states be replaced by $\mathbf{X}_{DMPC} = \bar{\mathcal{S}}(\Delta\mathbf{U}, \mathbf{V})$. Also, assume that $\mathcal{K}$ is the mapping from $\Delta\mathbf{U}$ into $\mathbf{V}$ that is, $\mathbf{V} = \mathcal{K}(\Delta\mathbf{U})$. Thus, problem (2.32) can be written in the reduced space form:

$$\begin{aligned} &\underset{\Delta\mathbf{U}}{min} \quad J_{CDMPC}(\bar{\mathcal{S}}(\Delta\mathbf{U}, \mathcal{K}(\Delta\mathbf{U})), \Delta\mathbf{U}) \\ &\text{subject to} \\ &\bar{\mathbb{E}}\Delta\mathbf{U} \leq \mathbf{0} \end{aligned} \tag{2.44}$$

Also, the aggregate of distributed controllers (2.32) will act as the plant-wide/centralized MPC controller (2.4), if the following condition is held:

$$\mathcal{S}(\Delta\mathbf{U}) = \bar{\mathcal{S}}(\Delta\mathbf{U}, \mathcal{K}(\Delta\mathbf{U})), \quad \forall\Delta\mathbf{U} \tag{2.45}$$

The two implications from (2.45) are that for all $\Delta\mathbf{U}$:

1. The objective functions of both plant-wide and distributed problems are equal:

$$J_{Plantwide}(\mathcal{S}(\Delta\mathbf{U}), \Delta\mathbf{U}) = J_{CDMPC}(\bar{\mathcal{S}}(\Delta\mathbf{U}, \mathcal{K}(\Delta\mathbf{U})), \Delta\mathbf{U}) \tag{2.46}$$

2. The coefficient matrices in the inequality constraints (2.43) and (2.44) are equal:

$$\mathbb{E} = \bar{\mathbb{E}} \tag{2.47}$$

The Lagrangians and objective functions of (2.43) and (2.44) along with Taylor series expansion provide the tool to quantitatively express the objective function values over two consecutive iterations. Before proceeding further, let the second

derivative of the plant-wide and distributed objective functions be $\mathbf{G} = \frac{d^2 J_{Plantwide}}{d\Delta \mathbf{U}^2}$ and $\mathbf{F} = \frac{d^2 J_{DMPC}}{d\Delta \mathbf{U}^2}$, respectively. The matrix $\mathbf{F}$ contains the diagonal matrices $\mathbf{F}_i$ obtained from second differentiation of the local objective functions.

The Lagrangian for problems (2.43) and (2.44) are:

$$L_{Plantwide} = J_{Plantwide} + \boldsymbol{\chi}^T \mathbb{E} \Delta \mathbf{U} \tag{2.48}$$

$$L_{CDMPC} = J_{CDMPC} + \boldsymbol{\chi}^T \mathbb{E} \Delta \mathbf{U} \tag{2.49}$$

In studying the behaviour of objective function over the iterations, some simple yet lengthy manipulations and calculations are needed. In what follows, for more clarification, the required procedures are described in four steps.

Step 1: Finding a relationship between $\Delta \mathbf{U}^*$, $\Delta \mathbf{U}^q$ and $\Delta \mathbf{U}^{q+1}$:

The Taylor series of the derivative of Lagrangian (2.48) with respect to $\Delta \mathbf{U}$ around the optimal solution $\Delta \mathbf{U}^*$ is:

$$\frac{dL_{Plantwide}}{d\Delta \mathbf{U}} = \frac{dL_{Plantwide}}{d\Delta \mathbf{U}}\bigg|_{\Delta \mathbf{U}^*} + \frac{d^2 L_{Plantwide}}{d\Delta \mathbf{U}^2}\bigg|_{\Delta \mathbf{U}^*} (\Delta \mathbf{U} - \Delta \mathbf{U}^*) \tag{2.50}$$

where the term $\frac{dL_{Plantwide}}{d\Delta \mathbf{U}}\Big|_{\Delta \mathbf{U}^*} = \mathbf{0}$ because $\Delta \mathbf{U}^*$ is the solution of the the first-order optimality condition $\frac{dL_{Plantwide}}{d\Delta \mathbf{U}} = \mathbf{0}$. Also, from (2.48) it is concluded that $\frac{d^2 L_{Plantwide}}{d\Delta \mathbf{U}^2} = \frac{d^2 J_{Plantwide}}{d\Delta \mathbf{U}^2} = \mathbf{G}$. Note that the higher terms in (2.50) are zero because the objective function is quadratic and the constraints are linear; consequently the higher than second order derivatives are zero. Therefore, (2.50) becomes:

$$\frac{dL_{Plantwide}}{d\Delta \mathbf{U}} = \mathbf{G}(\Delta \mathbf{U} - \Delta \mathbf{U}^*) \tag{2.51}$$

Similarly, the Taylor series of the derivative of Lagrangian (2.49) with respect to $\Delta \mathbf{U}$ around the optimal solution of the aggregate problem at iteration $q+1$, $\Delta \mathbf{U}^{q+1}$ is:

$$\frac{dL_{CDMPC}}{d\Delta \mathbf{U}} = \frac{dL_{CDMPC}}{d\Delta \mathbf{U}}\bigg|_{\Delta \mathbf{U}^{q+1}} + \frac{d^2 L_{CDMPC}}{d\Delta \mathbf{U}^2}\bigg|_{\Delta \mathbf{U}^{q+1}} (\Delta \mathbf{U} - \Delta \mathbf{U}^{q+1}) \tag{2.52}$$

where the term $\frac{dL_{CDMPC}}{d\Delta \mathbf{U}}\Big|_{\Delta \mathbf{U}^{q+1}} = \mathbf{0}$ because $\Delta \mathbf{U}^{q+1}$ is the solution of the the first-order optimality condition $\frac{dL_{CDMPC}}{d\Delta \mathbf{U}} = \mathbf{0}$ at iteration $q+1$. Also, from (2.49) it is

concluded that $\frac{d^2L_{CDMPC}}{d\Delta\mathbf{U}^2} = \frac{d^2J_{CDMPC}}{d\Delta\mathbf{U}^2} = \mathbf{F}$. Note that the higher terms in (2.52) are zero for the same reason mentioned above. Therefore, (2.52) becomes:

$$\frac{dL_{CDMPC}}{d\Delta\mathbf{U}} = \mathbf{F}(\Delta\mathbf{U} - \Delta\mathbf{U}^{q+1}) \tag{2.53}$$

Taking the derivatives of Lagrangians (2.48) and (2.49) with respect to $\Delta\mathbf{U}$ gives:

$$\frac{dL_{Plantwide}}{d\Delta\mathbf{U}} = \frac{dJ_{Plantwide}}{d\Delta\mathbf{U}} + \mathbb{E}^T\boldsymbol{\chi} \tag{2.54}$$

$$\frac{dL_{CDMPC}}{d\Delta\mathbf{U}} = \frac{dJ_{CDMPC}}{d\Delta\mathbf{U}} + \mathbb{E}^T\boldsymbol{\chi} \tag{2.55}$$

and from (2.46) it is concluded that $\frac{dL_{Plantwide}}{d\Delta\mathbf{U}} = \frac{dL_{CDMPC}}{d\Delta\mathbf{U}}$. Therefore, for all $\Delta\mathbf{U}$, (2.51) and (2.53) are equal. Consequently, if $\Delta\mathbf{U} = \Delta\mathbf{U}^q$, the following relationship will be obtained:

$$\Delta\mathbf{U}^q - \Delta\mathbf{U}^{q+1} = \mathbf{F}^{-1}\mathbf{G}(\Delta\mathbf{U}^q - \Delta\mathbf{U}^*) \tag{2.56}$$

Step 2: Finding an explicit expression for $\left.\frac{dJ_{Plantwide}}{d\Delta\mathbf{U}}\right|_{\Delta\mathbf{U}^q}$:

From (2.46) the following holds:

$$\left.\frac{dJ_{Plantwide}}{d\Delta\mathbf{U}}\right|_{\Delta\mathbf{U}^q} = \left.\frac{dJ_{DMPC}}{d\Delta\mathbf{U}}\right|_{\Delta\mathbf{U}^q} \tag{2.57}$$

The Taylor series of $\frac{dJ_{DMPC}}{d\Delta\mathbf{U}}$ around $\Delta\mathbf{U}^{q+1}$ is:

$$\frac{dJ_{DMPC}}{d\Delta\mathbf{U}} = \left.\frac{dJ_{DMPC}}{d\Delta\mathbf{U}}\right|_{\Delta\mathbf{U}^{q+1}} + \left.\frac{d^2J_{DMPC}}{d\Delta\mathbf{U}^2}\right|_{\Delta\mathbf{U}^{q+1}} (\Delta\mathbf{U} - \Delta\mathbf{U}^{q+1}) \tag{2.58}$$

where the remaining higher order terms are zero as the objective function is quadratic.

In order to calculate the first term in (2.58), from (2.53) and (2.55) the following relationship is obtained:

$$\begin{aligned} \mathbf{F}(\Delta\mathbf{U} - \Delta\mathbf{U}^{q+1}) &= \frac{dJ_{CDMPC}}{d\Delta\mathbf{U}} + \mathbb{E}^T\boldsymbol{\chi} \\ \Longrightarrow \quad \frac{dJ_{CDMPC}}{d\Delta\mathbf{U}} &= \mathbf{F}(\Delta\mathbf{U} - \Delta\mathbf{U}^{q+1}) - \mathbb{E}^T\boldsymbol{\chi} \end{aligned} \tag{2.59}$$

Thus, for $\Delta\mathbf{U} = \Delta\mathbf{U}^q$, equation (2.59) yields:

$$\left.\frac{dJ_{CDMPC}}{d\Delta\mathbf{U}}\right|_{\Delta\mathbf{U}^q} = \mathbf{F}(\Delta\mathbf{U}^q - \Delta\mathbf{U}^{q+1}) - \mathbb{E}^T\boldsymbol{\chi}^q \tag{2.60}$$

Finally, from (2.57) and (2.60), the desired expression is obtained as:

$$\left.\frac{dJ_{Plantwide}}{d\Delta\mathbf{U}}\right|_{\Delta\mathbf{U}^q} = \mathbf{F}(\Delta\mathbf{U}^q - \Delta\mathbf{U}^{q+1}) - \mathbb{E}^T\boldsymbol{\chi}^q \tag{2.61}$$

Step 3: Writing the Taylor series of $J_{Plantwide}$ around $\Delta\mathbf{U}^q$:

Now, the Taylor series expansion of the plant-wide objective function around $\Delta\mathbf{U}^q$, is formed as:

$$\begin{aligned} J_{Plantwide}(\Delta\mathbf{U}) = J_{Plantwide}(\Delta\mathbf{U}^q) &+ \left.\frac{dJ_{Plantwide}}{d\Delta\mathbf{U}}\right|_{\Delta\mathbf{U}^q}^T (\Delta\mathbf{U} - \Delta\mathbf{U}^q) \\ &+ (\Delta\mathbf{U} - \Delta\mathbf{U}^q)\frac{1}{2}\left.\frac{d^2J_{Plantwide}}{d\Delta\mathbf{U}^2}\right|_{\Delta\mathbf{U}^q}^T (\Delta\mathbf{U} - \Delta\mathbf{U}^q) \end{aligned} \tag{2.62}$$

where the remaining higher order terms are zero.

By replacing $\left.\frac{dJ_{Plantwide}}{d\Delta\mathbf{U}}\right|_{\Delta\mathbf{U}^q}$ and $\left.\frac{d^2J_{Plantwide}}{d\Delta\mathbf{U}^2}\right|_{\Delta\mathbf{U}^q}$ by (2.62) and $\mathbf{G}$, respectively, the following expression for $J_{Plantwide}$ is achieved:

$$\begin{aligned} J_{Plantwide}(\Delta\mathbf{U}) = J_{Plantwide}(\Delta\mathbf{U}^q) &\\ + (\Delta\mathbf{U}^q - \Delta\mathbf{U}^{q+1})^T\mathbf{F}(\Delta\mathbf{U} - \Delta\mathbf{U}^q) - (\boldsymbol{\chi}^q)^T\mathbb{E}^T(\Delta\mathbf{U} - \Delta\mathbf{U}^q) &\\ + (\Delta\mathbf{U} - \Delta\mathbf{U}^q)\frac{\mathbf{G}}{2}(\Delta\mathbf{U} - \Delta\mathbf{U}^q) & \end{aligned} \tag{2.63}$$

Step 4: Calculating $J_{Plantwide}(\Delta\mathbf{U}^q) - J_{Plantwide}(\Delta\mathbf{U}^{q+1})$:

Equality (2.46) implies that studying the trends of the objective function in the aggregate problem (2.44) over the iterations, is equivalent to studying changes of the objective function in the plant-wide problem (2.43). Therefore, using (2.63), the difference between the values of plant-wide objective function evaluated for the solution obtained at iterations q and $q+1$[18], becomes:

$$\begin{aligned} &J_{Plantwide}(\Delta\mathbf{U}^q) - J_{Plantwide}(\Delta\mathbf{U}^{q+1}) = \\ &(\Delta\mathbf{U}^q - \Delta\mathbf{U}^{q+1})^T\mathbf{F}(\Delta\mathbf{U}^q - \Delta\mathbf{U}^{q+1}) - (\boldsymbol{\chi}^{q+1})^T\mathbb{E}^T(\Delta\mathbf{U}^q - \Delta\mathbf{U}^{q+1}) \\ &\qquad - (\Delta\mathbf{U}^q - \Delta\mathbf{U}^{q+1})^T\frac{\mathbf{G}}{2}(\Delta\mathbf{U}^q - \Delta\mathbf{U}^{q+1}) \end{aligned} \tag{2.64}$$

[18]Note that the coordinator updates the coordinating variables. Then using these updated values, the solution of the aggregate problem at iteration q, i.e. $\Delta\mathbf{U}^q$, is calculated.

Using (2.56), (2.64) becomes:

$$J_{Plantwide}(\Delta\mathbf{U}^q) - J_{Plantwide}(\Delta\mathbf{U}^{q+1}) =$$
$$(\Delta\mathbf{U}^q - \Delta\mathbf{U}^*)^T(\mathbf{B}^{-1}\mathbf{G})^T\mathbf{F}(\mathbf{B}^{-1}\mathbf{G})(\Delta\mathbf{U}^q - \Delta\mathbf{U}^*) - (\boldsymbol{\chi}^{q+1})^T\mathbb{E}^T(\mathbf{B}^{-1}\mathbf{G})(\Delta\mathbf{U}^q - \Delta\mathbf{U}^*)$$
$$- (\Delta\mathbf{U}^q - \Delta\mathbf{U}^*)^T(\mathbf{B}^{-1}\mathbf{G})^T\frac{\mathbf{G}}{2}(\mathbf{B}^{-1}\mathbf{G})(\Delta\mathbf{U}^q - \Delta\mathbf{U}^*)$$

$\Longrightarrow$

$$J_{Plantwide}(\Delta\mathbf{U}^q) - J_{Plantwide}(\Delta\mathbf{U}^{q+1}) =$$
$$(\Delta\mathbf{U}^q - \Delta\mathbf{U}^*)^T\left[(\mathbf{B}^{-1}\mathbf{G})^T(\mathbf{F} - \frac{\mathbf{G}}{2})(\mathbf{B}^{-1}\mathbf{G})\right](\Delta\mathbf{U}^q - \Delta\mathbf{U}^*) \tag{2.65}$$
$$- (\boldsymbol{\chi}^{q+1})^T\mathbb{E}^T(\mathbf{B}^{-1}\mathbf{G})(\Delta\mathbf{U}^q - \Delta\mathbf{U}^*)$$

For the overall objective function of the distributed controllers, or equivalently the objective function of the plant-wide problem, to have a decreasing trend over the iterations, the right-hand side of (2.65) should be positive. For both plant-wide and distributed MPC, as they are both QP, if $\mathbf{G}$ and $\mathbf{F}_i$ are positive definite, a unique solution exists[19], which means that the optimality conditions are met. Thus, if inequality constraints exist, the non-negativity condition $\boldsymbol{\chi}^{q+1} \geq \mathbf{0}$, is satisfied. Based on the value of $\boldsymbol{\chi}^{q+1}$ three conclusions can be made:

1. If no inequality constraints are used in the MPC controllers, $\boldsymbol{\chi}^{q+1} = \mathbf{0}$ and the second term in the right-hand side of (2.65) is zero. Therefore, if $\mathbf{F} - \frac{\mathbf{G}}{2} \geq 0$, the first term in the right-hand side of (2.65) will be positive and thus, the overall objective function will decrease over the iterations (i.e., $J_{Plantwide}(\Delta\mathbf{U}^q) - J_{Plantwide}(\Delta\mathbf{U}^{q+1}) \geq 0$) and the coordination algorithm converges. This is the same result obtained by Marcos (2012).

2. If inequality constraints are used in the MPC controllers and if they are inactive (i.e., $\boldsymbol{\chi}^{q+1} = \mathbf{0}$), similar to the case of no inequalities, the coordinator will converge, if $\mathbf{F} - \frac{\mathbf{G}}{2} \geq 0$.

3. If inequality constraints are used in the MPC controllers and if they are active (i.e., $\boldsymbol{\chi}^{q+1} \geq \mathbf{0}$), then no conclusions can be made on the sign of the right-hand side of (2.65) and consequently on the convergence of the coordinator.

[19]Before designing an MPC, this assumption is implicitly taken into account by properly choosing the weighting matrices.

Therefore, if local inequalities exist in the CDMPC, proving convergence of the coordination algorithm when fixed-point iteration technique is used, cannot be done by the above method, as it is not possible to guarantee having inactive inequalities over the iterations at each sampling instant k.

2.4.2.2 Convergence of Coordinator Designed by Gradient-based Algorithm

If a gradient algorithm is used to obtain the coordinator, convergence of the coordination algorithm can be studied by defining an appropriate Lyapunov function.

Before proceeding with the prove, the following simpler notation is introduced:

1. The variables involved in the optimality conditions of the distributed controllers (2.31) are represented by vector $\mathbf{Z}_i \triangleq \left[\mathbf{X}_i^T, \boldsymbol{\Delta}\mathbf{U}_i^T, \boldsymbol{\Lambda}_i^T, \boldsymbol{\Omega}_i^T, \boldsymbol{\Upsilon}_i^T\right]^T$. Consequently, $\mathbf{Z} = \left[\mathbf{Z}_1^T, \dots \mathbf{Z}_N^T\right]^T$.
2. The coordinating variables are shown by the vector $\mathbf{W} \triangleq \left[\mathbf{p}^T, \overbrace{\mathbf{V}_1^T, \dots, \mathbf{V}_N^T}^{\mathbf{V}}\right]^T$.
3. The Lagrangian for the aggregate of the CDMPC is shown by $L(\mathbf{Z}, \mathbf{W})$.
4. The gradients of the Lagrangian with respect to the coordinating variables are shown by the vector $\nabla_{\mathbf{W}} L \triangleq \left[\nabla_{\mathbf{p}} L^T, -\nabla_{\mathbf{V}} L^T\right]^T$.

As suggested by Singh and Titli (1978), the following Lyapunov function is defined for the convergence proof of the gradient-based coordination algorithm:

$$\mathcal{L} \triangleq \frac{1}{2}(\nabla_{\mathbf{W}} L)^T (\nabla_{\mathbf{W}} L) \tag{2.66}$$

The function $\mathcal{L}$ is zero when $\mathbf{W} = \mathbf{W}^*$, where $\mathbf{W}^*$ is the optimal centralized solution, and positive definite (i.e., $\mathcal{L} > 0$) when $\mathbf{W} \neq \mathbf{W}^*$. The derivative of $\mathcal{L}$ is:

$$\dot{\mathcal{L}} = \frac{d\mathcal{L}}{dq} = (\nabla_{\mathbf{W}} L)^T \frac{d\nabla_{\mathbf{W}} L}{dq} = (\nabla_{\mathbf{W}} L)^T \frac{d\nabla_{\mathbf{W}} L}{d\mathbf{W}} \frac{d\mathbf{W}}{dq} \tag{2.67}$$

The use of gradient-based methods to find $\mathbf{W} = \left[\mathbf{p}^T, \mathbf{V}^T\right]^T$ that solves $\nabla_{\mathbf{p}} L = \mathbf{0}$ and $\nabla_{\mathbf{V}} L = \mathbf{0}$, implies that the changes in $\mathbf{W}$ over the iterations is proportional to

$\nabla_{\mathbf{W}} L$. Thus, it can be assumed that:

$$\frac{d\mathbf{W}}{dq} = \nabla_{\mathbf{W}} L \tag{2.68}$$

If (2.68) is used in (2.67), then the sign of $\dot{\mathcal{L}}$ will depend on the sign of $\frac{d\nabla_{\mathbf{W}} L}{d\mathbf{W}}$. In order to calculate $\frac{d\nabla_{\mathbf{W}} L}{d\mathbf{W}}$ in (2.67), optimality conditions of the aggregate of the distributed controllers are used. The optimality conditions that the coordinator is responsible to satisfy *during the iterations*, is $\nabla_{\mathbf{W}} L = \bar{\mathcal{S}}(\mathbf{Z}, \mathbf{W}) = \mathbf{0}$, where the function $\bar{\mathcal{S}}$ denotes that optimality conditions depend on both local and coordinating variables. Therefore, differentiating $\nabla_{\mathbf{W}} L = \bar{\mathcal{S}}(\mathbf{Z}, \mathbf{W})$ results in:

$$\begin{aligned} d\nabla_{\mathbf{W}} L &= (\nabla^2_{\mathbf{WW}} L) d\mathbf{W} + (\nabla^2_{\mathbf{WZ}} L) d\mathbf{Z} \\ \implies \frac{d\nabla_{\mathbf{W}} L}{d\mathbf{W}} &= (\nabla^2_{\mathbf{WW}} L) + (\nabla^2_{\mathbf{WZ}} L) \frac{d\mathbf{Z}}{d\mathbf{W}} \end{aligned} \tag{2.69}$$

Because the interaction equations are linear, *at each iteration*, $\nabla^2_{\mathbf{WW}} L = \mathbf{0}$. Therefore, (2.69) becomes:

$$\frac{d\nabla_{\mathbf{W}} L}{d\mathbf{W}} = (\nabla^2_{\mathbf{WZ}} L) \frac{d\mathbf{Z}}{d\mathbf{W}} \tag{2.70}$$

Also, *at each iteration*, optimality conditions of the local controllers are satisfied, that is:

$$\nabla_{\mathbf{Z}} L = \mathcal{S}(\mathbf{Z}, \mathbf{W}) = \mathbf{0} \tag{2.71}$$

where the function $\mathcal{S}$ indicates that optimality conditions of the local controllers depend on both local and coordinating variables. Thus, differentiating (2.71) yields:

$$\begin{aligned} (\nabla^2_{\mathbf{ZZ}} L) d\mathbf{Z} + (\nabla^2_{\mathbf{ZW}} L) d\mathbf{W} &= \mathbf{0} \\ \implies \frac{d\mathbf{Z}}{d\mathbf{W}} &= -(\nabla^2_{\mathbf{ZZ}} L)^{-1} (\nabla^2_{\mathbf{ZW}} L) \end{aligned} \tag{2.72}$$

If the chosen weighting matrices in the objective functions of distributed controllers, are positive definite and a feasible region exists, the local QP problems will have a solution and thus, $(\nabla^2_{\mathbf{ZZ}} L)$ will be non-singular and thus, invertible.

Combining (2.67), (2.68), (2.70) and (2.72) obtains:

$$\dot{\mathcal{L}} = -(\nabla_{\mathbf{W}} L)^T (\nabla^2_{\mathbf{WZ}} L)(\nabla^2_{\mathbf{ZZ}} L)^{-1} (\nabla^2_{\mathbf{ZW}} L)(\nabla_{\mathbf{W}} L) \tag{2.73}$$

From (2.73), it can be concluded that $\dot{\mathcal{L}}$ will be negative definite only if the matrix $\mathbb{S} \triangleq (\nabla^2_{\mathbf{WZ}} L)(\nabla^2_{\mathbf{ZZ}} L)^{-1}(\nabla^2_{\mathbf{ZW}} L)$ which is the Schur Complement of the matrix $\bar{\mathbb{S}} = \begin{bmatrix} \nabla^2_{\mathbf{ZZ}} L & \nabla^2_{\mathbf{WZ}} L \\ \nabla^2_{\mathbf{ZW}} L & \mathbf{0} \end{bmatrix}$[20], is positive definite. Thus, if this condition is met, the coordinator (2.41 and 2.42) will converge to the plant-wide solution.

2.5 Modified Pseudo-Model Coordinated Distributed MPC

In this section, for the first time, the MPMC method is extended to the distributed linear constrained MPC problem. The coordinating terms $\{CoT\}_i$, are derived and the coordinator is designed by numerically solving portions of the optimality conditions. A different type of coordinating terms and coordinating variables appear in the distributed controllers (2.8) because of the strategy used in dealing with the interaction equations. Accordingly, the coordinator is designed such that it gives updates for the resulting coordinating variables.

2.5.1 Pseudo-variables and Distributed Controllers

As was briefly mentioned in section 2.1, the main idea in the Pseudo-Model Coordination method is definition of desirable interconnection variables and then, penalization of deviations of these defined variables from their corresponding local variables. This is done by defining pseudo-variables that are *temporarily fixed in value*, for the unknown interaction variables $\mathbf{x}_j(k+l|k)$ and $\mathbf{\Delta u}_j(k+b|k)$. The pseudo-variables $\tilde{\mathbf{x}}_i$ and $\mathbf{\Delta}\tilde{\mathbf{u}}_i$

$$\tilde{\mathbf{x}}_i(k) \triangleq \mathbf{x}_i(k) \tag{2.74a}$$

$$\mathbf{\Delta}\tilde{\mathbf{u}}_i(k) \triangleq \mathbf{\Delta u}_i(k) \tag{2.74b}$$

are introduced. These pseudo-variables are used to represent the interaction variables in the interaction equations (2.6b). Also, it is desired that the value of each pseudo-variable $\tilde{\mathbf{x}}_i(k+l+1|k)$ and $\mathbf{\Delta}\tilde{\mathbf{u}}_i(k+b|k)$ be equal to the value of $\mathbf{x}_i(k+l+1|k)$ and

[20] Using the Lagrangian (2.33) to calculate the elements of the matrix $\mathbb{S}$, it is seen that these elements include the weighting matrices, the local and interaction models $\mathbf{A}_{ij}$ and $\mathbf{B}_{ij}$, for $i, j = 1, \ldots, N$.

$\boldsymbol{\Delta}\mathbf{u}_i(k+b|k)$, respectively. Therefore, the plant-wide optimization problem (2.4)[21] can be written as:

$$\min_{\mathbf{X}_i, \boldsymbol{\Delta}\mathbf{U}_i} \quad \sum_{i=1}^{N} \Big((\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i)^T \mathbb{Q}_{ii} (\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i \Big) \tag{2.75a}$$

subject to

$$\begin{aligned} \mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii} \left[\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1) \right] \\ + \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\alpha \mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1) \right] + \mathbf{v}_i(k+l|k) \end{aligned} \tag{2.75b}$$

$$\mathbf{v}_i(k+l|k) = \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\beta \mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+l|k) + \mathbf{B}_{ij} \sum_{a=0}^{l} \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+a|k) \right] \tag{2.75c}$$

$$\tilde{\mathbf{x}}_i(k+l|k) = \mathbf{x}_i(k+l|k) \tag{2.75d}$$

$$\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+b|k) = \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \tag{2.75e}$$

$$\begin{aligned} &\mathbf{y}_i^{min}(k+l+1) \leq \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \leq \mathbf{y}_i^{max}(k+l+1) \\ &\mathbf{u}_i^{min}(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}_i^{max}(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \leq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \leq \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b) \end{aligned} \tag{2.75f}$$

Next, the performance index (2.75a) is augmented with the *transformed interaction equations* (Sorenson and Koble (1984)) (2.75d) and (2.75e), by using quadratic penalty terms with penalty parameter ε and block-wise weighting matrices $\bar{\mathbb{Q}}_i$ and $\bar{\mathbb{R}}_i$. That is, in the Pseudo-Model Coordination method, the coordinating terms are defined by relaxing these two sets of equations using quadratic penalty functions (Pearson (1971), Singh (1975)); however, as was discussed in section 2.1, the resulting aggregate problem is an approximation of the plant-wide problem (2.75) unless a very large penalty parameter value is used (i.e, $\varepsilon \to \infty$). In other words, for small values of the penalty parameter, the equivalence between the aggregate distributed local optimization problems and the centralized problem is not retained.

[21] In order to design any coordinated distributed network, interaction models are required. Thus, with the presumption of availability of the interaction models, the three centralized, distributed and decentralized schemes can easily be converted into one another. This implies that by including the interaction models, the already existing decentralized MPC network can be converted into the plant-wide MPC (2.4) and there is no need to have a physical centralized MPC available.

Sorenson and Koble (1984) overcame this drawback by keeping the transformed equations as explicit constraints in addition to quadratically penalizing them. Then, they formed a relaxed problem by appending the transformed interaction equations to the objective functions of the local controllers. In the current work, the same approach is applied to (2.75), so that the following relaxed problem is formed:

$$\begin{aligned}\min_{\mathbf{X}_i,\boldsymbol{\Delta}\mathbf{U}_i} \quad & \sum_{i=1}^{N}\Big((\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)^T\,\mathbb{Q}_{ii}\,(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\Big)+\\ & \sum_{i=1}^{N}\frac{\varepsilon}{2}\left(||\mathbf{X}_i-\tilde{\mathbf{X}}_i||^2_{\mathbb{Q}_i}+||\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\mathbb{R}_i}\right)+ \qquad (2.76a)\\ & \sum_{i=1}^{N}\Big(\boldsymbol{\Gamma}_i^T(\mathbf{X}_i-\tilde{\mathbf{X}}_i)+\boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i)\Big)\end{aligned}$$

subject to

$$\begin{aligned}\mathbf{x}_i(k+l+1|k)=\mathbf{A}_{ii}\mathbf{x}_i(k+l|k)+\mathbf{B}_{ii}\left[\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k)+\mathbf{u}_i(k-1)\right]\\ +\sum_{\substack{j=1\\ j\neq i}}^{N}\left[\alpha\mathbf{A}_{ij}\mathbf{x}_j(k+l|k)+\mathbf{B}_{ij}\mathbf{u}_j(k-1)\right]+\mathbf{v}_i(k+l|k)\end{aligned} \qquad (2.76b)$$

$$\mathbf{v}_i(k+l|k)=\sum_{\substack{j=1\\ j\neq i}}^{N}\left[\beta\mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+l|k)+\mathbf{B}_{ij}\sum_{a=0}^{l}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+a|k)\right] \qquad (2.76c)$$

$$\begin{aligned}&\mathbf{y}_i^{min}(k+l+1)\leq\mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k)\leq\mathbf{y}_i^{max}(k+l+1)\\ &\mathbf{u}_i^{min}(k+b)\leq\mathbf{u}_i(k+b|k)\leq\mathbf{u}_i^{max}(k+b)\\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b)\leq\boldsymbol{\Delta}\mathbf{u}_i(k+b|k)\leq\boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b)\end{aligned} \qquad (2.76d)$$

where $\tilde{\mathbf{X}}=[\tilde{\mathbf{X}}_1^T,\ldots,\tilde{\mathbf{X}}_N^T]^T$ and $\tilde{\mathbf{X}}_i=[\tilde{\mathbf{x}}_i(k+1|k)^T,\ldots,\tilde{\mathbf{x}}_i(k+H_p-1|k)^T]^T$; similarly, $\boldsymbol{\Delta}\tilde{\mathbf{U}}=[\boldsymbol{\Delta}\tilde{\mathbf{U}}_1^T,\ldots,\boldsymbol{\Delta}\tilde{\mathbf{U}}_N^T]^T$ and $\boldsymbol{\Delta}\tilde{\mathbf{U}}_i=[\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k)^T,\ldots,\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k)^T]^T$; the vectors $\boldsymbol{\Gamma}_i$ and $\boldsymbol{\Pi}_i$ are the Lagrange multiplier vectors associated with the pseudo-variable equations. As was stated before, the pseudo-variables $\tilde{\mathbf{x}}_j(k+l+1|k)$ and $\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+b|k)$ are temporarily fixed in value, which indicates that they are among the coordinating variables and their values are determined by the coordinator. This also implies that the set of equations (2.76c) will entirely be handled by the coordinator as it only depends on the pseudo-variables. Optimization problem (2.76) represents

aggregate of the following CDMPC:

$$\begin{aligned}\min_{\mathbf{X}_i,\boldsymbol{\Delta}\mathbf{U}_i} \quad & (\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)^T\,\mathbb{Q}_{ii}\,(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i+\\ & \frac{\varepsilon}{2}\left(||\mathbf{X}_i-\tilde{\mathbf{X}}_i||^2_{\mathbb{Q}_i}+||\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\mathbb{R}_i}\right)+\\ & \boldsymbol{\Gamma}_i^T(\mathbf{X}_i-\tilde{\mathbf{X}}_i)+\boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i)\end{aligned} \tag{2.77a}$$

subject to

$$\begin{aligned}\mathbf{x}_i(k+l+1|k)=\mathbf{A}_{ii}\mathbf{x}_i(k+l|k)+\mathbf{B}_{ii}\left[\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k)+\mathbf{u}_i(k-1)\right]\\ +\sum_{\substack{j=1\\ j\neq i}}^{N}\left[\alpha\mathbf{A}_{ij}\mathbf{x}_j(k+l|k)+\mathbf{B}_{ij}\mathbf{u}_j(k-1)\right]+\mathbf{v}_i(k+l|k)\end{aligned} \tag{2.77b}$$

$$\begin{aligned}&\mathbf{y}_i^{min}(k+l+1)\le\mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k)\le\mathbf{y}_i^{max}(k+l+1)\\ &\mathbf{u}_i^{min}(k+b)\le\mathbf{u}_i(k+b|k)\le\mathbf{u}_i^{max}(k+b)\\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b)\le\boldsymbol{\Delta}\mathbf{u}_i(k+b|k)\le\boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b)\end{aligned} \tag{2.77c}$$

where $\frac{\varepsilon}{2}||\mathbf{X}_i-\tilde{\mathbf{X}}_i||^2_{\mathbb{Q}_i}$, $\frac{\varepsilon}{2}||\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\mathbb{R}_i}$, $\boldsymbol{\Gamma}_i^T(\mathbf{X}_i-\tilde{\mathbf{X}}_i)$ and $\boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i)$ are the four coordinating terms that establish the connection between the controllers and the coordinator; the coordinating variables $\boldsymbol{\Gamma}_i$ and $\boldsymbol{\Pi}_i$ have a similar role as the price vector $\mathbf{p}$ has in the GC and IPC Methods and thus, they shall be called prices as well; these prices along with the pseudo-variable vectors $\tilde{\mathbf{X}}_i$ and $\boldsymbol{\Delta}\tilde{\mathbf{U}}_i$ and the interaction variables $\mathbf{V}_i$ form the coordinating variables in the MPMC-DMPC.

2.5.2 Coordinator Design in the MPMC Method

Similar to the previous two coordination methods, after $\{CoT\}_i$ and the coordinating variables are defined, the next step is to use an appropriate solution strategy to solve the aggregate distributed problem (2.76). The adopted solution strategy ultimately designs the coordinator by providing update equations for the coordinating variables. In the MPMC method, the solution strategy is similar to the one used in the IPC method, which means that relevant portions of optimality conditions of the aggregate problem are solved by choosing a suitable numerical technique.

In order to make designing the coordinator simpler by using fewer variables, the aggregate of the CDMPC (2.76) combined with interaction equations, is written over

the entire prediction and control horizons and then re-stated in the following compact form:

$$\begin{aligned}
\min_{\mathbf{X}_i,\boldsymbol{\Delta}\mathbf{U}_i} \quad & \sum_{i=1}^{N}\Big((\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)^T\,\mathbb{Q}_{ii}\,(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\Big)+\\
& \sum_{i=1}^{N}\frac{\varepsilon}{2}\Big(||\mathbf{X}_i-\tilde{\mathbf{X}}_i||^2_{\bar{\mathbb{Q}}_i}+||\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\bar{\mathbb{R}}_i}\Big)+ \qquad (2.78a)\\
& \sum_{i=1}^{N}\Big(\boldsymbol{\Gamma}_i^T(\mathbf{X}_i-\tilde{\mathbf{X}}_i)+\boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i)\Big)\\
\text{subject to}\\
& \hat{\mathbb{A}}_{ii}\mathbf{X}_i+\hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i+\sum_{\substack{j=1\\j\neq i}}^{N}\Big[\hat{\mathbb{A}}_{ij}\tilde{\mathbf{X}}_j+\hat{\mathbb{B}}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{U}}_j\Big]=\\
& \overbrace{\bar{\mathbb{A}}_{ii}\mathbf{x}_i(k)+\bar{\mathbb{B}}_{ii}\mathbf{u}_i(k-1)+\sum_{\substack{j=1\\j\neq i}}^{N}\big[\bar{\mathbb{A}}_{ij}\mathbf{x}_j(k)+\bar{\mathbb{B}}_{ij}\mathbf{u}_j(k-1)\big]}^{\mathcal{Z}_i} \qquad (2.78b)\\
& \mathbf{Y}_i^{min}\leq\mathbb{C}_{ii}\mathbf{X}_i\leq\mathbf{Y}_i^{max}\\
& \mathfrak{U}_i^{min}\leq\mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\leq\mathfrak{U}_i^{max} \qquad (2.78c)\\
& i=1,\ldots,N
\end{aligned}$$

where details on obtaining the matrices containing the local and interaction models, have been provided in Appendix B (section B.4). It should be once again emphasized that, the compact form (2.78) is used to make deriving the coordinator's formulation simpler to follow. For implementation purposes, only the CDMPCs (2.77) and the to-be-obtained coordinator are needed.

Since portions of first-order optimality conditions will be used to design the coordinator in the MPMC approach, the first step would be to form the Lagrangian of the aggregate problem (2.78):

$$\begin{aligned}
& L(\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\tilde{\mathbf{X}},\boldsymbol{\Delta}\tilde{\mathbf{U}},\boldsymbol{\Lambda},\boldsymbol{\Omega},\boldsymbol{\Upsilon},\boldsymbol{\Gamma},\boldsymbol{\Pi})=\sum_{i=1}^{N}L_i(\mathbf{X}_i,\boldsymbol{\Delta}\mathbf{U}_i,\tilde{\mathbf{X}}_i,\boldsymbol{\Delta}\tilde{\mathbf{U}}_i,\boldsymbol{\Lambda}_i,\boldsymbol{\Omega}_i,\boldsymbol{\Upsilon}_i,\boldsymbol{\Gamma}_i,\boldsymbol{\Pi}_i)\\
&=\sum_{i=1}^{N}\Big((\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)^T\,\mathbb{Q}_{ii}\,(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\Big)
\end{aligned}$$

$$
\begin{aligned}
&+\sum_{i=1}^{N}\frac{\varepsilon}{2}\left(||\mathbf{X}_i-\tilde{\mathbf{X}}_i||^2_{\mathbb{Q}_i}+||\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\mathbb{R}_i}\right)\\
&+\sum_{i=1}^{N}\left(\boldsymbol{\Gamma}_i^T(\mathbf{X}_i-\tilde{\mathbf{X}}_i)+\boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i)\right)\\
&+\sum_{i=1}^{N}\boldsymbol{\Lambda}_i^T\left(\hat{\mathbb{A}}_{ii}\mathbf{X}_i+\hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i+\sum_{\substack{j=1\\j\neq i}}^{N}\left[\hat{\mathbb{A}}_{ij}\tilde{\mathbf{X}}_j+\hat{\mathbb{B}}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{U}}_j\right]-\mathcal{Z}_i\right)\\
&+\sum_{i=1}^{N}\boldsymbol{\Omega}_{i,min}^T\left(\mathbf{Y}_i^{min}-\mathbb{C}_{ii}\mathbf{X}_i\right)+\boldsymbol{\Omega}_{i,max}^T\left(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{Y}_i^{max}\right)\\
&+\sum_{i=1}^{N}\boldsymbol{\Upsilon}_{i,min}^T\left(\mathfrak{U}_i^{min}-\mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\right)+\boldsymbol{\Upsilon}_{i,max}^T\left(\mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i-\mathfrak{U}_i^{max}\right)
\end{aligned}
\tag{2.79}
$$

Using (2.79), the optimality conditions[22] are obtained as below:

$$
\begin{aligned}
&\nabla_\eta L(\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\tilde{\mathbf{X}},\boldsymbol{\Delta}\tilde{\mathbf{U}},\boldsymbol{\Lambda},\boldsymbol{\Omega},\boldsymbol{\Upsilon},\boldsymbol{\Gamma},\boldsymbol{\Pi})=\mathbf{0}\\
&\eta=\{\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\tilde{\mathbf{X}},\boldsymbol{\Delta}\tilde{\mathbf{U}},\boldsymbol{\Lambda},\boldsymbol{\Gamma},\boldsymbol{\Pi}\}
\end{aligned}
\tag{2.80}
$$

Of the equation set (2.80), $\nabla_{\mathbf{X}}L=\mathbf{0}$, $\nabla_{\boldsymbol{\Delta}\mathbf{U}}L=\mathbf{0}$, $\nabla_{\boldsymbol{\Lambda}}L=\mathbf{0}$, along with the feasibility, complementary slackness and non-negativity conditions resulting from the local inequality constraints are taken into account in the optimization problem of CDMPC (2.77). The remaining optimality conditions in (2.80) i.e., $\nabla_{\tilde{\mathbf{X}}}L=\mathbf{0}$,

[22] The precise statement of the necessary KKT conditions for the aggregate problem (2.78) is:

Optimality: $\nabla_{\mathbf{X}}L=\mathbf{0}$, $\nabla_{\boldsymbol{\Delta}\mathbf{U}}L=\mathbf{0}$, $\boxed{\nabla_{\tilde{\mathbf{X}}}L=\mathbf{0}}$ and $\boxed{\nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}}L=\mathbf{0}}$

Feasibilty: $\begin{cases}\nabla_{\boldsymbol{\Lambda}}L=\mathbf{0}\ ,\ \boxed{\nabla_{\boldsymbol{\Gamma}}L=\mathbf{0}} \text{ and } \boxed{\nabla_{\boldsymbol{\Pi}}L=\mathbf{0}}\\ \begin{bmatrix}\mathbf{Y}^{min}-\mathbb{C}\mathbf{X}\\ \mathbb{C}\mathbf{X}-\mathbf{Y}^{max}\\ \mathfrak{U}^{min}-\mathbb{U}\Delta\mathbf{U}\\ \mathbb{U}\Delta\mathbf{U}-\mathfrak{U}^{max}\end{bmatrix}\leq\mathbf{0}\end{cases}$

Complementary slackness: $\begin{bmatrix}\boldsymbol{\Omega}\\ \boldsymbol{\Upsilon}\end{bmatrix}^T\begin{bmatrix}\mathbf{Y}^{min}-\mathbb{C}\mathbf{X}\\ \mathbb{C}\mathbf{X}-\mathbf{Y}^{max}\\ \mathfrak{U}^{min}-\mathbb{U}\Delta\mathbf{U}\\ \mathbb{U}\Delta\mathbf{U}-\mathfrak{U}^{max}\end{bmatrix}=\mathbf{0}$

Non-negativity: $\begin{bmatrix}\boldsymbol{\Omega}\\ \boldsymbol{\Upsilon}\end{bmatrix}\geq\mathbf{0}$

Throughout this thesis, in order to use simpler language, wherever the term *optimality conditions* is used for the coordinator design in the MPMC method, it refers to the above Optimality and Feasibility conditions included in boxes.

$\nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}} L = \mathbf{0}$, $\nabla_{\boldsymbol{\Gamma}} L = \mathbf{0}$ and $\nabla_{\boldsymbol{\Pi}} L = \mathbf{0}$ are used to design the coordinator. The use of fixed-point iteration method in solving the algebraic equations resulting from the optimality conditions, produces the following iterative form of (2.80):

$$\nabla_{\eta} L(\mathbf{X}^q, \boldsymbol{\Delta}\mathbf{U}^q, \tilde{\mathbf{X}}^{q+1}, \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}, \boldsymbol{\Lambda}^q, \boldsymbol{\Omega}^q, \boldsymbol{\Upsilon}^q, \boldsymbol{\Gamma}^{q+1}, \boldsymbol{\Pi}^{q+1}) = \mathbf{0} \tag{2.81}$$

Taking the derivative of Lagrangian (2.79) with respect to the prices $\boldsymbol{\Omega}$ and $\boldsymbol{\Pi}$, setting the results to zero and solving the resulting equations yield the following update formulae for the pseudo-variables $\tilde{\mathbf{X}}$ and $\boldsymbol{\Delta}\tilde{\mathbf{U}}$:

$$\begin{cases} \nabla_{\boldsymbol{\Gamma}} L = \sum_{i=1}^{N} \nabla_{\boldsymbol{\Gamma}_i} L = \mathbf{0} \\ \nabla_{\boldsymbol{\Pi}} L = \sum_{i=1}^{N} \nabla_{\boldsymbol{\Pi}_i} L = \mathbf{0} \end{cases} \tag{2.82a}$$

$$\Longrightarrow$$

for i=1,. . . ,N :

$$\begin{cases} \mathbf{X}_i^q - \tilde{\mathbf{X}}_i^{q+1} = \mathbf{0} \\ \boldsymbol{\Delta}\mathbf{U}_i^q - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^{q+1} = \mathbf{0} \end{cases} \tag{2.82b}$$

$$\Longrightarrow$$

for i=1,. . . ,N :

$$\begin{cases} \tilde{\mathbf{X}}_i^{q+1} = \mathbf{X}_i^q \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^{q+1} = \boldsymbol{\Delta}\mathbf{U}_i^q \end{cases} \tag{2.82c}$$

Similarly, taking the derivative of Lagrangian (2.79) with respect to the pseudo-variables and iteratively solving the resulting equations yield the update formulae for the prices as:

$$\begin{cases} \nabla_{\tilde{\mathbf{X}}} L = \sum_{i=1}^{N} \nabla_{\tilde{\mathbf{X}}_i} L = \mathbf{0} \\ \nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}} L = \sum_{i=1}^{N} \nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_i} L = \mathbf{0} \end{cases} \tag{2.83a}$$

$$\Longrightarrow$$

for i=1,. . . ,N :

$$\begin{cases} -\varepsilon(\mathbf{X}_i^q - \tilde{\mathbf{X}}_i^{q+1})\bar{\mathbb{Q}}_i - \mathbf{\Gamma}_i^{q+1} + \sum\limits_{\substack{j=1\\j\neq i}}^{N} \hat{\mathbb{A}}_{ji}^T \mathbf{\Lambda}_j^q + \sum\limits_{\substack{j=1\\j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{A}}_{ji})^T \mathbf{\Omega}_{j,min}^q \\ \qquad\qquad - \sum\limits_{\substack{j=1\\j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{A}}_{ji})^T \mathbf{\Omega}_{j,max}^q = \mathbf{0} \\ -\varepsilon(\mathbf{\Delta U}_i^q - \mathbf{\Delta}\tilde{\mathbf{U}}_i^{q+1})\bar{\mathbb{R}}_i - \mathbf{\Pi}_i^{q+1} + \sum\limits_{\substack{j=1\\j\neq i}}^{N} \hat{\mathbb{B}}_{ji}^T \mathbf{\Lambda}_j^q + \sum\limits_{\substack{j=1\\j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{B}}_{ji})^T \mathbf{\Omega}_{j,min}^q \\ \qquad\qquad - \sum\limits_{\substack{j=1\\j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{B}}_{ji})^T \mathbf{\Omega}_{j,max}^q = \mathbf{0} \end{cases} \tag{2.83b}$$

$\Longrightarrow$

for i=1,. . . ,N :

$$\begin{cases} \mathbf{\Gamma}_i^{q+1} = \\ -\varepsilon(\mathbf{X}_i^q - \tilde{\mathbf{X}}_i^{q+1})\bar{\mathbb{Q}}_i + \sum\limits_{\substack{j=1\\j\neq i}}^{N} \hat{\mathbb{A}}_{ji}^T \mathbf{\Lambda}_j^q + \sum\limits_{\substack{j=1\\j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{A}}_{ji})^T \mathbf{\Omega}_{j,min}^q - \sum\limits_{\substack{j=1\\j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{A}}_{ji})^T \mathbf{\Omega}_{j,max}^q \\ \mathbf{\Pi}_i^{q+1} = \\ -\varepsilon(\mathbf{\Delta U}_i^q - \mathbf{\Delta}\tilde{\mathbf{U}}_i^{q+1})\bar{\mathbb{R}}_i + \sum\limits_{\substack{j=1\\j\neq i}}^{N} \hat{\mathbb{B}}_{ji}^T \mathbf{\Lambda}_j^q + \sum\limits_{\substack{j=1\\j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{B}}_{ji})^T \mathbf{\Omega}_{j,min}^q - \sum\limits_{\substack{j=1\\j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{B}}_{ji})^T \mathbf{\Omega}_{j,max}^q \end{cases} \tag{2.83c}$$

The sets of update equations in (2.82c) and (2.83c) along with the following iterative form of the interaction equations (2.76) ,

for i=1,. . . ,N :

$$\mathbf{v}_i^{q+1}(k+l|k) = \sum_{\substack{j=1\\j\neq i}}^{N} \left[\beta \mathbf{A}_{ij} \tilde{\mathbf{x}}_j^{q+1}(k+l|k) + \mathbf{B}_{ij} \sum_{a=0}^{l} \mathbf{\Delta}\tilde{\mathbf{u}}_j^{q+1}(k+a|k) \right] \tag{2.84}$$

compose the coordinator for the CDMPC in the MPMC approach.

Remark 2.5.1 *If gradient-based method is used to solve* $\nabla_{\tilde{\mathbf{X}}} L = \mathbf{0}$, $\nabla_{\mathbf{\Delta}\tilde{\mathbf{U}}} L = \mathbf{0}$,

$\nabla_{\mathbf{\Gamma}} L = \mathbf{0}$ *and* $\nabla_{\mathbf{\Pi}} L = \mathbf{0}$*, the coordinator will become:*

$$\mathbf{\Gamma}_i^{q+1} = \mathbf{\Gamma}_i^q + \epsilon_1[\mathbf{X}_i^q - \tilde{\mathbf{X}}_i^q] \tag{2.85}$$

$$\mathbf{\Pi}_i^{q+1} = \mathbf{\Pi}_i^q + \epsilon_2[\mathbf{\Delta U}_i^q - \mathbf{\Delta}\tilde{\mathbf{U}}_i^q] \tag{2.86}$$

$$\begin{aligned}\tilde{\mathbf{X}}_i^{q+1} = \tilde{\mathbf{X}}_i^q - \epsilon_3[&-\varepsilon(\mathbf{X}_i^q - \tilde{\mathbf{X}}_i^q)\bar{\mathbb{Q}}_i - \mathbf{\Gamma}_i^q + \sum_{\substack{j=1\\ j\neq i}}^{N} \hat{\mathbb{A}}_{ji}^T \mathbf{\Lambda}_j^q \\ &+ \sum_{\substack{j=1\\ j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{A}}_{ji})^T \mathbf{\Omega}_{j,min}^q - \sum_{\substack{j=1\\ j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{A}}_{ji})^T \mathbf{\Omega}_{j,max}^q]\end{aligned} \tag{2.87}$$

$$\begin{aligned}\mathbf{\Delta}\tilde{\mathbf{U}}_i^{q+1} = \mathbf{\Delta}\tilde{\mathbf{U}}_i^q - \epsilon_4[&-\varepsilon(\mathbf{\Delta U}_i^q - \mathbf{\Delta}\tilde{\mathbf{U}}_i^q)\bar{\mathbb{R}}_i - \mathbf{\Pi}_i^q + \sum_{\substack{j=1\\ j\neq i}}^{N} \hat{\mathbb{B}}_{ji}^T \mathbf{\Lambda}_j^q \\ &+ \sum_{\substack{j=1\\ j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{B}}_{ji})^T \mathbf{\Omega}_{j,min}^q - \sum_{\substack{j=1\\ j\neq i}}^{N} (\mathbb{C}_{jj}\hat{\mathbb{A}}_{jj}^{-1}\hat{\mathbb{B}}_{ji})^T \mathbf{\Omega}_{j,max}^q]\end{aligned} \tag{2.88}$$

and (2.84)

where $i = 1, \ldots, N$*; and* ϵ_1*,* ϵ_2*,* ϵ_3 *and* ϵ_4 *are tuning parameters.*

Similar to the previous two coordination methods, the MPMC approach has a hierarchical structure, in which at every sampling instant the proposed CDMPC (2.77) exchange information with the coordinator, until convergence. The convergence criterion used for the coordinator is $\left|\left|\begin{bmatrix}\mathbf{X} - \tilde{\mathbf{X}} \\ \mathbf{\Delta U} - \mathbf{\Delta}\tilde{\mathbf{U}}\end{bmatrix}\right|\right| \leq \mathfrak{e}$, where $\mathfrak{e}$ is the chosen error tolerance. When the coordinator converges, the following conditions are met:

- The values of the price vectors $\mathbf{\Gamma}_i$ and $\mathbf{\Pi}_i$[23] equal the value of corresponding Lagrange multiplier associated with the transformed interaction equality constraints.
- The transformed interaction equations are satisfied.
- The interaction equations are satisfied
- The vector $\mathbf{v}_i$ gives exact values of local interaction effects.

[23]These prices give estimations for the relevant Lagrange multipliers associated with the local transformed interaction equality constraints (2.75d) and (2.75e).

Once the coordinator converges, the first sets of the calculated optimal control variables $\mathbf{u}_i(k|k)$ are implemented on the local processes.

Implementation of the MPMC-DMPC network, is systematically performed according to Algorithm 3, and the corresponding information flow pattern can be found in Appendix A.

Algorithm 3 : Implementation of MPMC-DMPC network

---

1. Coordinator: Iteration counter q is set to 1.
2. Coordinator: Coordinating variables $\mathbf{\Gamma}_i$, $\mathbf{\Pi}_i$, $\tilde{\mathbf{X}}_\mathbf{i}$ and $\mathbf{\Delta}\tilde{\mathbf{U}}_i$ are arbitrarily initialized.
3. Coordinator: Coordinating variables $\mathbf{v}_i$ are calculated using (2.84).
4. Coordinator: Coordination variables are sent to the local controllers.
5. Local Controllers: Local optimization problems (2.77) are solved.
6. Local Controllers: Local optimal solutions $\mathbf{X}_i$ and $\mathbf{\Delta}\mathbf{U}_i$ and Lagrange multipliers $\mathbf{\Lambda}_i$ and/or $\mathbf{\Omega}_i$ are sent to the coordinator.
7. Coordinator: If $\left|\left|\begin{bmatrix} \mathbf{X}-\tilde{\mathbf{X}} \\ \mathbf{\Delta U}-\mathbf{\Delta}\tilde{\mathbf{U}} \end{bmatrix}\right|\right| \leq \mathfrak{e}$, algorithm stops. Otherwise, next step is taken.
8. Coordinator: Pseudo-variables and prices are updated using (2.82c) and (2.83c).
9. Coordinator: Iteration counter is increased by 1.
10. Steps 3-7 are repeated.

2.5.2.1 Coordinator Convergence Study

Considering the analogies between the solution strategies used in designing the coordinator by either fixed-point iteration or gradient-based methods in both the IPC and MPMC approaches, the same concepts and procedures employed in section 2.4.2.2, can be used to study convergence of the coordinator in the MPMC method. As the *general* results are similar, they have not been repeated in this section.

2.6 Case Study

In this section performance of the three discussed CDMPC schemes is tested on a process consisting of two interconnected CSTRs (Sun and El-Farra (2008)). The process flow diagram and details on the first principle model, can be found in Chapter 4, Section 4.3.

The plant includes four states, which are the concentration and temperature of each reactor and four manipulated variables that are the heat input rates and inlet concentrations. This process has three steady-states among which two are locally asymptotically stable and one is unstable at $(T_1^s, C_{A1}^s, T_2^s, C_{A2}^s) = (457.9K, 1.77\frac{kmol}{m^3}, 415.5K, 1.75\frac{kmol}{m^3})$. The control goal is to stabilize the plant around the unstable steady-state point. It is assumed that all states are measured and no disturbance affects the process. The following discrete-time linear state-space description of the process has been obtained by linearizing the nonlinear model (4.42) of Section 4.3, and discretizing the result using a sampling time of 0.005 hr:

$$\begin{aligned}\mathbf{x}(k+1) = &\left[\begin{array}{cc:cc} 1.1357 & 6.1467 & 0.1852 & 0.6696 \\ -0.0014 & 0.7954 & -0.0002 & 0.1466 \\ \hdashline 0.0704 & 0.2556 & 0.9917 & 1.5870 \\ -0.0001 & 0.0557 & -0.0005 & 0.8875 \end{array}\right] \mathbf{x}(k) + \\ &\left[\begin{array}{cc:cc} 0.0000 & 0.0778 & 0.0000 & 0.0113 \\ -0.0000 & 0.0223 & -0.0000 & 0.0039 \\ \hdashline 0.0000 & 0.0022 & 0.0000 & 0.0404 \\ -0.0000 & 0.0007 & -0.0000 & 0.0471 \end{array}\right] \mathbf{u}(k)\end{aligned} \tag{2.89}$$

where $\mathbf{x}(k) = \left[\mathbf{x}_1(k)^T,\ \mathbf{x}_2(k)^T\right]^T \triangleq \left[[\bar{T}_1(k)\ \bar{C}_{A1}(k)],\ [\bar{T}_2(k)\ \bar{C}_{A2}(k)]\right]^T$. Similarly, $\mathbf{u}(k) = \left[\mathbf{u}_1(k)^T,\ \mathbf{u}_2(k)^T\right]^T \triangleq \left[[\bar{Q}_1(k),\ \bar{C}_{A0}(k)],\ [\bar{Q}_2(k),\ \bar{C}_{A03}(k)]\right]^T$. Both $\mathbf{x}$ and $\mathbf{u}$ are in deviation variable form. The plant is assumed to consist of two local units, each containing one of the CSTRs. Also, it is assumed that two decentralized MPC are controlling the plant. The parameters used in the simulations are listed in Table 2.1.

	MPC 1	MPC 2
Initial Conditions	$\mathbf{x}_1(0) = \begin{bmatrix} 5 & -1.5 \end{bmatrix}^T$	$\mathbf{x}_2(0) = \begin{bmatrix} -5 & 1.7 \end{bmatrix}^T$
Weighting Matrices	$\mathbf{Q}_{11} = 5\mathbf{I}$ $\mathbf{R}_{11} = 15\mathbf{I}$	$\mathbf{Q}_{22} = 5\mathbf{I}$ $\mathbf{R}_{22} = 15\mathbf{I}$
Prediction Horizon	10	10
Control Horizon	5	5

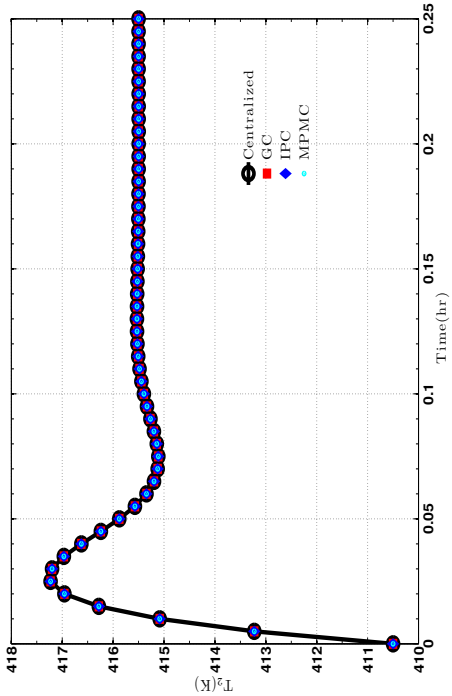
Table 2.1: Parameters used in the CDMPC

2.6.1 Simulation Results

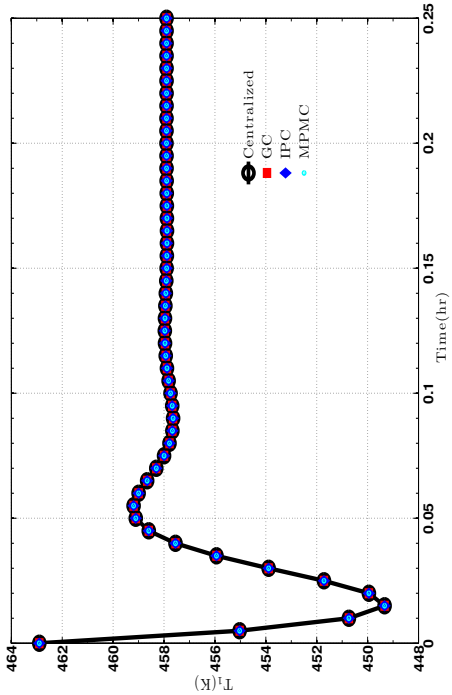
Simulations are performed using the proposed CDMPC schemes, centralized and decentralized MPC and the results are compared[24]. In Figures 2.1 and 2.2, simulation results for output and manipulated variable trajectories obtained by the three coordinated methods are plotted along with the centralized MPC solution as the performance benchmark. The results indicate that the CDMPC, in all the three coordination methods, produce the centralized MPC solution.

For this example, the GC method encountered convergence problems when limits on the process outputs and manipulated variables were used. Therefore, the results are obtained for unconstrained MPC, as the main purpose of this case study, is showing that numerical simulations agree with what is suggested by theory; however, the consequence of not using inequality constraints in the controllers can be seen in Figure 2.3(d), where the controllers calculated negative values for inlet concentration. In the next chapter, a modified coordinator for the GC method will be proposed, which does not have convergence issues in the presence of local inequalities. In Appendix D, simulation results are provided for the constrained version of the current example.

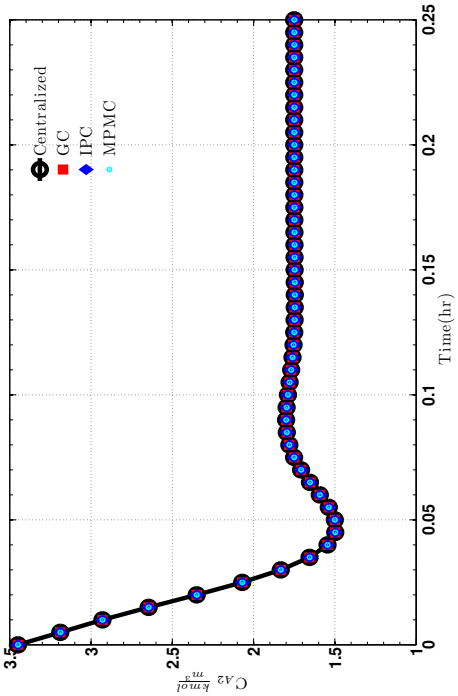
[24]Throughout this thesis, simulations for the IPC and MPMC methods are carried out using coordinators obtained by fixed-point iteration technique.



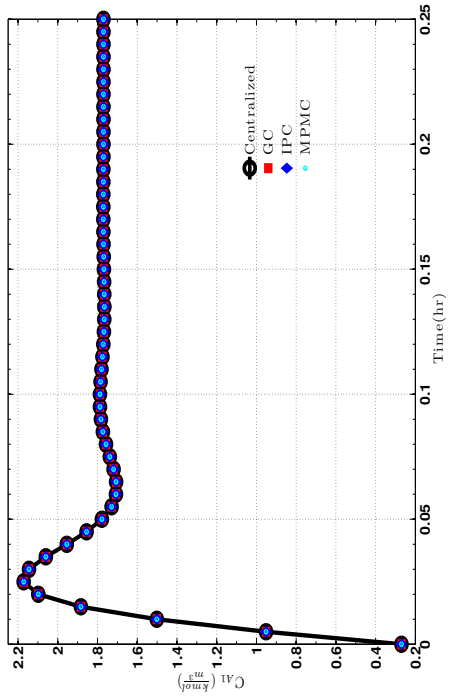
(b) Temperature of unit 2



(a) Temperature of unit 1

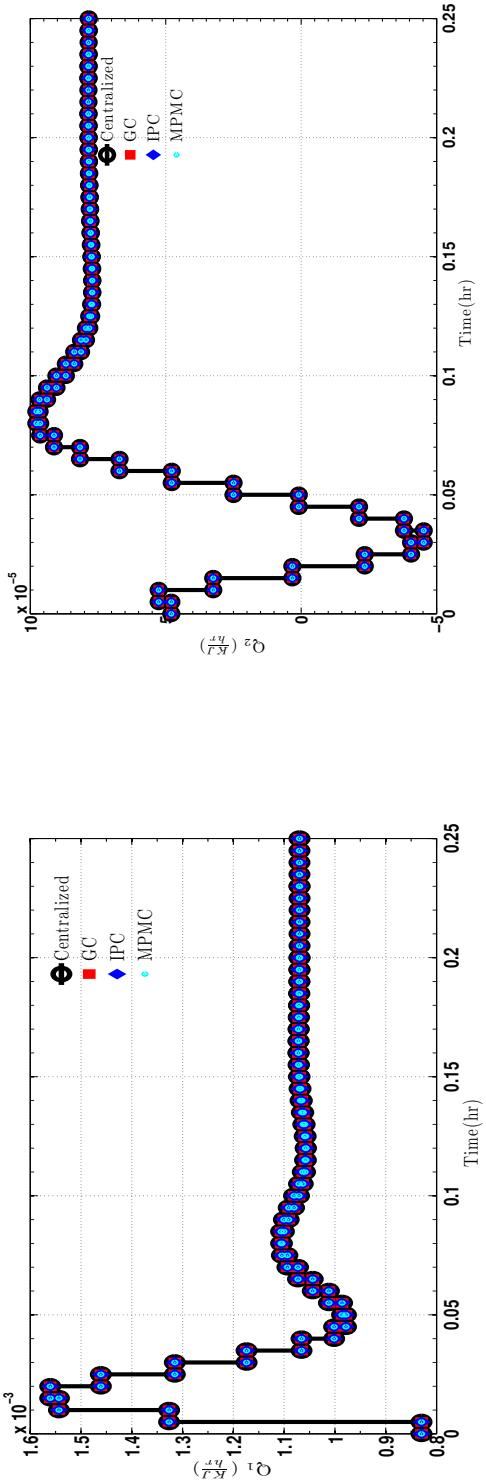


(d) Concentration of unit 2



(c) Concentration of unit 1

Figure 2.1: Output trajectories resulting from CDMPC and centralized MPC



(a) Heat input rate for unit 1

(b) Heat input rate for unit 2

(c) Inlet reactant concentration for unit 1

(d) Inlet reactant concentration for unit 2

Figure 2.2: Control input trajectories resulting from CDMPC and centralized MPC

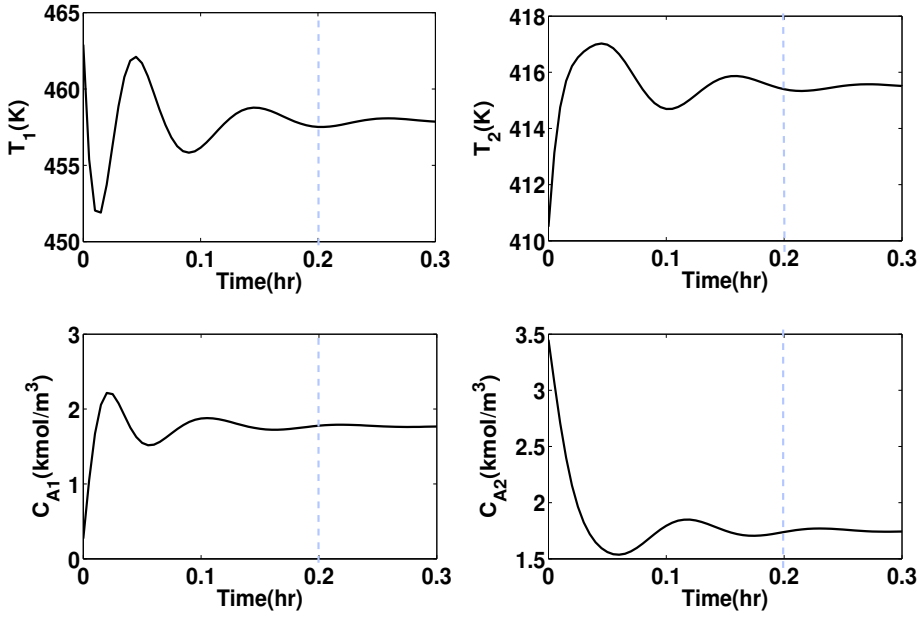


Figure 2.3: Process output trajectories resulting from decentralized MPC

In Figure 2.3, process output trajectories resulting from the decentralized controllers have been plotted. The simulation results show that though the two controllers have stabilized the process around the unstable operating point, they have produced oscillatory behaviour in the process outputs. Also, comparing with Figure 2.1, the decentralized controllers need more time to settle down the outputs. All of these performance degradations are due to ignoring the interactions in designing the decentralized MPC.

The number of communication cycles required for the coordinator to converge during sampling intervals is plotted versus the simulation time in Figure 2.4.

It should be mentioned that in all the coordinated distributed simulations, the stopping criteria of $\|\mathfrak{E}\| \leq 10^{-6}$ is used and no local inequalities are present. Also, in the GC approach, Newton's method with constant step-size of $\epsilon = 1$ is used. Clearly changing each of these factors will affect the coordinator's convergence rate and consequently the number of required communication cycles. But, the general results to be concluded from Figure 2.4 are that under the same simulation conditions, the coordinators in the IPC and the MPMC methods show very similar convergence

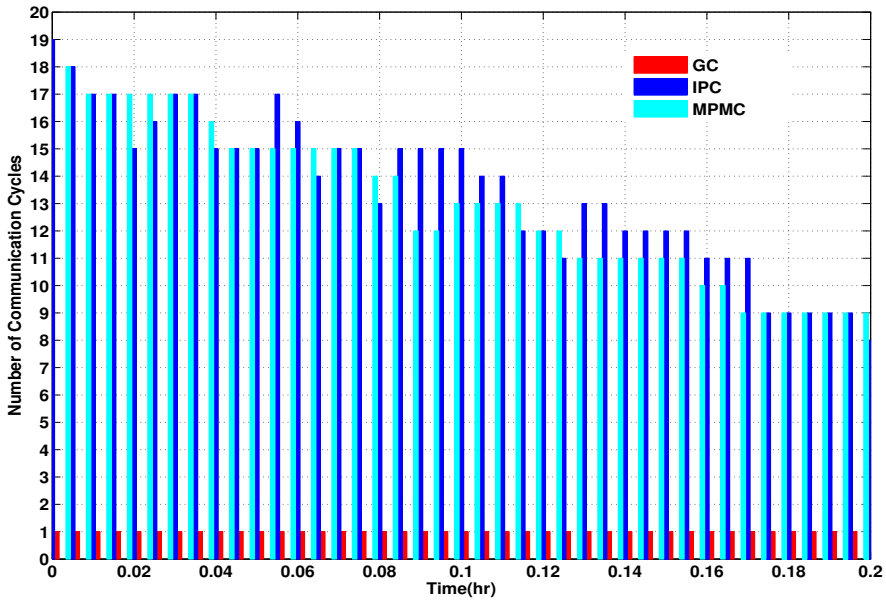


Figure 2.4: Number of iterations at every sampling instant vs. time

behaviour. Also, if the Hessian matrix obtained from the local sensitivity information is negative definite, the Newton-based coordinator in the GC method, converges significantly faster than the coordinators in the other two approaches.

2.7 Summary

In this chapter, three coordination methods, namely the GC, IPC and MPMC, have been applied to decentralized linear constrained MPC. Accordingly, three CDMPC schemes have successfully been developed for plant-wide MPC problem.

The common concept used in the coordination methods studied in this chapter, is to convert the existing network of decentralized MPC into a network of CDMPC by doing minor modifications in their optimization problems. Some form of interaction equation relaxation is the basis for performing the required modifications. The relaxation results in the addition of the so-called coordinating terms to the local objective functions. The coordinating terms contain coordinating variables that connect local controllers to the coordinator. Depending on how interaction equations are manipulated and what form of relaxation is used, different CDMPC formulations

are obtained.

After modifying the decentralized MPCs, the coordinator is designed by the numerical algorithm used to solve the optimization problem resulting from the aggregate of CDMPC. In the GC method, the coordinator update equations are determined by the numerical method used to solve a relevant dual optimization problem. In the IPC and MPMC methods, the numerical technique chosen to solve portions of the optimality conditions of the aggregate CDMPC dictates the coordinator's formulation. Convergence of coordination algorithms for the proposed CDMPC networks are studied. It is shown that at each sampling time, over the communication cycles, the proposed coordinators converge and optimal plant-wide solution is achieved.

The presented CDMPC schemes of this chapter, can be compared in terms the following characteristics:

Ease of implementation: The three proposed CDMPC schemes are equally easy to implement due to the simple communication strategy used in coordinated distributed control structures.

Required tuning: The coordinator in the GC method contains a step-size as its tuning parameter. In the IPC and MPMC methods, the coordinator designed using the fixed-point iteration technique does not contain tuning parameters, and the coordinator designed based on using a gradient-based method contains constant tuning parameters.

Required computation and communication: In the GC-DMPC scheme, if the coordinator needs to calculate optimal step-size, then within each communication cycle, a univariate optimization should be performed. In this case, computations and the number of iterations of the CDMPC network may considerably increase. The required computations and communication cycles for the IPC-DMPC and MPMC-DMPC approaches are very similar and usually less than what is needed by the GC-DMPC network.

Handling inequality constraints: The proposed coordinator in this chapter for the GC-DMPC cannot effectively handle active local inequality constraints. Thus, performance of the coordination algorithm has to be improved. In Chapter 3,

the proposed GC-DMPC scheme of Chapter 2 is modified to enhance convergence performance of the coordinator and simulation results are provided in Appendix D. Simulation studies indicate that the coordinator in both IPC-DMPC and MPMC-DMPC schemes are able to effectively handle local inequality constraints

Simplicity of the coordinator: If the coordinator in the GC method can converge using a simple first-order gradient method without the need to calculate the optimal step-size, then the coordinator's equations will be very simple; otherwise, if optimal step-size has to be determined and/or other gradient methods are used, the resulting coordinator can become relatively complex. The coordinator in the IPC and MPMC methods, involves very simple computations and equations.

Explicit use of interaction models in the local controllers: In the GC-DMPC and IPC-DMPC schemes, interaction models partially appear in the objective functions of the CDMPCs due to relaxation of the interaction equations. In the MPMC method because of using the pseudo-variables and relaxation of the transformed interaction equations, interaction models do not appear in the CDMPC formulations. This characteristic plays an important role in applying the MPMC method to the plant-wide nonlinear MPC problem of dynamically interconnected nonlinear processes, discussed in Chapter 4.

It can be concluded that only if an arbitrary constant step-size can be used and local inequality constraints are not present, the GC-DMPC scheme that uses a Newton-based coordinator, will have the best coordination performance; otherwise, the IPC and MPMC methods are preferred as they involve simpler computations, need reasonable number of communication cycles and can efficiently handle active local inequality constraints. Also, between IPC and MPMC methods, the latter has priority over the former because distributed controllers in the MPMC method do not contain the interaction models.

Chapter 3

Chance-Constrained Coordinated Distributed MPC

In this chapter, the Goal Coordination, Interaction Prediction Coordination and Modified Pseudo-Model Coordination methods are applied to the plant-wide chance-constrained MPC problem. The presence of uncertain disturbances in the local process models results in uncertain predicted outputs. Therefore, the predicted outputs are required to be held within a specified range with a given probability. To improve the convergence behaviour of the coordinator in the presence of local inequalities resulting from the chance-constraints, in the GC method, the CDMPC formulation is modified and a new coordinator is designed by using a Separated Augmented Lagrangian method. Detailed formulations for the three probabilistic CDMPC networks are presented. The effectiveness of the proposed coordinated distributed schemes are tested through a simulation example.

Uncertainty is an intrinsic characteristic of any process; however, traditional process design, optimization and control problems are deterministic, as uncertainty effects are not explicitly included in mathematical description of the processes. Therefore, commonly, in conventional MPC design, despite the uncertainties in the real process, a deterministic process model is used. It is clear that when discrepancies between the real uncertain process and its model are large, the nominal MPC could result in

unacceptable performance. Synthesizing an MPC such that the effects of uncertainties are explicitly taken into account, may efficiently deal with this type of performance degradation while assuring feasibility at a high level of control accuracy. Therefore, for practical purposes, robust MPC strategies should be considered. On the other hand, in the context of plant-wide MPC, performance of decentralized MPC network can greatly be enhanced by using coordinated distributed schemes. A promising approach to practical plant-wide MPC problem would emerge if the two concepts of robust MPC and CDMPC are integrated to develop robust CDMPC networks.

Research on robust MPC can be categorized in two general groups. The first category is known as Robust MPC where *robust optimization* techniques are used to synthesize a robust model predictive controller. Depending on the modelling and identification procedures, polytopic (multi-model) or structured feedback uncertainty paradigms can be used to model the uncertain system (Kothare *et al.* (1996)). In such MPC problems, the bounds on uncertainties in the process model are known. Thus, a worst-case optimization is performed to find an optimal solution, which is feasible for all the uncertainties within the specified range. One characteristic of Robust MPC is that it is a conservative control strategy because of its worst-case oriented approach. Also, most Robust MPC methods are computationally expensive and consequently not suitable for online implementation (Kothare *et al.* (1996)). Examples of research on Robust MPC can be found in Kothare *et al.* (1996), Bemporad and Morari (1999), Cuzzola *et al.* (2002), Wan and Kothare (2003), Langson *et al.* (2004) and Wang and Rawlings (2004).

In the second group of robust MPC methods, *stochastic optimization* techniques are employed to solve the resulting optimization problem under uncertainty. In stochastic optimization, the probability distributions governing the uncertain data and parameters are known or can be estimated. The goal is to find some policy that is feasible for all (or almost all) the possible uncertain data and optimize the expectations of the objective function (COSP (2012)). A stochastic optimization problem in its initial form, involves a performance index and constraints, which are functions of uncertain parameters. This form of the problem is not solvable because a decision should be made before knowing the realizations of the uncertain data.

Therefore, to make the problem solvable, it is revised so that a so-called *deterministic equivalent* (Kall and Wallace (1994)) is formed. The simplest way to cast an stochastic optimization problem as a deterministic problem is to replace stochastic variables by their expected values. Another way to generate such a deterministic equivalent problem is known as the *two-stage stochastic program with recourse*[1], in which first stage decisions are made, knowing only the probability distribution for the stochastic elements. To ensure constraint feasibility, a recourse (at a cost) to second stage variables, which vary for different realizations of the stochastic elements, is used. First stage decision variables are obtained by minimizing the total expected cost, which is the sum of the *known costs* for the first stage decisions plus the *expected cost* for the second stage decisions (Beasley (2013)).

The third approach to generate a deterministic equivalent is employing probabilistic constraints. This methodology is known as *Chance-Constrained* or *Probabilistic* Optimization. In this technique, the stochastic optimization problem is relaxed into an equivalent nonlinear optimization problem. The resulting *deterministic* problem can then be solved by nonlinear programming (NLP) methods (Li *et al.* (2008)). Probabilistic constraints can either be defined as *single/individual* or *joint*. Single chance constraints are satisfied independently, while joint probabilistic constraints are required to be fulfilled simultaneously within the given probability. Single chance constraints are simple to handle because they usually can be written in terms of the quantile function, that is the inverse of the Cumulative Distribution Function (CDF). This implies that if the stochastic variables have strictly increasing and continuous CDF, the corresponding single constraints can be converted into linear inequality constraints. Joint chance constraints correspond to multivariate CDF where no quantile function is available. Therefore, to solve the optimization problem by an NLP method, the multivariate (joint) CDF and its gradient should be calculated. If the dimension of the uncertain parameters is larger than three, this would be a computationally expensive task as it involves numerical computation of multiple integrals (Li *et al.* (2008)).

[1]Two-stage stochastic program with recourse can be extended to Multi-stage stochastic program with recourse (Kall and Wallace (1994)).

Although a relatively rich literature is available for stochastic optimization methods, there is a small literature for stochastic MPC. Probabilistic optimization can be a powerful method for synthesizing robust MPC. One advantage of chance-constrained MPC is that the probabilistic constraints provide a practical way to account for the effects of uncertainties on process outputs. Optimal control inputs are calculated such that optimal control performance is achieved, while process outputs are held within the desired limits with a given probability. Consequently, in comparison with the two-stage stochastic MPC, chance-constrained MPC appears to be a more practical approach.

One of the first research studies in the context of probabilistic MPC, was done by Schwarm and Nikolaou (1999) where uncertainties were included in the parameters of a step-response model. They used single probabilistic constraints on the predicted outputs. The structure of each of the output constraints was such that the product of a random vector containing the uncertain model parameter times a vector involving linear functions of manipulated variable changes (decision variables) was less than or equal to a deterministic scalar. Also they assumed that the uncertain parameters were normally distributed. They transformed the normally distributed parameters on the left hand-side of the output constraints into standard normally distributed variables and then converted the probabilistic constraints into equivalent linear inequality constraints by using quantile functions. Through a high-purity distillation column simulation study, they showed their proposed chance-constrained MPC could enhance robustness properties of standard MPC.

Li *et al.* (2000) employed both model and disturbance uncertainties with multivariate normal distributions in a Multi-Input Multi-Output (MIMO) step-response model. In the objective function of the MPC problem, they only considered minimization of the control moves in the future horizon. They re-wrote the process model such that the predicted outputs were represented by a deterministic vector plus the product of a deterministic matrix times a random vector. The random vector contained the uncertain step-response coefficients and disturbances. Based on the new form of the model, they wrote the inequality constraints on the predicted outputs such that a standard normal random vector was less than or equal to a

vector whose elements were functions of the manipulated variables. They used joint chance constraints on these inequalities and applied a sampling method to approximate the values of the multivariate CDFs and their gradients to be used in the Sequential Quadratic Programming (SQP) method as the NLP solver. They applied the proposed controller to operate a distillation column and showed that satisfactory control performance was achieved.

Li *et al.* (2002) also presented their proposed chance-constrained MPC in Li *et al.* (2000), for a Single-Input Single-Output (SISO) linear model with an uncertain disturbance. They demonstrated the effectiveness of the control strategy with an example of a tank process.

Blackmore and Ono (2009) proposed a new approach for joint chance-constrained finite horizon MPC that did not require the evaluation of multivariate probability densities. The objective function used for the MPC formulation was assumed to be a function of the manipulated variables and expected values of the predicted states. They used state-space representation of the process with uncertain normally distributed disturbances and employed joint probabilistic constraints on the predicted states. Rather than using a sampling approach, they employed *convex risk allocation* in which they approximated the joint chance-constraints by a set of individual constraints. They showed that a feasible solution to such an approximation was a feasible solution to the original problem. Though this indicated that the solution to the approximate problem with the single probabilistic constraints was conservative, they empirically showed the introduced conservatism was small.

Chance-constrained CDMPC is an untouched topic and no solid study has yet been reported in the literature. This work intends to initiate the study of how uncertainties can affect developing CDMPC networks, and particularly coordinator synthesis. As this is the very first attempt in this context, a new yet tractable, chance-constrained MPC formulation that uses state-space models with uncertain disturbance inputs, and single probabilistic constraints on the uncertain predicted outputs, has been employed. In addition to a simpler MPC formulation, using single chance-constraints provides the benefit of converting the nonlinear probabilistic constraints into equivalent linear inequality constraints and thus, assuring convex

CDMPC. It is assumed that a network of probabilistic decentralized MPC currently controls the plant. The GC, IPC and MPMC methods are applied to the existing decentralized MPC to develop chance-constrained CDMPC network. Contributions of this chapter are:

- Using the GC method, the chance-constrained decentralized MPCs are converted into chance-constrained CDMPCs, and the coordinator is designed accordingly. To deal with the coordinator's convergence issues in the presence of local inequality constraints, an improved first-order gradient-based optimization approach, based on augmented Lagrangian method, is suggested to solve the resulting dual optimization problem. The chosen method for solving the dual optimization problem results in a slight modification in the CDMPC formulations and a new coordinator. The improved coordinator can be used for both deterministic and probabilistic CDMPC problems.

- The Interaction Prediction Coordination approach is employed to obtain chance-constrained CDMPC by performing modifications on the decentralized MPC formulations. Detailed formulation for the corresponding coordinator is presented.

- The Modified Pseudo-Model Coordination method is used to alter formulations of the probabilistic decentralized MPC to probabilistic CDMPC and construct the coordinator.

This chapter also aims to show how coordination methods can be adapted to various plant-wide MPC problems other than the standard plant-wide linear constrained MPC.

3.1 Background

In this section, the mathematical background required for constructing probabilistic CDMPC networks and the notation used throughout this chapter, is presented.

3.1.1 Model of the Uncertain Plant

The following linear time-invariant state-space model is used to mathematically represent the local process models:

$$\mathbf{x}_i(k+1) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{u}_i(k) + \mathbf{D}_{ii}\mathbf{d}_i(k) + \sum_{\substack{j=1 \\ j\neq i}}^{N} [\mathbf{A}_{ij}\mathbf{x}_j(k) + \mathbf{B}_{ij}\mathbf{u}_j(k) + \mathbf{D}_{ij}\mathbf{d}_j(k)] \tag{3.1a}$$

$$\mathbf{y}_i(k) = \mathbf{C}_{ii}\mathbf{x}_i(k) \tag{3.1b}$$

where $i = 1, \ldots, N$. The vectors $\mathbf{x}_i(k) \in \mathbb{R}^{n_{xi}}$, $\mathbf{u}_i(k) \in \mathbb{R}^{n_{ui}}$, $\mathbf{d}_i(k) \in \mathbb{R}^{n_{di}}$ and $\mathbf{y}_i(k) \in \mathbb{R}^{n_{yi}}$ contain the states, inputs, disturbances and outputs of the local unit i at time k in deviation form, respectively. It is assumed that the disturbances are uncertain variables that follow multivariate normal distributions.

3.1.2 Plant-Wide Probabilistic MPC

As was discussed in the previous chapter, in order to construct a coordinated distributed network, a performance benchmark is needed. Inspired by the limited number of references available for probabilistic MPC, in this chapter, a probabilistic MPC formulation is introduced using the following specifications:

- A state-space prediction model with an uncertain disturbance input vector[2], is available.
- Manipulated variable changes and deviations of the expectation of the predicted output values from set-points are quadratically penalized in the objective function.
- Single chance-constraints are used for process limits on the uncertain predicted outputs.

[2]In order to be consistent with the CDMPC schemes presented in chapter 2, the state-space model is used in the probabilistic MPC. On the other hand, almost all the available references for probabilistic MPC use Input-Output models with uncertain model parameters and/or disturbances. Only Blackmore and Ono (2009) used a state-space model in which random disturbances and model uncertainties were assumed to be modelled by a random input vector.

Based on the above assumptions, the following centralized chance-constrained MPC formulation is used to represent the maximum achievable plant performance:

$$\min_{\bar{\mathbf{X}},\boldsymbol{\Delta}\mathbf{U}} \quad \sum_{i=1}^{N} \left[\left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i \right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i \right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i \right] \tag{3.2a}$$

subject to

$$\begin{aligned} \bar{\mathbf{x}}_i(k+l+1|k) = \mathbf{A}_{ii}\bar{\mathbf{x}}_i(k+l|k) + \mathbf{B}_{ii} \left[\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1) \right] + \mathbf{D}_{ii}\bar{\mathbf{d}}_i(k+l|k) \\ + \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\alpha \mathbf{A}_{ij}\bar{\mathbf{x}}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \mathbf{D}_{ij}\bar{\mathbf{d}}_j(k+l|k) \right] + \mathbf{v}_i(k+l|k) \end{aligned} \tag{3.2b}$$

$$\mathbf{v}_i(k+l|k) = \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\beta \mathbf{A}_{ij}\bar{\mathbf{x}}_j(k+l|k) + \mathbf{B}_{ij} \sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_j(k+a|k) \right] \tag{3.2c}$$

$$\begin{aligned} &\mathtt{Pr}\{\mathbf{C}_{ii,g}\mathbf{x}_i(k+l+1|k) \geq y_{i,g}^{min}(k+l+1)\} \geq \zeta_{i,g,l}^{\mathtt{L}} \\ &\mathtt{Pr}\{\mathbf{C}_{ii,g}\mathbf{x}_i(k+l+1|k) \leq y_{i,g}^{max}(k+l+1)\} \geq \zeta_{i,g,l}^{\mathtt{U}} \end{aligned} \tag{3.2d}$$

$$\begin{aligned} &\mathbf{u}_i^{min}(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}_i^{max}(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \leq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \leq \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b) \\ &i = 1, \ldots, N \end{aligned} \tag{3.2e}$$

where $g = 1, \ldots, n_{yi}$. The vector $\bar{\mathbf{d}}_i(k+l|k)$ contains expected values of the future uncertain disturbances. The vector $\bar{\mathbf{x}}_i(k+l+1|k)$ includes the predicted states obtained using $\bar{\mathbf{d}}_i(k+l|k)$. Note that, for the constraints on the predicted outputs, local dynamic equations (3.2b) with the uncertain disturbance vector $\mathbf{d}_i(k+l|k)$ rather than $\bar{\mathbf{d}}_i(k+l|k)$, should be considered to account for the effects of uncertain disturbances propagated to the predicted states and predicted outputs. Since the model is linear, the predicted states and outputs will have the same distribution as the uncertain disturbances. The constraints on the uncertain predicted outputs are written in the single probabilistic form (3.2d), where $\zeta_{i,g}^{\mathtt{L}}$ and $\zeta_{i,g}^{\mathtt{U}}$ are the pre-specified probability levels for each element of the predicted output vector. $\mathbf{C}_{ii,g}$ corresponds to the g_{th} row of the $\mathbf{C}_{ii}$ matrix.

The nonlinear chance-constraints (3.2d) can be converted into equivalent linear inequality constraints using the quantile functions. Therefore, the nonlinear chance-

constrained problem (3.2) is transformed into an equivalent Quadratic Programming (QP) problem with linear constraints. To do so, first the centralized prediction model is written over the entire prediction and control horizons in the following compact form:

$$\hat{\mathbb{A}}\mathbf{X} + \hat{\mathbb{B}}\boldsymbol{\Delta}\mathbf{U} + \hat{\mathbb{D}}\mathfrak{D} = \bar{\mathbb{A}}\mathbf{x}(k) + \bar{\mathbb{B}}\mathbf{u}(k-1) \tag{3.3}$$

where the derivation details have been provided in Appendix C, section C.4. Also, the constraints on the predicted outputs of local units can be written as:

$$\begin{aligned} \mathbb{C}\mathbf{X} &\geq \mathbf{Y}^{min} \\ \mathbb{C}\mathbf{X} &\leq \mathbf{Y}^{max} \end{aligned} \tag{3.4}$$

Assuming $\hat{\mathbb{A}}$ is non-singular, combining (3.3) and (3.4) results in the following inequality constraints:

$$\begin{aligned} \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}}\mathfrak{D} &\leq -\mathbf{Y}^{min} - \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{B}}\boldsymbol{\Delta}\mathbf{U} + \mathbb{C}\hat{\mathbb{A}}^{-1}\bar{\mathbb{A}}\mathbf{x}(k) + \mathbb{C}\hat{\mathbb{A}}^{-1}\bar{\mathbb{B}}\mathbf{u}(k-1) \\ \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}}\mathfrak{D} &\geq -\mathbf{Y}^{max} - \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{B}}\boldsymbol{\Delta}\mathbf{U} + \mathbb{C}\hat{\mathbb{A}}^{-1}\bar{\mathbb{A}}\mathbf{x}(k) + \mathbb{C}\hat{\mathbb{A}}^{-1}\bar{\mathbb{B}}\mathbf{u}(k-1) \end{aligned} \tag{3.5}$$

In the left-hand side of the inequalities (3.5), constant matrices $\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}}$ are multiplied by the normally distributed random vector $\mathfrak{D}$ with the expected value vector $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Xi}$, i.e $\mathfrak{D} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Xi})$. Thus, $\boldsymbol{\xi} \triangleq \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}}\mathfrak{D}$ has also a normal multivariate distribution with the expected value vector $\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}}\boldsymbol{\mu}$ and the covariance matrix $(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})\boldsymbol{\Xi}(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})^T$. Using the following *coordinate transformation* (Li *et al.* (2002)), $\boldsymbol{\xi}$ can be standardized:

$$\boldsymbol{\xi}' = \left[(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})\boldsymbol{\Xi}(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})^T\right]^{-\frac{1}{2}} \left[\boldsymbol{\xi} - \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}}\boldsymbol{\mu}\right] \tag{3.6}$$

where $\boldsymbol{\xi}' \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Combining (3.5) and (3.6) leads to the following inequalities:

$$\begin{aligned} \boldsymbol{\xi}' &\leq \left[(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})\boldsymbol{\Xi}(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})^T\right]^{-\frac{1}{2}} \times \\ &\left[-\mathbf{Y}^{min} - \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{B}}\boldsymbol{\Delta}\mathbf{U} + \mathbb{C}\hat{\mathbb{A}}^{-1}\bar{\mathbb{A}}\mathbf{x}(k) + \mathbb{C}\hat{\mathbb{A}}^{-1}\bar{\mathbb{B}}\mathbf{u}(k-1) - \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}}\boldsymbol{\mu}\right] \\ \boldsymbol{\xi}' &\geq \left[(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})\boldsymbol{\Xi}(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})^T\right]^{-\frac{1}{2}} \times \\ &\left[-\mathbf{Y}^{max} - \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{B}}\boldsymbol{\Delta}\mathbf{U} + \mathbb{C}\hat{\mathbb{A}}^{-1}\bar{\mathbb{A}}\mathbf{x}(k) + \mathbb{C}\hat{\mathbb{A}}^{-1}\bar{\mathbb{B}}\mathbf{u}(k-1) - \mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}}\boldsymbol{\mu}\right] \end{aligned} \tag{3.7}$$

So far the predicted output constraints (3.4) have been written in the equivalent form (3.7). Since the purpose of presenting the plant-wide formulation is to provide the benchmark for the distributed control network, the inequality constraints (3.7) should be written in terms of the local process units. When coordinate transformation is used to convert the normally distributed random disturbances to standard normally distributed counterparts, the term $(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})\boldsymbol{\Xi}(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})^T$ requires attention because its distributed equivalent is not $(\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii})\boldsymbol{\Xi}_{ii}(\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii})^T + \sum_{\substack{j=1\\ j\neq i}}^{N}(\mathbb{C}_{ij}\hat{\mathbb{A}}_{ij}^{-1}\hat{\mathbb{D}}_{ij})\boldsymbol{\Xi}_{ij}(\mathbb{C}_{ij}\hat{\mathbb{A}}_{ij}^{-1}\hat{\mathbb{D}}_{ij})^T$. Thus, in (3.3), the overall model (C.4) rather than (C.6), is used to avoid this mistake, which introduces mismatch in the formulations of the distributed controllers by ignoring the effects of off-diagonal blocks in the multiplication of the matrices. More explanation is provided in Appendix C, section C.2, using a simple example. One way to correctly present this term in the distributed form, is to first calculate $\mathbb{G} = \left[(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})\boldsymbol{\Xi}(\mathbb{C}\hat{\mathbb{A}}^{-1}\hat{\mathbb{D}})^T\right]^{-\frac{1}{2}}$; then define $\mathbb{G}_i$ by picking the relevant i_{th} rows and all their corresponding columns of $\mathbb{G}$. The dimension of $\mathbb{G}_i$ would be $(n_{yi}H_p)\times(n_yH_p)$. Considering (C.6), inequalities in (3.7) for each subsystem become:

$$\begin{aligned}\boldsymbol{\xi}_i' &\leq \mathfrak{f}_i^{min}\\ \boldsymbol{\xi}_i' &\geq \mathfrak{f}_i^{max}\end{aligned} \tag{3.8}$$

where $i = 1, \ldots, N$. The vectors $\mathfrak{f}_i^{min}$ and $\mathfrak{f}_i^{max}$ which are functions of the predicted manipulated vector $\boldsymbol{\Delta}\mathbf{U}_i$, are defined as below:

$$\begin{aligned}\mathfrak{f}_i^{min} \triangleq\ & \mathbb{G}_i[-\mathbf{Y}_i^{min} - \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i + \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\mathbf{V}_i\\ &+ \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{A}}_{ii}\mathbf{x}_i(k) + \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{B}}_{ii}\mathbf{u}_i(k-1) - \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii}\boldsymbol{\mu}_i\\ &- \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ij}\boldsymbol{\mu}_j + \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{A}}_{ij}\mathbf{x}_j(k) + \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{B}}_{ij}\mathbf{u}_j(k-1)]\\ \mathfrak{f}_i^{max} \triangleq\ & \mathbb{G}_i[-\mathbf{Y}_i^{max} - \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i + \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\mathbf{V}_i\\ &+ \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{A}}_{ii}\mathbf{x}_i(k) + \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{B}}_{ii}\mathbf{u}_i(k-1) - \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii}\boldsymbol{\mu}_i\\ &- \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ij}\boldsymbol{\mu}_j + \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{A}}_{ij}\mathbf{x}_j(k) + \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{B}}_{ij}\mathbf{u}_j(k-1)]\end{aligned} \tag{3.9}$$

The relationship between the single chance-constraints (3.2d) and inequalities (3.8) is that the probability of each element in (3.8) is equivalent to the left hand-side of the constraints in (3.2d). That is, the probabilistic constraints (3.2d) can be described by the following equivalent constraints:

$$\begin{aligned} &\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k) \le f^{min}_{i,g}(k+l+1|k)\} \ge \zeta^{\mathtt{L}}_{i,g,l} \\ &\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k) \ge f^{max}_{i,g}(k+l+1|k)\} \ge \zeta^{\mathtt{U}}_{i,g,l} \end{aligned} \tag{3.10}$$

Now by using the quantile function $\mathtt{F}^{-1}(.)$, the probability constraints (3.10) can be converted into the following linear inequality constraints:

$$\begin{aligned} &f^{min}_{i,g}(k+l+1|k) \ge \mathtt{F}^{-1}(\zeta^{\mathtt{L}}_{i,g,l}) \\ &f^{max}_{i,g}(k+l+1|k) \le \mathtt{F}^{-1}(1-\zeta^{\mathtt{U}}_{i,g,l}) \end{aligned} \tag{3.11}$$

where in the vector form these inequality constraints becomes:

$$\begin{aligned} &\mathfrak{f}^{min}_i \ge \mathfrak{F}^{\mathtt{L}}_i \\ &\mathfrak{f}^{max}_i \le \mathfrak{F}^{\mathtt{U}}_i \end{aligned} \tag{3.12}$$

where

$\mathfrak{F}^{\mathtt{L}}_i \triangleq \left[\mathtt{F}^{-1}(\zeta^{\mathtt{L}}_{i,1,0}), \mathtt{F}^{-1}(\zeta^{\mathtt{L}}_{i,2,0}), \ldots, \mathtt{F}^{-1}(\zeta^{\mathtt{L}}_{i,n_{yi},0}), \ldots, \mathtt{F}^{-1}(\zeta^{\mathtt{L}}_{i,1,H_p-1}), \ldots, \mathtt{F}^{-1}(\zeta^{\mathtt{L}}_{i,n_{yi},H_p-1})\right]^T$

and

$\mathfrak{F}^{\mathtt{U}}_i \triangleq \left[\mathtt{F}^{-1}(1-\zeta^{\mathtt{U}}_{i,1,0}), \ldots, \mathtt{F}^{-1}(1-\zeta^{\mathtt{U}}_{i,n_{yi},0}), \ldots, \mathtt{F}^{-1}(1-\zeta^{\mathtt{U}}_{i,1,H_p-1}), \ldots, \mathtt{F}^{-1}(1-\zeta^{\mathtt{U}}_{i,n_{yi},H_p-1})\right]^T$.

3.1.3 Decentralized Chance-Constrained MPC

In addition to the plant-wide probabilistic MPC, the decentralized scheme provides another measure for performance comparison of chance-constrained CDMPC, since coordinated distributed controllers are obtained by modifying the individual controllers in the decentralized scheme and adding a coordinator. In the decentralized case, the effects of interactions are not accounted for in the local process models. Therefore, instead of solving one large optimization problem, N smaller decoupled optimization problems are solved. The decentralized chance-constrained MPC are

formulated as below:

$$\min_{\bar{\mathbf{X}}_i, \boldsymbol{\Delta}\mathbf{U}_i} \quad \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i \tag{3.13a}$$

subject to

$$\bar{\mathbf{x}}_i(k+l+1|k) = \mathbf{A}_{ii}\bar{\mathbf{x}}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] + \mathbf{D}_{ii}\bar{\mathbf{d}}_i(k+l|k) \tag{3.13b}$$

$$\begin{aligned} &\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k) \leq f_{i,g}^{min}(k+l+1|k)\} \geq \zeta_{i,g,l}^{\mathtt{L}} \\ &\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k) \geq f_{i,g}^{max}(k+l+1|k)\} \geq \zeta_{i,g,l}^{\mathtt{U}} \end{aligned} \tag{3.13c}$$

$$\begin{aligned} &\mathbf{u}_i^{min}(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}_i^{max}(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \leq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \leq \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b) \end{aligned} \tag{3.13d}$$

where $\mathfrak{f}_i^{min}$ and $\mathfrak{f}_i^{max}$ are defined as:

$$\begin{aligned} \mathfrak{f}_i^{min} \triangleq\ & \left[(\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii})\boldsymbol{\Xi}_{ii}(\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii})^T\right]^{-\frac{1}{2}} \times \\ & \left[-\mathbf{Y}_i^{min} - \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i + \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{A}}_{ii}\mathbf{x}_i(k) + \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{B}}_{ii}\mathbf{u}_i(k-1) - \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii}\boldsymbol{\mu}_i\right] \\ \mathfrak{f}_i^{max} \triangleq\ & \left[(\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii})\boldsymbol{\Xi}_{ii}(\mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii})^T\right]^{-\frac{1}{2}} \times \\ & \left[-\mathbf{Y}_i^{max} - \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i + \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{A}}_{ii}\mathbf{x}_i(k) + \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\bar{\mathbb{B}}_{ii}\mathbf{u}_i(k-1) - \mathbb{C}_{ii}\hat{\mathbb{A}}_{ii}^{-1}\hat{\mathbb{D}}_{ii}\boldsymbol{\mu}_i\right] \end{aligned} \tag{3.14}$$

3.2 Chance-Constrained CDMPC Problem Statement

The following general structure is proposed for the chance-constrained CDMPC:

$$\min_{\bar{\mathbf{X}}_i, \boldsymbol{\Delta}\mathbf{U}_i} \quad \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i + \{CoT\}_i \tag{3.15a}$$

subject to

$$\begin{aligned} \bar{\mathbf{x}}_i(k+l+1|k) = {} & \mathbf{A}_{ii}\bar{\mathbf{x}}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] + \mathbf{D}_{ii}\bar{\mathbf{d}}_i(k+l|k) \\ & + \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\alpha\mathbf{A}_{ij}\bar{\mathbf{x}}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \mathbf{D}_{ij}\bar{\mathbf{d}}_j(k+l|k)\right] + \mathbf{v}_i(k+l|k) \end{aligned} \tag{3.15b}$$

$$\begin{aligned} &\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k) \leq f^{min}_{i,g}(k+l+1|k)\} \geq \zeta^{\mathtt{L}}_{i,g,l} \\ &\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k) \geq f^{max}_{i,g}(k+l+1|k)\} \geq \zeta^{\mathtt{U}}_{i,g,l} \end{aligned} \tag{3.15c}$$

$$\begin{aligned} &\mathbf{u}^{min}_i(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}^{max}_i(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}^{min}_i(k+b) \leq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \leq \boldsymbol{\Delta}\mathbf{u}^{max}_i(k+b) \end{aligned} \tag{3.15d}$$

where the scalars $f^{min}_{i,g}(k+l+1|k)$ and $f^{max}_{i,g}(k+l+1|k)$ are elements of the vectors $\mathfrak{f}^{min}_i$ and $\mathfrak{f}^{max}_i$ defined in (3.9). Similar to the previous chapter, the *coordinating terms* $\{CoT\}_i$, connect the local controllers to the coordinator. These terms involve coordinating variables that are calculated and updated by the coordinator.

Comparing (3.15) with (3.13) shows that the probabilistic CDMPC is formed by modifying the already available decentralized MPC. Once the modifications are made, a coordinator is included. The modifications are made at two locations: 1) inclusion of known interaction terms[3] and predicted interaction vector $\mathbf{v}_i$ in local prediction model (3.15b); 2) inclusion of coordinating terms $\{CoT\}_i$ in the local objective function (3.15a).

In each coordination method, relaxation of the interaction equations determines the coordinating terms used in the CDMPC. The numerical strategy chosen to solve the optimization problem resulting from the aggregate of CDMPC, leads to coordinator synthesis.

3.3 Goal Coordinated Probabilistic DMPC

As discussed in Chapter 2, the core idea in the GC method is to first relax the interaction equations so that a separable optimization problem is produced and then, find the optimal solutions of the resulting separable optimization problem by numerically solving a relevant dual optimization problem. For the chance-constrained MPC problem, the interaction equations (3.2c) are similar to those in the deterministic case. Therefore, the procedure in Section 2.3, is followed to form the CDMPC and construct the coordinator. This means that, the term $\mathbf{p}^T\boldsymbol{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i \\ \boldsymbol{\Delta}\mathbf{U}_i \\ \mathbf{V}_i \end{bmatrix}$

[3]Since the coordinator has access to the complete plant model and variables, for implementation purposes, the known interaction terms can be calculated *in* the coordinator and sent to the CDMPC as a constant vector.

defines the $\{CoT\}_i$ in (3.15a), where the constant matrix $\mathbf{\Theta}_i$ contains the interaction models and is built according to (2.11b). Also, the gradient vector $\mathbf{J}$ and the Hessian $\mathbf{H}$, calculated by (2.22) and (2.23), are used in the coordinator's update equations (2.24). Therefore, applying the GC method to develop the chance-constrained CDMPC is straightforward.

In the GC method, a key issue to be taken care of, specially considering that probabilistic constraints are an important part of the local controllers, is coordinator's convergence problems in the presence of active local inequalities. As discussed in Chapter 2, in the Newton-based coordinator, active local inequalities can destroy the negative definiteness of the Hessian matrix, leading to poor convergence. On the other hand, using first-order gradient-based methods can be impractically slow and even oscillatory, specially near the optimum point[4]. Furthermore, both second and first-order gradient-based approaches use step-size in their update equations. Using an optimal step-size, which is calculated by performing a uni-variate optimization, can improve convergence; however, at each communication cycle, inner iterations are needed. Therefore, in what follows, rather than repeating the CDMPC and coordinator formulations of Chapter 2, a new first-order gradient-based coordinator with improved convergence, in the sense that it converges in finite number of iterations, is synthesized.

To this end a numerical optimization method is required that: 1) can efficiently solve the dual optimization problem; 2) only uses first-order gradient information; and 3) avoids optimal step-size calculations. These are the characteristics of the *method of multipliers* or the *augmented Lagrangian method* (Leunberger (1984), Nocedal and Wright (2006)), which is essentially a combination of the primal-dual and penalty methods. The main idea of this approach is to solve the constrained optimization problem by replacing it with a sequence of simpler problems. In doing so, the constraints to be relaxed are added to the Lagrangian by using Lagrange multipliers and also by quadratically penalizing them. The resulting augmented Lagrangian, which is a combination of the Lagrangian and quadratic penalty function, is an *exact penalty function*. Several algorithm have been designed that iteratively

[4]The optimum point corresponds to $\mathbf{p} = \mathbf{p}^*$ and is achieved when $\nabla_{\mathbf{p}}\varphi(\mathbf{p}) = \mathfrak{E} = \mathbf{0}$.

improve the value of the approximated Lagrange multipliers using the *gradient* of the augmented Lagrangian with respect to the Lagrange multipliers, which equals the relaxed constraints. At each iteration, the penalty parameter and the estimate of Lagrange multipliers are fixed and simpler optimizations are performed with respect to the decision variables. Upon convergence of the algorithm, the optimal solution to the original optimization problem is found. More details on augmented Lagrangian methods can be found in Nocedal and Wright (2006), Hamdi and Mishra (2011) and references therein.

If the method of multipliers is applied to the plant-wide MPC problem, the interaction equations will be the constraints to be relaxed and quadratically penalized; however, their squared Euclidean norm is not separable, i.e., $||\sum_{i=1}^{N} \boldsymbol{\Theta}_i \mathbf{Z}_i||^2 \neq ||\boldsymbol{\Theta}_1 \mathbf{Z}_1||^2 + \cdots + ||\boldsymbol{\Theta}_N \mathbf{Z}_N||^2$. The non-separability issue is addressed by employing the Separable Augmented Lagrangian Algorithms (SALA) proposed by Hamdi *et al.* (1997). First, the additive separable form of the interaction equations (3.2c), are replaced by the following equivalent equations:

$$\boldsymbol{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i \\ \boldsymbol{\Delta}\mathbf{U}_i \\ \mathbf{V}_i \end{bmatrix} + \boldsymbol{\rho}_i = \mathbf{0} \qquad i = 1, \ldots, N \tag{3.16a}$$

$$\sum_{i=1}^{N} \boldsymbol{\rho}_i = \mathbf{0} \tag{3.16b}$$

where $\boldsymbol{\rho}_i$ is the allocation vector with the dimension $H_p \times n_{xi}$. It is clear that the squared Euclidean norm of (3.16a) is separable. Now, the same steps discussed in Section 2.3 can be followed. The only difference is that, the complicating constraints (3.16a) are relaxed by associating Lagrange multipliers $\mathbf{p}$ with them, *and* quadratically penalized using penalty parameter μ. Ultimately, the following chance-constrained CDMPC is obtained:

$$\min_{\bar{\mathbf{X}}_i,\boldsymbol{\Delta}\mathbf{U}_i,\mathbf{V}_i} \quad \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i-\mathbf{r}_i\right)^T\mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i-\mathbf{r}_i\right)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i + \mathbf{p}^T\left(\boldsymbol{\Theta}_i\begin{bmatrix}\bar{\mathbf{X}}_i\\ \boldsymbol{\Delta}\mathbf{U}_i\\ \mathbf{V}_i\end{bmatrix}+\boldsymbol{\rho}_i\right)+\frac{1}{2\mu}\left\|\boldsymbol{\Theta}_i\begin{bmatrix}\bar{\mathbf{X}}_i\\ \boldsymbol{\Delta}\mathbf{U}_i\\ \mathbf{V}_i\end{bmatrix}+\boldsymbol{\rho}_i\right\|^2 \tag{3.17a}$$

subject to

$$\bar{\mathbf{x}}_i(k+l+1|k)=\mathbf{A}_{ii}\bar{\mathbf{x}}_i(k+l|k)+\mathbf{B}_{ii}\left[\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k)+\mathbf{u}_i(k-1)\right]+\mathbf{D}_{ii}\bar{\mathbf{d}}_i(k+l|k) + \sum_{\substack{j=1\\ j\neq i}}^{N}\left[\alpha\mathbf{A}_{ij}\bar{\mathbf{x}}_j(k+l|k)+\mathbf{B}_{ij}\mathbf{u}_j(k-1)+\mathbf{D}_{ij}\bar{\mathbf{d}}_j(k+l|k)\right]+\mathbf{v}_i(k+l|k) \tag{3.17b}$$

$$\begin{aligned}&\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k)\leq f_{i,g}^{min}(k+l+1|k)\}\geq\zeta_{i,g,l}^{\mathtt{L}}\\ &\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k)\geq f_{i,g}^{max}(k+l+1|k)\}\geq\zeta_{i,g,l}^{\mathtt{U}}\end{aligned} \tag{3.17c}$$

$$\begin{aligned}&\mathbf{u}_i^{min}(k+b)\leq\mathbf{u}_i(k+b|k)\leq\mathbf{u}_i^{max}(k+b)\\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b)\leq\boldsymbol{\Delta}\mathbf{u}_i(k+b|k)\leq\boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b)\end{aligned} \tag{3.17d}$$

where the last two terms in the objective function are the coordinating term $\{CoT\}_i$. The price vector $\mathbf{p}$ and the allocation vector $\boldsymbol{\rho}_i$ are the coordinating variables.

At this stage the coordinator is designed. The objective function of the aggregate of distributed controllers (3.17) is:

$$\mathcal{J}(\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\mathbf{V},\mathbf{p},\boldsymbol{\rho})=\sum_{i=1}^{N}\left[\left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i-\mathbf{r}_i\right)^T\mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i-\mathbf{r}_i\right)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\right] + \mathbf{p}^T\sum_{i=1}^{N}[\boldsymbol{\Theta}_i\begin{bmatrix}\bar{\mathbf{X}}_i\\ \boldsymbol{\Delta}\mathbf{U}_i\\ \mathbf{V}_i\end{bmatrix}+\boldsymbol{\rho}_i]+\frac{1}{2\mu}\sum_{i=1}^{N}\left\|\boldsymbol{\Theta}_i\begin{bmatrix}\bar{\mathbf{X}}_i\\ \boldsymbol{\Delta}\mathbf{U}_i\\ \mathbf{V}_i\end{bmatrix}+\boldsymbol{\rho}_i\right\|^2 \tag{3.18}$$

The Lagrangian dual function is defined as:

$$\varphi(\mathbf{p},\boldsymbol{\rho})\triangleq\inf_{\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\mathbf{V}}\quad\{\mathcal{J}(\mathbf{X},\boldsymbol{\Delta}\mathbf{U},\mathbf{V},\mathbf{p},\boldsymbol{\rho})|(3.17b),\quad(3.17c)\quad\&\quad(3.17d),\quad i=1,\ldots,N\} \tag{3.19}$$

and the resulting dual optimization problem becomes:

$$\max_{\mathbf{p},\boldsymbol{\rho}}\quad\varphi(\mathbf{p},\boldsymbol{\rho}) \tag{3.20}$$

Using the proposed algorithm in SALA[5] to solve the dual optimization problem (3.20), the following update equations for the coordinator are obtained:

$$\mathbf{p}^{q+1} = \mathbf{p}^q + \frac{1}{\mu\left(H_p \sum_{i=1}^{N} n_{xi}\right)} \sum_{i=1}^{N} \mathbf{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i^q \\ \mathbf{\Delta U}_i^q \\ \mathbf{V}_i^q \end{bmatrix} \tag{3.21}$$

$$\boldsymbol{\rho}_i^{q+1} = -\mathbf{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i^q \\ \mathbf{\Delta U}_i^q \\ \mathbf{V}_i^q \end{bmatrix} + \frac{1}{\left(H_p \sum_{i=1}^{N} n_{xi}\right)} \sum_{i=1}^{N} \mathbf{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i^q \\ \mathbf{\Delta U}_i^q \\ \mathbf{V}_i^q \end{bmatrix} \qquad i = 1, \ldots, N \tag{3.22}$$

$$\mu^{q+1} = \kappa \mu^q \tag{3.23}$$

where $0 < \kappa \leq 1$. It should be noted that finding the most effective value for the penalty parameter μ improves the efficiency of SALA (Dussault *et al.* (2005)); however, in the current work, the simple update equation (3.23), as was suggested in the original SALA, is used.

As expected, since SALA numerically solves the dual optimization problem, a hierarchical structure is obtained for the coordinated distributed network. During each sampling time, the probabilistic CDMPC (3.17) and the coordinator (3.21,3.22 and 3.23) exchange information until convergence, where:

1. The value of price vector $\mathbf{p}$ equals the Lagrange multipliers $\mathbf{p}^*$, associated with the equality constraints in the plant-wide MPC problem.
2. The sum of allocation vectors $\boldsymbol{\rho}_i$ becomes zero ($\sum_{i=1}^{N} \boldsymbol{\rho}_i = \mathbf{0}$).
3. The interaction equality constraints are satisfied.

One of the advantages of using SALA in synthesizing the coordinator, is that the need to calculate the optimal step-size is removed. Also, for faster convergence, the penalty parameter can be changed; however, this value affects the coordinator's convergence rate, not the accuracy of coordinator's convergence to the plant-wide optimal solution. That is, regardless of what value μ has, the coordinator will converge and achieve the centralized solution.

[5]Details on the steps involved in SALA, can be found in Hamdi *et al.* (1997), Dussault *et al.* (2005) and Hamdi and Mishra (2011).

It should be emphasized that the coordinator's performance improvement presented in this section can be employed in the deterministic CDMPC discussed in section 2.3, as well.

Implementation of the coordinated distributed chance-constrained MPC network, developed using SALA, can be systematically done according to Algorithm 4.

Algorithm 4 : Implementation of Probabilistic GC(SALA)-CDMPC Network

---

1. Coordinator: Iteration counter q is set to 1.
2. Coordinator: Price vector $\mathbf{p}$ is arbitrarily initialized.
3. Coordinator: Allocation vectors $\boldsymbol{\rho}_i$, are initialized such that $\sum_{i=1}^{N} \boldsymbol{\rho}_i^0 = \mathbf{0}$.
4. Coordinator: Penalty parameter μ is arbitrarily chosen.
5. Coordinator: Coordinating variables and penalty parameter are sent to the local controllers.
6. Local Controllers: Local optimization problems (3.17) are solved.
7. Local Controllers: Local optimal solutions $\bar{\mathbf{X}}_i$, $\boldsymbol{\Delta}\mathbf{U}_i$ and $\mathbf{V}_i$ are sent to the coordinator.
8. Coordinator: If $\mathfrak{E} = \sum_{i=1}^{N} \boldsymbol{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i^q \\ \boldsymbol{\Delta}\mathbf{U}_i^q \\ \mathbf{V}_i^q \end{bmatrix} \leq \mathfrak{e}$ stops. Otherwise, continue.
9. Coordinator: Coordinating variables and the penalty parameter are updated using (3.21), (3.22) and (3.23).
10. Coordinator: Iteration counter is increased by 1.
11. Steps 5-8 are repeated.

3.4 Interaction Prediction Coordinated Probabilistic DMPC

Considering that in the plant-wide probabilistic MPC problem (3.2), the interaction equations are similar to the deterministic case, in applying the IPC method to convert the probabilistic decentralized MPC into chance-constrained CDMPC, the

coordinating term is defined the same way as in section 2.4. Accordingly, the chance-constrained CDMPC (3.15) becomes:

$$\min_{\bar{\mathbf{X}}_i, \boldsymbol{\Delta}\mathbf{U}_i} \quad \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i + \mathbf{p}^T \boldsymbol{\Phi}_i \begin{bmatrix} \bar{\mathbf{X}}_i \\ \boldsymbol{\Delta}\mathbf{U}_i \end{bmatrix} \tag{3.24a}$$

subject to

$$\bar{\mathbf{x}}_i(k+l+1|k) = \mathbf{A}_{ii}\bar{\mathbf{x}}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] + \mathbf{D}_{ii}\bar{\mathbf{d}}_i(k+l|k) + \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\alpha\mathbf{A}_{ij}\bar{\mathbf{x}}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \mathbf{D}_{ij}\bar{\mathbf{d}}_j(k+l|k)\right] + \mathbf{v}_i(k+l|k) \tag{3.24b}$$

$$\begin{aligned} &\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k) \leq f_{i,g}^{min}(k+l+1|k)\} \geq \zeta_{i,g,l}^{\mathtt{L}} \\ &\mathtt{Pr}\{\xi'_{i,g}(k+l+1|k) \geq f_{i,g}^{max}(k+l+1|k)\} \geq \zeta_{i,g,l}^{\mathtt{U}} \end{aligned} \tag{3.24c}$$

$$\begin{aligned} &\mathbf{u}_i^{min}(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}_i^{max}(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \leq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \leq \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b) \end{aligned} \tag{3.24d}$$

where the vectors $\mathbf{p}$ and $\mathbf{v}_i$ are the coordinating variables.

The coordinator in the IPC method is designed by numerically solving portions of first-order optimality conditions for the optimization problem obtained by the aggregate of CDMPCs. In doing so, the following compact form for the composite CDMPC (3.24) will be used:

$$\min_{\bar{\mathbf{X}}_i, \boldsymbol{\Delta}\mathbf{U}_i} \quad \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i + \mathbf{p}^T \boldsymbol{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i \\ \boldsymbol{\Delta}\mathbf{U}_i \\ \mathbf{V}_i \end{bmatrix} \tag{3.25a}$$

subject to

$$\hat{\mathbb{A}}_{ii}\bar{\mathbf{X}}_i + \hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i - \mathbf{V}_i = \overbrace{\bar{\mathbb{A}}_{ii}\mathbf{x}_i(k) + \bar{\mathbb{B}}_{ii}\mathbf{u}_i(k) - \hat{\mathbb{D}}_{ii}\bar{\mathfrak{D}}_i + \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\bar{\mathbb{A}}_{ij}\mathbf{x}_j + \bar{\mathbb{B}}_{ij}\mathbf{u}_j(k-1) - \hat{\mathbb{D}}_{ij}\bar{\mathfrak{D}}_j\right]}^{\mathbf{z}_i} \tag{3.25b}$$

$$\mathfrak{f}_i^{min} \geq \mathfrak{F}_i^{\mathtt{L}} \tag{3.25c}$$

$$\mathfrak{f}_i^{max} \leq \mathfrak{F}_i^{\mathtt{U}}$$

$$\mathfrak{U}_i^{min} \leq \mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i \leq \mathfrak{U}_i^{max} \tag{3.25d}$$

where (3.25b) is the compact form (C.6), in which the expected values are used for the predicted states and future disturbances. The single probabilistic constraints are

written in terms of their equivalent inequality constraints according to (3.12). Details on obtaining the compact form for the constraints on the manipulated variables and their changes can be found in Appendix B, Section B.3.2. The vector $\mathbf{Z}_i$ contains all the known terms in the local model.

The Lagrangian of problem (3.25) is then formed as:

$$
\begin{aligned}
&L(\bar{\mathbf{X}}, \boldsymbol{\Delta}\mathbf{U}, \mathbf{V}, \boldsymbol{\Lambda}, \boldsymbol{\Omega}, \boldsymbol{\Upsilon}, \mathbf{p}) = \sum_{i=1}^{N} L_i(\bar{\mathbf{X}}_i, \boldsymbol{\Delta}\mathbf{U}_i, \mathbf{V}_i, \boldsymbol{\Lambda}_i, \boldsymbol{\Omega}_i, \boldsymbol{\Upsilon}_i, \mathbf{p}) = \\
&\sum_{i=1}^{N} \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii} \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i + \mathbf{p}^T \sum_{i=1}^{N} \boldsymbol{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i \\ \boldsymbol{\Delta}\mathbf{U}_i \\ \mathbf{V}_i \end{bmatrix} + \\
&\sum_{i=1}^{N} \boldsymbol{\Lambda}_i^T \left(\hat{\mathbb{A}}_{ii}\bar{\mathbf{X}}_i + \hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i - \mathbf{V}_i - \mathcal{Z}_i\right) + \\
&\sum_{i=1}^{N} \boldsymbol{\Omega}_{i,min}^T \left(\mathfrak{F}_i^{\mathtt{L}} - \mathfrak{f}_i^{min}\right) + \boldsymbol{\Omega}_{i,max}^T \left(\mathfrak{f}_i^{max} - \mathfrak{F}_i^{\mathtt{U}}\right) + \\
&\sum_{i=1}^{N} \boldsymbol{\Upsilon}_{i,min}^T \left(\mathfrak{U}_i^{min} - \mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\right) + \boldsymbol{\Upsilon}_{i,max}^T \left(\mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i - \mathfrak{U}_i^{max}\right)
\end{aligned}
\tag{3.26}
$$

where $\boldsymbol{\Lambda}$, $\boldsymbol{\Omega}$ and $\boldsymbol{\Upsilon}$ are the Lagrange multipliers associated with the relevant constraints.

Choosing the fixed-point iteration technique to numerically solve the optimality conditions $\nabla_{\mathbf{p}} L = \mathbf{0}$ and $\nabla_{\mathbf{V}} L = \mathbf{0}$, provides the update equations for the coordinating variables $\mathbf{p}$ and $\mathbf{V}_i$, and consequently design the coordinator. Satisfaction of the remaining optimality conditions is taken care of by the distributed controllers (3.24). Similar to Section 2.4, taking the derivative of the Lagrangian with respect to the price vector yields:

$$
\begin{aligned}
&\nabla_{\mathbf{p}} L = \sum_{i=1}^{N} \boldsymbol{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i^q \\ \boldsymbol{\Delta}\mathbf{U}_i^q \\ \mathbf{V}_i^{q+1} \end{bmatrix} = \mathbf{0} \implies \\
&\mathbf{v}_i^{q+1}(k+l|k) = \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\beta \mathbf{A}_{ij} \bar{\mathbf{x}}_j^q(k+l|k) + \mathbf{B}_{ij} \sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_j^q(k+a|k)\right] \\
&\text{for } i = 1, \ldots, N
\end{aligned}
\tag{3.27}
$$

Taking the derivative of Lagrangian with respect to the predicted interaction effects

$\mathbf{v}_i$, and equaling to zero lead to the following equations:

$$\nabla_{\mathbf{V}} L = \sum_{i=1}^{N} \nabla_{\mathbf{V}_i} L = \mathbf{0} \implies$$
$$\mathbf{p}^{q+1} \begin{bmatrix} \mathbf{I}_1 & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{I}_N \end{bmatrix} - \begin{bmatrix} \mathbf{\Lambda}_1^q \\ \vdots \\ \mathbf{\Lambda}_N^q \end{bmatrix} - \begin{bmatrix} (\frac{d\mathfrak{f}_1^{min}}{d\mathbf{V}_1})^T \mathbf{\Omega}_{1,min}^q \\ \vdots \\ (\frac{d\mathfrak{f}_N^{min}}{d\mathbf{V}_N})^T \mathbf{\Omega}_{N,min}^q \end{bmatrix} + \begin{bmatrix} (\frac{d\mathfrak{f}_1^{max}}{d\mathbf{V}_1})^T \mathbf{\Omega}_{1,max}^q \\ \vdots \\ (\frac{d\mathfrak{f}_N^{max}}{d\mathbf{V}_N})^T \mathbf{\Omega}_{N,max}^q \end{bmatrix} = \mathbf{0} \tag{3.28}$$

The derivatives $\frac{d\mathfrak{f}_i^*}{d\mathbf{V}_i}$ appear in (3.28) are calculated by using (3.9) as:

$$\frac{d\mathfrak{f}_i^*}{d\mathbf{V}_i} = \mathbb{G}_i \mathbb{C}_{ii} \hat{\mathbb{A}}_{ii}^{-1} \tag{3.29}$$

Replacing (3.29) in (3.28) and re-arranging the terms result in the following update equations for the price vector $\mathbf{p}$:

$$\mathbf{p}^{q+1} = \begin{bmatrix} \mathbf{\Lambda}_1^q \\ \vdots \\ \mathbf{\Lambda}_N^q \end{bmatrix} + \begin{bmatrix} (\mathbb{G}_1 \mathbb{C}_{11} \hat{\mathbb{A}}_{11}^{-1})^T \mathbf{\Omega}_{1,min}^q \\ \vdots \\ (\mathbb{G}_N \mathbb{C}_{NN} \hat{\mathbb{A}}_{NN}^{-1})^T \mathbf{\Omega}_{N,min}^q \end{bmatrix} - \begin{bmatrix} (\mathbb{G}_1 \mathbb{C}_{11} \hat{\mathbb{A}}_{11}^{-1})^T \mathbf{\Omega}_{1,max}^q \\ \vdots \\ (\mathbb{G}_N \mathbb{C}_{NN} \hat{\mathbb{A}}_{NN}^{-1})^T \mathbf{\Omega}_{N,max}^q \end{bmatrix} \tag{3.30}$$

During each sampling time, the probabilistic CDMPC (3.24) and the coordinator (3.27 and 3.30) exchange information until $||\mathbf{V}^{q+1} - \mathbf{V}^q|| \leq \mathfrak{e}$. Upon convergence, the value of the price vector $\mathbf{p}$ becomes equal to the Lagrange multiplier value $\mathbf{p}^*$ associated with equality constraints of the plant-wide probabilistic MPC problem.

Remark 3.4.1 *If rather than fixed-point iterations, a gradient-based method is used to numerically solve $\nabla_{\mathbf{p}} L = \mathbf{0}$ and $\nabla_{\mathbf{V}} L = \mathbf{0}$, the following coordinator will be obtained:*

$$\mathbf{p}^{q+1} = \mathbf{p}^q + \epsilon_1 (\overbrace{\sum_{i=1}^{N} \mathbf{\Theta}_i \begin{bmatrix} \bar{\mathbf{X}}_i^q \\ \mathbf{\Delta U}_i^q \\ \mathbf{V}_i^q \end{bmatrix}}^{(\nabla_{\mathbf{p}} L)^q}) \tag{3.31}$$

$$\mathbf{V}^{q+1} = \mathbf{V}^q - \epsilon_2 (\overbrace{\mathbf{p}^q - \begin{bmatrix} \mathbf{\Lambda}_1^q \\ \vdots \\ \mathbf{\Lambda}_N^q \end{bmatrix} - \begin{bmatrix} (\mathbb{G}_1 \mathbb{C}_{11} \hat{\mathbb{A}}_{11}^{-1})^T \mathbf{\Omega}_{1,min}^q \\ \vdots \\ (\mathbb{G}_N \mathbb{C}_{NN} \hat{\mathbb{A}}_{NN}^{-1})^T \mathbf{\Omega}_{N,min}^q \end{bmatrix} + \begin{bmatrix} (\mathbb{G}_1 \mathbb{C}_{11} \hat{\mathbb{A}}_{11}^{-1})^T \mathbf{\Omega}_{1,max}^q \\ \vdots \\ (\mathbb{G}_N \mathbb{C}_{NN} \hat{\mathbb{A}}_{NN}^{-1})^T \mathbf{\Omega}_{N,max}^q \end{bmatrix}}^{(\nabla_{\mathbf{V}} L)^q}) \tag{3.32}$$

where $i = 1, \ldots, N$; and ϵ_1 and ϵ_2 are tuning parameters.

3.5 Modified Pseudo-Model Chance-Constrained CDMPC

Similar to the other two coordination methods, applying the MPMC approach to the chance-constrained MPC problem, is a straightforward process. The pseudo-variables $\tilde{\mathbf{x}}_i(k)$ and $\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k)$ are defined using the following transformed interaction equations:

$$\tilde{\mathbf{x}}_i(k) = \bar{\mathbf{x}}_i(k) \tag{3.33a}$$

$$\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k) = \boldsymbol{\Delta}\mathbf{u}_i(k) \tag{3.33b}$$

Following the procedures discussed in Section 2.5, the chance-constrained CDMPC will have the following structure:

$$\begin{aligned}\min_{\bar{\mathbf{X}}_i, \boldsymbol{\Delta}\mathbf{U}_i} \quad & \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i + \\ & \frac{\varepsilon}{2}\left(||\bar{\mathbf{X}}_i - \tilde{\mathbf{X}}_i||^2_{\bar{\mathbb{Q}}_i} + ||\boldsymbol{\Delta}\mathbf{U}_i - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\bar{\mathbb{R}}_i}\right) + \\ & \boldsymbol{\Gamma}_i^T(\bar{\mathbf{X}}_i - \tilde{\mathbf{X}}_i) + \boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i)\end{aligned} \tag{3.34a}$$

subject to

$$\begin{aligned}\bar{\mathbf{x}}_i(k+l+1|k) = \mathbf{A}_{ii}\bar{\mathbf{x}}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] + \mathbf{D}_{ii}\bar{\mathbf{d}}_i(k+l|k) \\ + \sum_{\substack{j=1 \\ j\neq i}}^{N}\left[\alpha\mathbf{A}_{ij}\bar{\mathbf{x}}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \mathbf{D}_{ij}\bar{\mathbf{d}}_j(k+l|k)\right] + \mathbf{v}_i(k+l|k)\end{aligned} \tag{3.34b}$$

$$\begin{aligned}&\Pr\{\xi'_{i,g}(k+l+1|k) \leq f^{min}_{i,g}(k+l+1|k)\} \geq \zeta^{\mathtt{L}}_{i,g,l} \\ &\Pr\{\xi'_{i,g}(k+l+1|k) \geq f^{max}_{i,g}(k+l+1|k)\} \geq \zeta^{\mathtt{U}}_{i,g,l}\end{aligned} \tag{3.34c}$$

$$\begin{aligned}&\mathbf{u}^{min}_i(k+b) \leq \mathbf{u}_i(k+b|k) \leq \mathbf{u}^{max}_i(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}^{min}_i(k+b) \leq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \leq \boldsymbol{\Delta}\mathbf{u}^{max}_i(k+b)\end{aligned} \tag{3.34d}$$

where $\tilde{\mathbf{X}}_i$, $\boldsymbol{\Delta}\tilde{\mathbf{U}}_i$, $\boldsymbol{\Gamma}_i$, $\boldsymbol{\Pi}_i$ and $\mathbf{V}_i$ are coordinating variables.

The numerical method used in solving portions of the optimality conditions of the aggregate of distributed controllers (3.34), dictates the coordinator formulation. To work with a simpler problem, the aggregate problem is written in the following

compact form:

$$\begin{aligned}\min_{\bar{\mathbf{X}}_i,\boldsymbol{\Delta}\mathbf{U}_i} \quad & \left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i-\mathbf{r}_i\right)^T\mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i-\mathbf{r}_i\right)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i+\\ & \frac{\varepsilon}{2}\left(||\bar{\mathbf{X}}_i-\tilde{\mathbf{X}}_i||^2_{\bar{\mathbb{Q}}_i}+||\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\bar{\mathbb{R}}_i}\right)+ \\ & \boldsymbol{\Gamma}_i^T(\bar{\mathbf{X}}_i-\tilde{\mathbf{X}}_i)+\boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i)\end{aligned} \tag{3.35a}$$

subject to

$$\begin{aligned}&\hat{\mathbb{A}}_{ii}\bar{\mathbf{X}}_i+\hat{\mathbb{B}}_{ii}\boldsymbol{\Delta}\mathbf{U}_i+\sum_{\substack{j=1\\j\neq i}}^{N}\left[\hat{\mathbb{A}}_{ij}\tilde{\mathbf{X}}_j+\hat{\mathbb{B}}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{U}}_j\right]= \\ &\overbrace{\bar{\mathbb{A}}_{ii}\mathbf{x}_i(k)+\bar{\mathbb{B}}_{ii}\mathbf{u}_i(k)-\hat{\mathbb{D}}_{ii}\bar{\mathfrak{D}}_i+\sum_{\substack{j=1\\j\neq i}}^{N}\left[\bar{\mathbb{A}}_{ij}\mathbf{x}_j+\bar{\mathbb{B}}_{ij}\mathbf{u}_j(k-1)-\hat{\mathbb{D}}_{ij}\bar{\mathfrak{D}}_j\right]}^{\mathbf{z}_i}\end{aligned} \tag{3.35b}$$

$$\begin{aligned}&\mathfrak{f}_i^{min}\geq\mathfrak{F}_i^{\mathtt{L}} \\ &\mathfrak{f}_i^{max}\leq\mathfrak{F}_i^{\mathtt{U}}\end{aligned} \tag{3.35c}$$

$$\mathfrak{U}_i^{min}\leq\mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\leq\mathfrak{U}_i^{max} \tag{3.35d}$$

where (3.35b) is the compact form (C.11), in which the expected values are used for the predicted states and future disturbances. The single probabilistic constraints are written in terms of their equivalent inequality constraints according to (3.12). Details for obtaining the compact form of the constraints for the manipulated variables and their changes can be found in Appendix B, Section B.3.2. The vector $\mathbf{Z}_i$ contains known information in the local prediction models.

The Lagrangian of problem (3.35) is written as:

$$\begin{aligned}&L(\bar{\mathbf{X}},\boldsymbol{\Delta}\mathbf{U},\tilde{\mathbf{X}},\boldsymbol{\Delta}\tilde{\mathbf{U}},\boldsymbol{\Lambda},\boldsymbol{\Omega},\boldsymbol{\Upsilon},\boldsymbol{\Gamma},\boldsymbol{\Pi})=\sum_{i=1}^{N}L_i(\bar{\mathbf{X}}_i,\boldsymbol{\Delta}\mathbf{U}_i,\tilde{\mathbf{X}}_i,\boldsymbol{\Delta}\tilde{\mathbf{U}}_i,\boldsymbol{\Lambda}_i,\boldsymbol{\Omega}_i,\boldsymbol{\Upsilon}_i,\boldsymbol{\Gamma}_i,\boldsymbol{\Pi}_i) \\ &=\sum_{i=1}^{N}\left(\left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i-\mathbf{r}_i\right)^T\mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\bar{\mathbf{X}}_i-\mathbf{r}_i\right)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\right) \\ &+\sum_{i=1}^{N}\frac{\varepsilon}{2}\left(||\bar{\mathbf{X}}_i-\tilde{\mathbf{X}}_i||^2_{\bar{\mathbb{Q}}_i}+||\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\bar{\mathbb{R}}_i}\right) \\ &+\sum_{i=1}^{N}\left(\boldsymbol{\Gamma}_i^T(\bar{\mathbf{X}}_i-\tilde{\mathbf{X}}_i)+\boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i)\right)\end{aligned}$$

$$
\begin{aligned}
&+\sum_{i=1}^{N}\mathbf{\Lambda}_i^T\left(\hat{\mathbb{A}}_{ii}\bar{\mathbf{X}}_i+\hat{\mathbb{B}}_{ii}\mathbf{\Delta U}_i+\sum_{\substack{j=1\\j\neq i}}^{N}\left[\hat{\mathbb{A}}_{ij}\tilde{\mathbf{X}}_j+\hat{\mathbb{B}}_{ij}\mathbf{\Delta}\tilde{\mathbf{U}}_j\right]-\mathcal{Z}_i\right)\\
&+\sum_{i=1}^{N}\mathbf{\Omega}_{i,min}^T\left(\mathfrak{F}_i^{\mathtt{L}}-\mathfrak{f}_i^{min}\right)+\mathbf{\Omega}_{i,max}^T\left(\mathfrak{f}_i^{max}-\mathfrak{F}_i^{\mathtt{U}}\right)+\\
&+\sum_{i=1}^{N}\mathbf{\Upsilon}_{i,min}^T\left(\mathfrak{U}_i^{min}-\mathbb{U}_{ii}\mathbf{\Delta U}_i\right)+\mathbf{\Upsilon}_{i,max}^T\left(\mathbb{U}_{ii}\mathbf{\Delta U}_i-\mathfrak{U}_i^{max}\right)
\end{aligned}
\tag{3.36}
$$

In designing the coordinator, the optimality conditions $\nabla_{\tilde{\mathbf{X}}}L=\mathbf{0}$, $\nabla_{\mathbf{\Delta}\tilde{\mathbf{U}}}L=\mathbf{0}$, $\nabla_{\mathbf{\Gamma}}L=\mathbf{0}$ and $\nabla_{\mathbf{\Pi}}L=\mathbf{0}$ are used to obtain the prediction equations for the coordinating variables. The rest of the optimality conditions are handled by the probabilistic CDMPC (3.34).

If a fixed-point iteration method is used, the update equations for the pseudo-variables $\tilde{\mathbf{X}}$ and $\mathbf{\Delta}\tilde{\mathbf{U}}$ will be obtained as:

$$
\begin{cases}
\nabla_{\mathbf{\Gamma}}L=\sum_{i=1}^{N}\nabla_{\mathbf{\Gamma}_i}L=\mathbf{0}\\
\nabla_{\mathbf{\Pi}}L=\sum_{i=1}^{N}\nabla_{\mathbf{\Pi}_i}L=\mathbf{0}
\end{cases}
\tag{3.37a}
$$

$$
\Longrightarrow
$$

$$
\text{for i=1,...,N :}\quad
\begin{cases}
\tilde{\mathbf{X}}_i^{q+1}=\bar{\mathbf{X}}_i^{q}\\
\mathbf{\Delta}\tilde{\mathbf{U}}_i^{q+1}=\mathbf{\Delta U}_i^{q}
\end{cases}
\tag{3.37b}
$$

Thus, at each communication cycle, the updated values of pseudo-variables $\tilde{\mathbf{X}}$ and $\mathbf{\Delta}\tilde{\mathbf{U}}$ equal the current values of local optimal decision variables.

Similarly, at each iteration, the value of the price vectors $\mathbf{\Gamma}_i$ and $\mathbf{\Pi}_i$ are corrected by a combination of current local Lagrange multipliers and interaction matrices along with the applicable updated pseudo-variables, as follows:

$$
\begin{cases}
\nabla_{\tilde{\mathbf{X}}}L=\sum_{i=1}^{N}\nabla_{\tilde{\mathbf{X}}_i}L=\mathbf{0}\\
\nabla_{\mathbf{\Delta}\tilde{\mathbf{U}}}L=\sum_{i=1}^{N}\nabla_{\mathbf{\Delta}\tilde{\mathbf{U}}_i}L=\mathbf{0}
\end{cases}
\tag{3.38a}
$$

Consequently the following update equations are obtained for i=1,...,N :

$$\begin{cases} \boldsymbol{\Gamma}_i^{q+1} = \\ -\varepsilon(\bar{\mathbf{X}}_i^q - \tilde{\mathbf{X}}_i^{q+1})\bar{\mathbb{Q}}_i + \sum\limits_{\substack{j=1 \\ j\neq i}}^{N} \hat{\mathbb{A}}_{ji}^T \boldsymbol{\Lambda}_j^q + \sum\limits_{\substack{j=1 \\ j\neq i}}^{N} (\mathbb{G}_j \mathbb{C}_{jj} \hat{\mathbb{A}}_{jj}^{-1} \hat{\mathbb{A}}_{ji})^T \boldsymbol{\Omega}_{j,min}^q - \sum\limits_{\substack{j=1 \\ j\neq i}}^{N} (\mathbb{G}_j \mathbb{C}_{jj} \hat{\mathbb{A}}_{jj}^{-1} \hat{\mathbb{A}}_{ji})^T \boldsymbol{\Omega}_{j,max}^q \\ \boldsymbol{\Pi}_i^{q+1} = \\ -\varepsilon(\boldsymbol{\Delta}\mathbf{U}_i^q - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^{q+1})\bar{\mathbb{R}}_i + \sum\limits_{\substack{j=1 \\ j\neq i}}^{N} \hat{\mathbb{B}}_{ji}^T \boldsymbol{\Lambda}_j^q + \sum\limits_{\substack{j=1 \\ j\neq i}}^{N} (\mathbb{G}_j \mathbb{C}_{jj} \hat{\mathbb{A}}_{jj}^{-1} \hat{\mathbb{B}}_{ji})^T \boldsymbol{\Omega}_{j,min}^q - \sum\limits_{\substack{j=1 \\ j\neq i}}^{N} (\mathbb{G}_j \mathbb{C}_{jj} \hat{\mathbb{A}}_{jj}^{-1} \hat{\mathbb{B}}_{ji})^T \boldsymbol{\Omega}_{j,max}^q \end{cases} \tag{3.38b}$$

where in order to obtain $\frac{d\mathfrak{f}^*}{d\bar{\mathbf{X}}}$ and $\frac{d\mathfrak{f}^*}{d\boldsymbol{\Delta}\tilde{\mathbf{U}}}$, the interaction term $\mathbf{V}_i$ in (3.9), has been replaced by the equivalent expression $\sum\limits_{\substack{j=1 \\ j\neq i}}^{N} \Big[\hat{\mathbb{A}}_{ij}\tilde{\mathbf{X}}_j + \hat{\mathbb{B}}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{U}}_j\Big]$.

The values of the predicted interaction vectors $\mathbf{v}_i$, are corrected using updated values of the pseudo-variables in the interaction equations:

for i=1,. . . ,N :

$$\mathbf{v}_i^{q+1}(k+l|k) = \sum_{\substack{j=1 \\ j\neq i}}^{N} \left[\beta \mathbf{A}_{ij} \tilde{\mathbf{x}}_j^{q+1}(k+l|k) + \mathbf{B}_{ij} \sum_{a=0}^{l} \boldsymbol{\Delta}\tilde{\mathbf{u}}_j^{q+1}(k+a|k)\right] \tag{3.39}$$

During each sampling time k, the chance-constrained CDMPCs (3.34) communicate with the coordinator (3.37b, 3.38b and 3.39) until the error, which is defined as $\left|\left|\begin{bmatrix} \bar{\mathbf{X}} - \tilde{\mathbf{X}} \\ \boldsymbol{\Delta}\mathbf{U} - \boldsymbol{\Delta}\tilde{\mathbf{U}} \end{bmatrix}\right|\right|$, is equal to or less than the pre-specified tolerance $\mathfrak{e}$. Satisfying this stopping criteria means that, the price vectors $\boldsymbol{\Gamma}_i$ and $\boldsymbol{\Pi}_i$ equal Lagrange multipliers associated with the corresponding transformed interaction equality constraints. Thus, the values of pseudo-variables become equal to the correct values of interaction variables and consequently, the interaction equations are satisfied.

Remark 3.5.1 *If rather than fixed-point iteration method, a gradient-based method is used to numerically solve the optimality conditions* $\nabla_{\boldsymbol{\Gamma}} L = \mathbf{0}$, $\nabla_{\boldsymbol{\Pi}} L = \mathbf{0}$, $\nabla_{\tilde{\mathbf{X}}} L = \mathbf{0}$

and $\nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}} L = \mathbf{0}$*, the coordinator's update equations become:*

$$\boldsymbol{\Gamma}_i^{q+1} = \boldsymbol{\Gamma}_i^q + \epsilon_1[\bar{\mathbf{X}}_i^q - \tilde{\mathbf{X}}_i^q] \tag{3.40}$$

$$\boldsymbol{\Pi}_i^{q+1} = \boldsymbol{\Pi}_i^q + \epsilon_2[\boldsymbol{\Delta}\mathbf{U}_i^q - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^q] \tag{3.41}$$

$$\begin{aligned}\tilde{\mathbf{X}}_i^{q+1} = \tilde{\mathbf{X}}_i^q - \epsilon_3[&-\varepsilon(\bar{\mathbf{X}}_i^q - \tilde{\mathbf{X}}_i^q)\bar{\mathbb{Q}}_i - \boldsymbol{\Gamma}_i^q + \sum_{\substack{j=1\\ j\neq i}}^{N} \hat{\mathbb{A}}_{ji}^T \boldsymbol{\Lambda}_j^q \\ &+ \sum_{\substack{j=1\\ j\neq i}}^{N} (\mathbb{G}_j \mathbb{C}_{jj} \hat{\mathbb{A}}_{jj}^{-1} \hat{\mathbb{A}}_{ji})^T \boldsymbol{\Omega}_{j,min}^q - \sum_{\substack{j=1\\ j\neq i}}^{N} (\mathbb{G}_j \mathbb{C}_{jj} \hat{\mathbb{A}}_{jj}^{-1} \hat{\mathbb{A}}_{ji})^T \boldsymbol{\Omega}_{j,max}^q]\end{aligned} \tag{3.42}$$

$$\begin{aligned}\boldsymbol{\Delta}\tilde{\mathbf{U}}_i^{q+1} = \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^q - \epsilon_4[&-\varepsilon(\boldsymbol{\Delta}\mathbf{U}_i^q - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^q)\bar{\mathbb{R}}_i - \boldsymbol{\Pi}_i^q + \sum_{\substack{j=1\\ j\neq i}}^{N} \hat{\mathbb{B}}_{ji}^T \boldsymbol{\Lambda}_j^q \\ &+ \sum_{\substack{j=1\\ j\neq i}}^{N} (\mathbb{G}_j \mathbb{C}_{jj} \hat{\mathbb{A}}_{jj}^{-1} \hat{\mathbb{B}}_{ji})^T \boldsymbol{\Omega}_{j,min}^q - \sum_{\substack{j=1\\ j\neq i}}^{N} (\mathbb{G}_j \mathbb{C}_{jj} \hat{\mathbb{A}}_{jj}^{-1} \hat{\mathbb{B}}_{ji})^T \boldsymbol{\Omega}_{j,max}^q]\end{aligned} \tag{3.43}$$

and (3.39)

where $i = 1, \ldots, N$*; and* ϵ_1*,* ϵ_2*,* ϵ_3 *and* ϵ_4 *are tuning parameters.*

3.6 Case Study

In order to test the performance of the proposed chance-constrained CDMPC schemes, uncertain disturbances are used in the simulation example Section 2.6. Linearizing the nonlinear model used in Chapter 4, Section 4.3, and discretizing the obtained linear model with a sampling time of $0.005hr$, result in the following discrete-time state-space process model:

$$\mathbf{x}(k+1) = \left[\begin{array}{cc:cc} 1.1357 & 6.1467 & 0.1852 & 0.6696 \\ -0.0014 & 0.7954 & -0.0002 & 0.1466 \\ \hdashline 0.0704 & 0.2556 & 0.9917 & 1.5870 \\ -0.0001 & 0.0557 & -0.0005 & 0.8875 \end{array}\right] \mathbf{x}(k) + \tag{3.44}$$

$$\left[\begin{array}{cc:cc} 0.0000 & 0.0778 & 0.0000 & 0.0113 \\ -0.0000 & 0.0223 & -0.0000 & 0.0039 \\ \hdashline 0.0000 & 0.0022 & 0.0000 & 0.0404 \\ -0.000 & 0.0007 & -0.0000 & 0.0471 \end{array}\right] \mathbf{u}(k) + \tag{3.45}$$

	MPC 1	MPC 2
Initial Conditions	$\mathbf{x}_1(0) = \begin{bmatrix} 5 & -1.5 \end{bmatrix}^T$	$\mathbf{x}_2(0) = \begin{bmatrix} -5 & 1.7 \end{bmatrix}^T$
Weighting Matrices	$\mathbf{Q}_{11} = 5\mathbf{I}$ $\mathbf{R}_{11} = 15\mathbf{I}$	$\mathbf{Q}_{22} = 5\mathbf{I}$ $\mathbf{R}_{22} = 15\mathbf{I}$
Upper Bounds	$\mathbf{y}_1^{max} = \begin{bmatrix} 5 & 1.5 \end{bmatrix}^T$ $\mathbf{u}_1^{max} = \begin{bmatrix} 5 & 4 \end{bmatrix}^T$	$\mathbf{y}_2^{max} = \begin{bmatrix} 5 & 1.5 \end{bmatrix}^T$ $\mathbf{u}_2^{max} = \begin{bmatrix} 5 & 2 \end{bmatrix}^T$
Lower Bounds	$\mathbf{y}_1^{min} = \begin{bmatrix} -15 & -1 \end{bmatrix}^T$ $\mathbf{u}_1^{min} = \begin{bmatrix} -5 & -4 \end{bmatrix}^T$	$\mathbf{y}_2^{min} = \begin{bmatrix} -5 & -1 \end{bmatrix}^T$ $\mathbf{u}_2^{min} = \begin{bmatrix} -5 & -2 \end{bmatrix}^T$
Prediction Horizon	10	10
Control Horizon	5	5

Table 3.1: Parameters used in the probabilistic CDMPC

$$\left[\begin{array}{ccc:ccc} -0.8070 & -0.2263 & 0.0266 & 0.0064 & -0.0174 & 0.0045 \\ 0.0105 & 0.0001 & -0.0000 & -0.0000 & 0.0000 & -0.0000 \\ \hdashline -0.0263 & -0.0073 & 0.0009 & 0.0703 & -0.1912 & 0.0497 \\ 0.0003 & 0.0000 & -0.0000 & 0.0000 & 0.0004 & -0.0000 \end{array}\right] \mathbf{d}(k) \qquad (3.46)$$

It is assumed that the six disturbances affecting the system have normal multivariate distributions.

In each of the two local chance-constrained MPCs, the predicted temperatures and concentrations have to be held within the desired range with the given probability of $\zeta_{i,g,l}^{\mathtt{U}} = 0.8$ and $\zeta_{i,g,l}^{\mathtt{L}} = 0.85$ where $i = 1, 2$, $g = 1, \ldots, 2 \times 10$ and $l = 0, \ldots, 2 \times 9$. Other parameters used in the simulations are given in Table 3.1.

Simulations are performed for the centralized probabilistic MPC and the three proposed chance-constrained CDMPC. In order to compare simulation results of the different schemes, the same stochastic sequence for the uncertain disturbances has been used in all the simulation runs. MATLAB's *quadprog* function with the *active-set* algorithm, has been used to solve the MPC optimization problems. Also, for the coordination methods $\mathfrak{e} = 10^{-6}$ is chosen for the coordinator's stopping criteria. In the simulations using the GC (SALA) method, the penalty parameter was kept constant during the coordinator's iterations ($\kappa = 1$).

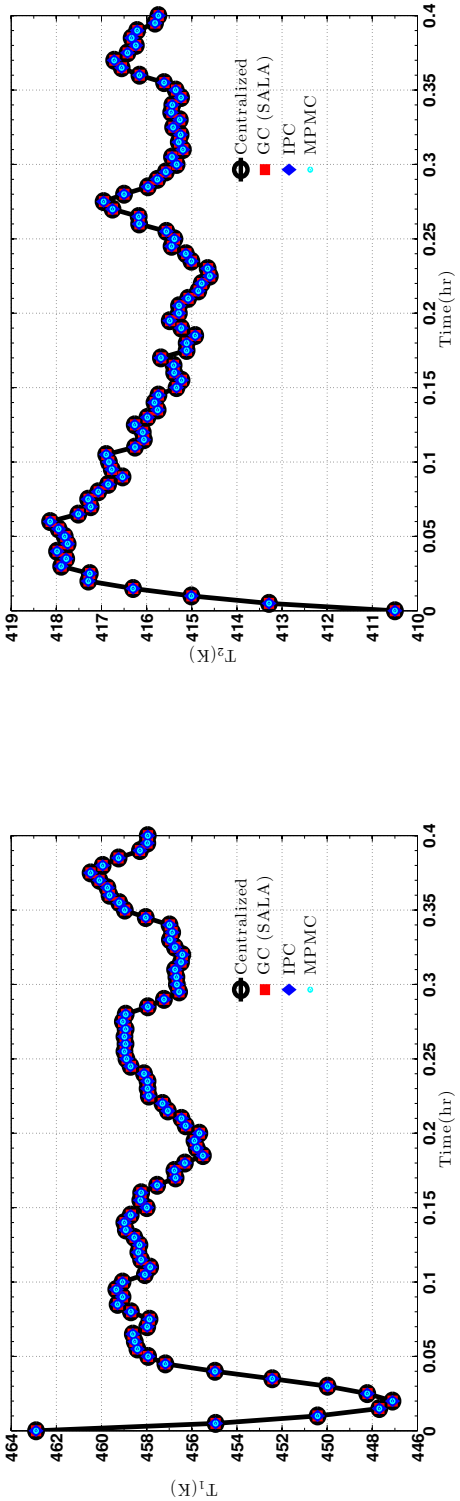
3.6.1 Simulation Results

Figures 3.1 and 3.2 show that process outputs and control inputs trajectories of the local subsystems, resulting from the three chance-constrained CDMPC networks, match the centralized profiles[6]. Therefore, all three CDMPC schemes have successfully achieved the plant-wide MPC performance.

To compare performance of the probabilistic CDMPC networks, in Figure 3.3, number of communication cycles to convergence of coordination algorithm have been plotted. The conspicuous observation that emerged from Figure 3.3 is that in the GC method, the number of communication cycles was considerably larger in comparison with the IPC and MPMC approaches. This is not surprising, as instead of Newton's method, which has quite a fast convergence rate, a first-order gradient-based optimization method (SALA) solves the dual optimization problem[7]. At the expense of increased communication cycles, the coordinator in the GC method converged in the presence of active local inequality constraints. It should be mentioned that for this example, Newton's method failed to converge due to an ill-conditioned Hessian matrix. Also, Figure 3.3 shows that number of required communication cycles in the IPC and MPMC methods, are very close and both are significantly lower than what is needed by the GC (SALA) approach.

[6]The penalty parameter of $\mu = 0.001$ has been used in the simulation using SALA approach.

[7]In Appendix D, simulations results for the deterministic example of chapter 2, are presented where the first-order gradient-based coordinator obtained using SALA, is used in the GC method. Simulations were performed for both unconstrained and constrained CDMPC.



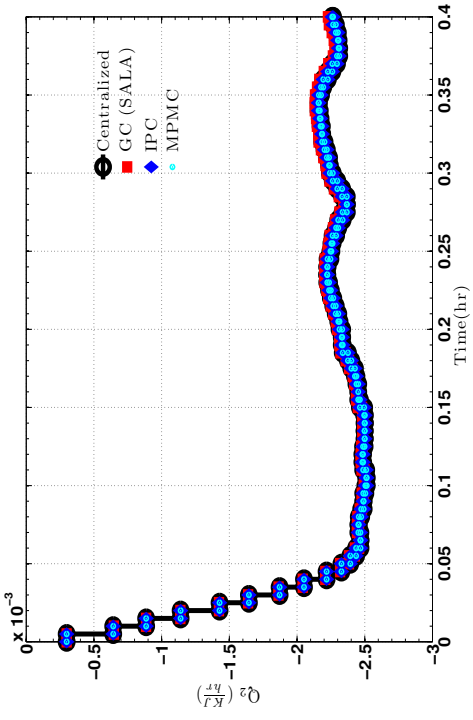
(a) Temperature of unit 1

(b) Temperature of unit 2

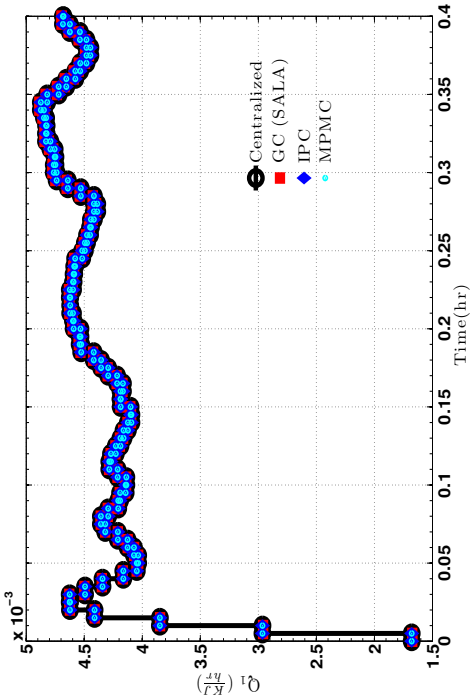
(c) Concentration of unit 1

(d) Concentration of unit 2

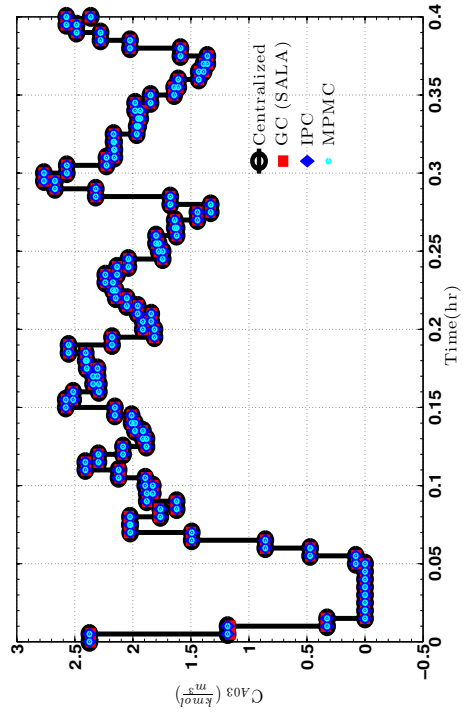
Figure 3.1: Output trajectories resulted from centralized and probabilistic CDMPC



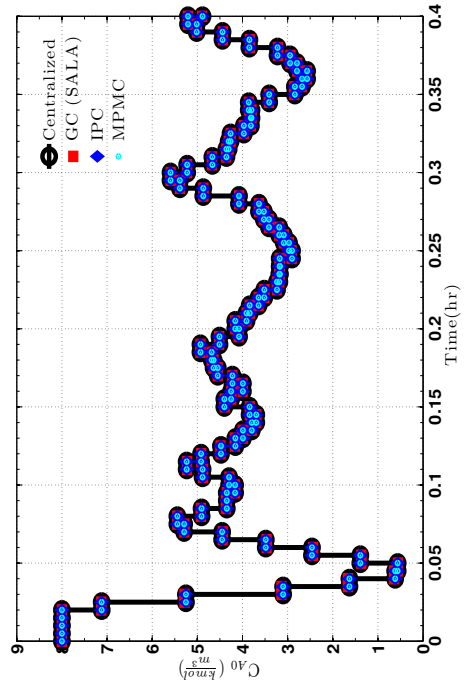
(a) Heat input rate for unit 1



(b) Heat input rate for unit 2



(c) Inlet reactant concentration for unit 1



(d) Inlet reactant concentration for unit 2

Figure 3.2: Control input trajectories resulted from centralized and probabilistic CDMPC

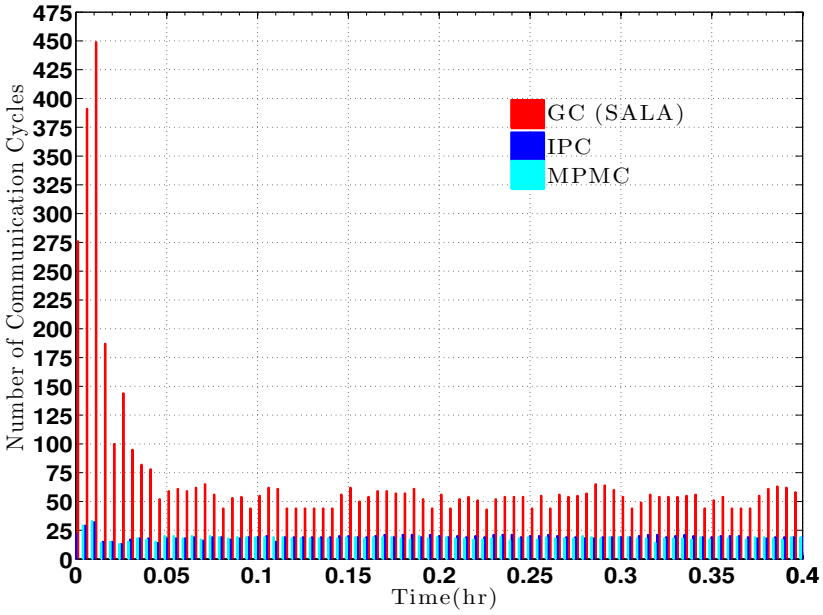


Figure 3.3: Number of communication cycles

As previously discussed, the efficiency of SALA can be improved by finding the most effective penalty parameter μ (Dussault *et al.* (2005)). In order to study the effect of this parameter on the speed of the coordination algorithm, simulations with different values of μ have been performed. Simulation results are shown in Figure 3.4. To avoid repetition, the plots for the process outputs and manipulated variables trajectories have not been shown, but as expected, they are the same as those in Figures 3.1 and 3.2. Figure 3.4 shows that lower values of the penalty parameters result in faster convergence. Therefore, knowing the lowest possible value for this parameter can help improve the coordinator's convergence rate.

3.7 Summary

This chapter attempts to initiate the study of uncertainty impacts on developing CDMPC networks. Among the available MPC methods that explicitly account for the effects of uncertainty, in this chapter, chance-constrained MPC is chosen to address the plant-wide single chance-constrained MPC problem under uncertain disturbances. The GC, IPC and MPMC methods are used to change the existing

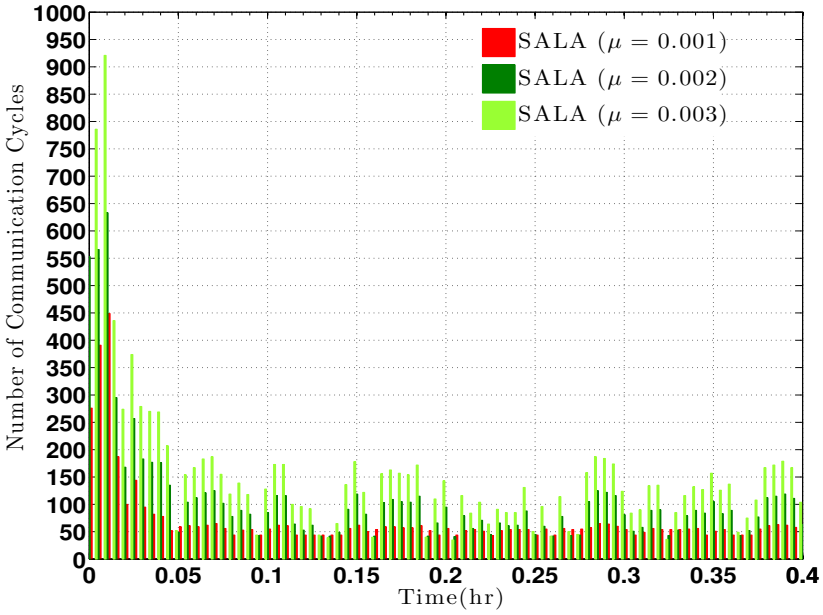


Figure 3.4: Number of communication cycles for different penalty parameter values

decentralized chance-constrained MPC network to probabilistic CDMPC network. Detailed probabilistic CDMPC and coordinator formulations are presented. The use of single chance-constraints in this chapter, provides the advantage of having convex CDMPC. Therefore, the coordination methods are able to guarantee producing optimal plant-wide probabilistic MPC solution.

One of the key features of the probabilistic MPC problem introduced in this chapter, is that in the process models, the expected values of uncertain disturbances are used. Therefore, both local and interaction disturbance terms are known and are taken into account by the local dynamic equations. On the other hand, interaction equations contain unknown interaction terms resulting from predicted states and control input changes of other local processes. The interaction equations are similar to those in the deterministic case. Since the starting point of developing a CDMPC network is relaxing the interaction equations, the same procedures used in applying the coordination methods to deterministic MPC, are employed in the probabilistic case. In the GC method, the update equations in the coordinator remain the same as those in the deterministic MPC, because the effects of local inequalities do not

explicitly appear in the coordinator's equations. On the contrary, the coordinators in the IPC and MPMC methods, explicitly account for the impacts of active local inequalities on the coordination process. In probabilistic CDMPC, probabilistic constraints are parts of local inequalities. Therefore, the final update equations of these coordinators differ from those in the deterministic MPC.

Since inequality constraints are an inherent part of probabilistic MPC, to deal with the convergence issues of the standard gradient-based coordinator in the GC method caused by active local inequalities, an alternative first-order gradient-based optimization method (SALA) is used to solve the resulting dual optimization problem. Using SALA results in a slight modification in the coordinating terms of the CDMPCs and a new formulation for the coordinator. The proposed modified GC-DMPC can be used for both deterministic and stochastic cases. Since a first-order gradient method is used, convergence rate of the coordination algorithm is slower than a Newton-based coordinator.

It should be mentioned that if uncertainties are in the model parameters, still the important step would be to use the available stochastic information in the process models, such that the two sets of local dynamic and interaction equations can be formed. Once the local dynamic and interaction equations are obtained, the standard procedures in the coordination methods can be followed to develop a probabilistic CDMPC network.

If joint chance constraints are used in the MPC formulation, the procedures in applying coordination methods for developing CDMPC networks will remain the same; however, the main challenge would be if the type of multivariate probability distribution used for the probabilistic constraints lead to a non-convex optimization problem, as reaching optimal plant-wide performance cannot be guaranteed by duality-based coordination methods.

Chapter 4

MPMC Method and Plant-Wide Nonlinear MPC Problem

In this chapter, a linear CDMPC scheme[1] is proposed that has the performance of plant-wide nonlinear MPC. The Pseudo-Model Coordination method together with an exact linearization procedure, is employed to create a linear CDMPC network which can be used as a practical alternative to the nonlinear plant-wide constrained MPC. The linear CDMPC network is the result of modifications performed on the existing network of linear decentralized MPCs. Nonlinearities in the local processes, are accounted for in the interaction equations. Therefore, in the process of relaxing the interaction equations, the nonlinearities become part of the coordinator. Performance of the proposed linear CDMPC scheme is investigated using a simulation case study.

Although processes are usually nonlinear, in the vast majority of MPC applications linear models are used to predict the dynamic response of nonlinear processes because: 1) linear model identification techniques are well-studied and easy to use; 2) linear models are able to well forecast dynamics of the process maintained in the neighbourhood of its operating point; and 3) MPC with quadratic objective function and linear constraints is a convex optimization problem that can be solved

[1]In this chapter, the term *linear* in linear CDMPC network is used to emphasize that the distributed controllers are linear.

efficiently using convex optimization methods. Despite these advantages, there are times that linear models are not able to adequately represent the dynamics of the process. For instance, frequent changes in the operating points, the need for having tighter performance specifications and more rigid environmental regulations result in cases where having better control performance requires nonlinear models to improve the quality of the prediction.

Synthesizing nonlinear MPC usually poses two main challenges. The first challenge is the issue of efficiently solving non-convex optimization problems, as nonlinear models may result in a non-convex problem. Unlike convex QP, non-convex optimization is, in general, computationally demanding and much more difficult to solve accurately. On the other hand, on-line implementation of MPC dictates a time limitation, as the nonlinear and often non-convex optimization problem must be solved within each sampling period. Therefore, computationally efficient algorithms for solving nonlinear MPC problems play an important role in making these control problems implementable in practice (Findeisen and Allgower (2002), Cannon (2004), Camacho and Bordons (2007)).

The second challenge in designing nonlinear MPC is stability of the closed-loop system[2]. Even if the solution of finite horizon open-loop nonlinear MPC problem is found, the calculated optimal control inputs could result in an unstable closed-loop system (Rawlings (1999), Findeisen and Allgower (2002), Camacho and Bordons (2007)). In the state-space framework, several methods have been proposed to guarantee stability of the controlled system, either by using a terminal set and/or a

[2]Assuring a closed-loop stable system is equally important in linear MPC design. Generally, one way to achieve a closed-loop stable system in either linear or nonlinear MPC, is to use a long enough prediction horizon H_p; however, increasing H_p increases the size of the optimization problem. Nowadays, because of efficient convex optimization algorithms and powerful computers, H_p can be tuned for stability, in linear(convex) MPC. In non-convex MPC, using a long prediction horizon is usually not practiced, since the resulting increase in the problem size may lead to computationally intractable problem.

The second approach to ensure close-loop stability, would be to find a terminal region with an invariance property, off-line, and then include it the MPC synthesis. In the MPC formulation, the terminal region is usually expressed in the form of a weighted terminal penalty term in the objective function and/or a set of terminal constraints.

In chapter 2, it is assumed that long enough prediction horizon is used; however, a terminal region can also be obtained for the plant-wide benchmark problem and then be incorporated appropriately in developing the CDMPC networks.

terminal cost (Camacho and Bordons (2007), Mayne *et al.* (2000), Chen and Allgower (1998)) or a Lyapunov-based MPC (Mhaskar *et al.* (2006)).

The quasi-infinite horizon nonlinear MPC proposed by Chen and Allgower (1998), is one of the comprehensive nonlinear MPC design approaches where a terminal state penalty term is added to the finite horizon quadratic objective functional and a terminal region constraint to the inequality constraints of the standard nonlinear MPC setup[3]. An off-line procedure was proposed to find the terminal region. Chen and Allgower (1998) proved that the proposed method would guarantee asymptotic stability of the closed-loop system independent of the performance tuning parameters (i.e., $\mathbf{Q}$, $\mathbf{R}$, H_p and H_c) of the standard MPC formulation, if a feasible solution at time $t = 0$ could be found. Another method for nonlinear MPC synthesis was proposed by Mhaskar *et al.* (2006) that guaranteed controller feasibility and closed-loop stability. In this method, to explicitly characterize the stability region, Lyapunov function-based stability constraints, were used in the MPC problem formulation. The stability constraints contained a Lyapunov function and an explicit nonlinear control law, which was obtained off-line.

In the context of plant-wide nonlinear MPC, nonlinear cooperative-based DMPC has been addressed in several research work, e.g., Liu *et al.* (2009), Liu *et al.* (2010*a*), Stewart *et al.* (2011) and Chen *et al.* (2012); however, application of coordination methods to the plant-wide nonlinear MPC problem is an untouched topic. One reason is that developing linear CDMPC networks based on multilevel optimization techniques, is a fairly new research area and thus, the more advanced variations, such as nonlinear MPC problems, are yet to be investigated. Also, coordination methods are based on using duality. In other words, instead of solving the primal optimization (centralized MPC), a relaxed optimization problem (aggregate of CDMPC) that involves estimations of Lagrange multipliers (prices), is solved. Only for convex optimization problems, can it be guaranteed that the relaxed problem can re-produce the solution of the primal problem. If the optimization problem is non-convex, a duality gap may exist and the centralized solution cannot be recovered by the

[3]The original work was presented in the continuous time, but as stated by Findeisen and Allgower (2002), the method is applicable to discrete-time formulations as well.

aggregate of the CDMPC.

In order to address the duality gap issue, the non-convex MPC problem[4] can be converted into an approximate or preferably equivalent convex problem. Coordination methods can then be applied to the approximate convex problem. One way to perform such conversion would be to successively linearize the nonlinear dynamics about trial points. A disadvantage of this approach is that, because the nonlinear dynamics are approximated by linear models, approximation errors exist and the resulting linear problem may not be equivalent to the original nonlinear problem. Another way, would be to find a method that transforms the original problem into an equivalent convex relaxed problem such that when solved, the convex problem produces the solution to the (non-convex) nonlinear problem. The idea of solving an equivalent convex problem to the primal non-convex optimization, is credited to Sorenson and Koble (1984), based on a method proposed by Hassan and Singh (1976).

Hassan and Singh (1976) presented a two-level method for solving optimal control problem of continuous-time nonlinear dynamic systems. The main idea was to use a steady-state point of the system to expand the nonlinear dynamic equations $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t)$ in a Taylor series, in order to write the dynamics in a perturbed form $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{D}(\mathbf{x}, \mathbf{u})$. The $\mathbf{x}$ and $\mathbf{u}$ variables in the perturbed term $\mathbf{D}(\mathbf{x}, \mathbf{u}) = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) - \mathbf{A}\mathbf{x} - \mathbf{B}\mathbf{u}$, were replaced by pseudo-variables. The equations defining the pseudo-variables, were relaxed by quadratic penalty terms. The perturbed term was iteratively compensated for at a second level by numerically solving portions of the optimality conditions, using a fixed-point iteration method. The convergence of the iterative algorithm was proved. In the method proposed by Hassan and Singh (1976), the original nonlinear optimization problem was solved by independent linear quadratic sub-problems along with prediction equations in the second level.

Sorenson and Koble (1984) adapted the above method, to their proposed unifying framework using the MPMC approach. By applying the MPMC method to the discrete-time nonlinear optimal control problem of interconnected systems, Sorenson and Koble (1984) showed that their proposed Problem Manipulation Solution

[4]In this chapter, the term *non-convex MPC* refers to nonlinear MPC, in which the nonlinear prediction model has lead to a non-convex optimization problem.

Strategy (PMSS) approach was not only useful in understanding the coordination methods used in solving linear optimal control problems of interconnected dynamic systems, but also could provide conceptual basis for developing new algorithms. The two important contributions achieved by Sorenson and Koble (1984) were, 1) the resulting two-level problem was *equivalent to* the original nonlinear optimal control problem, since in the MPMC method, in addition to quadratic penalization, Lagrangian relaxation was used in relaxing the pseudo-variable equations. Note that, in the original method by Hassan and Singh (1976), the resulting two-level system was an *approximation of* the original problem; 2) the proposed method was explicitly developed for optimal control of interconnected nonlinear dynamical systems.

The contribution of this chapter is in developing an equivalent linear CDMPC network for plant-wide nonlinear MPC problem. In the context of multilevel optimization-based CDMPC, this is the first time, a linear CDMPC network that achieves optimal plant-wide nonlinear MPC performance, is developed. The MPMC approach for optimal control of interconnected nonlinear systems proposed by Sorenson and Koble (1984), is extended to the plant-wide nonlinear MPC problem. It is assumed that the nonlinear functions in the process models are at least once continuously differentiable. By applying a combination of an exact linearization technique and the MPMC approach, the existing decentralized linear MPC is converted into linear CDMPC. The coordinator is synthesized based on satisfaction of relevant first-order optimality conditions of the aggregate of distributed controllers using numerical methods. In addition to the benefits of CDMPC, the proposed approach provides the significant advantage of solving QP problems in the local controllers. Thus, substantial computational efficiency can be expected, compared with the distributed and decentralized nonlinear MPC that have to solve (non-convex) NLP. This chapters intends to address flexibility, maintainability and computational issues of plant-wide nonlinear MPC problem. Since in the proposed coordinated distributed scheme, linear CDMPCs are used, it is assumed that closed-loop stability can be achieved by properly tuning the prediction horizon.

4.1 Background

In this chapter, the development of CDMPC is presented for the following two types of discrete-time nonlinear process model:

Model I:

The first type of model is assumed to be obtained by discretizing a continuous model. Consider the following continuous-time state-space model, which is available to mathematically represent the nonlinear plant that consists of N interconnected processes:

$$\frac{d\mathbf{x}_i}{dt} = \mathbf{f}_i(\mathbf{x}_i, \mathbf{u}_i, \mathbf{x}_j, \mathbf{u}_j) \tag{4.1a}$$

$$\mathbf{y}_i = \mathbf{C}_{ii}\mathbf{x}_i \tag{4.1b}$$

where $i, j = 1, \ldots, N$ and $j \neq i$. The states $\mathbf{x}_i$, manipulated variables $\mathbf{u}_i$ and outputs $\mathbf{y}_i$ *are not* deviation variables. Euler's method is used to discretize the continuous-time model (4.1a). The resulting discrete-time nonlinear model has the following form:

$$\mathbf{x}_i(k+1) = \mathbf{x}_i(k) + \Delta t\, \mathbf{f}_i(\mathbf{x}_i(k), \mathbf{u}_i(k), \mathbf{x}_j(k), \mathbf{u}_j(k)) \tag{4.2a}$$

$$\mathbf{y}_i(k) = \mathbf{C}_{ii}\mathbf{x}_i(k) \tag{4.2b}$$

where Δt is the sampling time.

Model II:

In the second type of model, it is assumed that the discrete-time nonlinear model is identified directly and no approximation such as Euler's method, is used. This form of nonlinear model is written as:

$$\mathbf{x}_i(k+1) = \mathbf{f}_i(\mathbf{x}_i(k), \mathbf{u}_i(k), \mathbf{x}_j(k), \mathbf{u}_j(k)) \tag{4.3a}$$

$$\mathbf{y}_i(k) = \mathbf{C}_{ii}\mathbf{x}_i(k) \tag{4.3b}$$

where $i, j = 1, \ldots, N$ and $j \neq i$. The states $\mathbf{x}_i$, manipulated variables $\mathbf{u}_i$ and outputs $\mathbf{y}_i$, *are not* deviation variables.

In both **Model I** and **Model II**, it is assumed that all nonlinearity is limited to the dynamics and the state-output relationship is linear.

4.1.1 Exact Linearization of Nonlinear Plant Model

The exact linearization scheme used in Hassan and Singh (1976) and Sorenson and Koble (1984), is essentially re-statement of the discrete-time nonlinear model by two sets of equations. Re-stating **Model I** by two systems of equations can be explained in the following steps:

1. The continuous models (4.1a) are linearized by writing Taylor series expansion around a nominal operating point of the plant to calculate the block-wise matrices $\mathbf{A}^c$ and $\mathbf{B}^c$, where the superscript c stands for *contiunous*.

2. Using the sampling time Δt, the obtained linear continuous model is discretized to obtain the block-wise matrices $\mathbf{A}^d$ and $\mathbf{B}^d$. The following decentralized linear discrete-time model is obtained:

$$\mathbf{x}_i(k+1) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{u}_i(k) \tag{4.4}$$

3. Using the linear discrete-time model (4.4), the discrete-time nonlinear model (4.1a) can be written as:

$$\begin{aligned}\mathbf{x}_i(k+1) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{u}_i(k) \\ + \mathbf{x}_i(k) + \Delta t\, \mathbf{f}_i(\mathbf{x}_i(k), \mathbf{u}_i(k), \mathbf{x}_j(k), \mathbf{u}_j(k)) - \mathbf{A}_{ii}\mathbf{x}_i(k) - \mathbf{B}_{ii}\mathbf{u}_i(k)\end{aligned} \tag{4.5}$$

4. The nonlinear model (4.5) is written in terms of the following two systems of equations:

$$\mathbf{x}_i(k+1) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{u}_i(k) + \mathbf{v}_i(k) \tag{4.6a}$$

$$\mathbf{v}_i(k) = \mathbf{x}_i(k) + \Delta t\, \mathbf{f}_i(\mathbf{x}_i(k), \mathbf{u}_i(k), \mathbf{x}_j(k), \mathbf{u}_j(k)) - \mathbf{A}_{ii}\mathbf{x}_i(k) - \mathbf{B}_{ii}\mathbf{u}_i(k) \tag{4.6b}$$

The following steps convert **Model II** into the desired equivalent system of equations:

1. The discrete-time nonlinear model is linearized by writing Taylor series expansion around a nominal operating point, to obtain:

$$\mathbf{x}_i(k+1) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{u}_i(k) \tag{4.7}$$

where $\mathbf{A}_{ii}$ and $\mathbf{B}_{ii}$ are the first partial derivatives $\nabla_{\mathbf{x}_i}\mathbf{f}_i$ and $\nabla_{\mathbf{u}_i}\mathbf{f}_i$, evaluated at $\mathbf{x}_i^{nominal}$ and $\mathbf{u}_i^{nominal}$, respectively.

2. Using the linear discrete-time model (4.7), the discrete-time nonlinear model (4.3a) can be written as:

$$\begin{aligned}\mathbf{x}_i(k+1) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{u}_i(k) \\ + \mathbf{f}_i(\mathbf{x}_i(k), \mathbf{u}_i(k), \mathbf{x}_j(k), \mathbf{u}_j(k)) - \mathbf{A}_{ii}\mathbf{x}_i(k) - \mathbf{B}_{ii}\mathbf{u}_i(k)\end{aligned} \tag{4.8}$$

3. The system of equations (4.8) can alternatively be written as:

$$\mathbf{x}_i(k+1) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{u}_i(k) + \mathbf{v}_i(k) \tag{4.9a}$$

$$\mathbf{v}_i(k) = \mathbf{f}_i(\mathbf{x}_i(k), \mathbf{u}_i(k), \mathbf{x}_j(k), \mathbf{u}_j(k)) - \mathbf{A}_{ii}\mathbf{x}_i(k) - \mathbf{B}_{ii}\mathbf{u}_i(k) \tag{4.9b}$$

4.1.2 Plant-Wide Nonlinear MPC Problem

Similar to the previous chapters, a benchmark is needed to set the performance goal, which leads to determining coordinating terms in the distributed controllers and designing the coordinator. It is assumed that the nonlinear function $\mathbf{f}_i$, is at least once continuously differentiable. The maximum performance is gained by the following plant-wide nonlinear MPC problem:

$$\min_{\mathbf{X},\boldsymbol{\Delta}\mathbf{U}} \quad \sum_{i=1}^{N}\Big[(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i)^T\,\mathbb{Q}_{ii}\,(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i) + \boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\Big] \tag{4.10a}$$

subject to

$$\begin{aligned}&\mathbf{x}_i(k+l+1|k) = \mathbf{x}_i(k+l|k) + \\ &\Delta t\,\mathbf{f}_i\left(\mathbf{x}_i(k+l|k), \boldsymbol{\Delta}\mathbf{u}_i(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_i(k+l|k), \mathbf{x}_j(k+l|k), \boldsymbol{\Delta}\mathbf{u}_j(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_j(k+l|k)\right)\end{aligned} \tag{4.10b}$$

$$\begin{aligned}&\mathbf{y}_i^{min}(k+l+1) \le \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \le \mathbf{y}_i^{max}(k+l+1) \\ &\mathbf{u}_i^{min}(k+b) \le \mathbf{u}_i(k+b|k) \le \mathbf{u}_i^{max}(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \le \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \le \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b)\end{aligned} \tag{4.10c}$$

where all the variables are in their original, and not deviation, form.

Remark 4.1.1 *In the benchmark problem (4.10), the equality constraints (4.10b) are based on using discrete-time nonlinear model (4.2a). If rather than* ***Model I****, a*

discrete-time nonlinear model is available in the form of ***Model II*** *using (4.3a), the equality constraints (4.10b) are replaced by:*

$$\mathbf{x}_i(k+l+1|k) = \mathbf{f}_i\left(\mathbf{x}_i(k+l|k), \boldsymbol{\Delta}\mathbf{u}_i(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_i(k+l|k), \mathbf{x}_j(k+l|k), \boldsymbol{\Delta}\mathbf{u}_j(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_j(k+l|k)\right) \tag{4.11}$$

4.1.3 Decentralized *Linear* Constrained MPC

It is assumed that the nonlinear plant is currently controlled by the following set of decentralized linear constrained MPC:

$$\min_{\mathbf{X},\boldsymbol{\Delta}\mathbf{U}} \quad \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i \tag{4.12a}$$

subject to

$$\mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}^d \mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}^d \left[\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] \tag{4.12b}$$

$$\begin{aligned} &\mathbf{y}_i^{min}(k+l+1) \le \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \le \mathbf{y}_i^{max}(k+l+1) \\ &\mathbf{u}_i^{min}(k+b) \le \mathbf{u}_i(k+b|k) \le \mathbf{u}_i^{max}(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \le \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \le \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b) \end{aligned} \tag{4.12c}$$

The above decentralized MPC could result in serious performance degradation because, not only the interaction effects have not been taken into account, but also the linear models may not be able to give satisfactory prediction of the dynamic behaviour. The CDMPC network developed in the subsequent sections, will 1) account for the interaction effects; and 2) exactly predict the nonlinear dynamics while using linear models in the CDMPC.

4.2 Plant-Wide Nonlinear MPC via MPMC-DMPC

4.2.1 Exact Linearization of Local Prediction Models

In this work, for designing CDMPC networks, it is required that nonlinear discrete-time prediction model be available in the form of either **Model I** or **Model II**. The

existing decentralized linear MPC (4.12), use linear discrete-time prediction model (4.12b). The goal is to re-state the prediction model in the plant-wide problem in terms of two systems of equations, as discussed in section 4.1.1. Using (4.12b), the prediction model (4.10b) can be re-written as:

$$
\begin{aligned}
&\mathbf{x}_i(k+l+1|k) = \\
&\mathbf{A}^d_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}^d_{ii}\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{B}^d_{ii}\mathbf{u}_i(k-1) \\
&+ \mathbf{x}_i(k+l|k) \\
&+ \Delta t\,\mathbf{f}_i\left(\mathbf{x}_i(k+l|k), \boldsymbol{\Delta}\mathbf{u}_i(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_i(k+l|k), \mathbf{x}_j(k+l|k), \boldsymbol{\Delta}\mathbf{u}_j(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_j(k+l|k)\right) \\
&- \mathbf{A}^d_{ii}\mathbf{x}_i(k+l|k) - \mathbf{B}^d_{ii}\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) - \mathbf{B}^d_{ii}\mathbf{u}_i(k-1)
\end{aligned}
\tag{4.13}
$$

The system of equations (4.13), can alternatively be expressed by the following two sets of equations:

$$
\mathbf{x}_i(k+l+1|k) = \mathbf{A}^d_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}^d_{ii}\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{v}_i(k+l|k) \tag{4.14a}
$$

$$
\begin{aligned}
&\mathbf{v}_i(k+l|k) = \mathbf{x}_i(k+l|k) \\
&+ \Delta t\,\mathbf{f}_i\left(\mathbf{x}_i(k+l|k), \boldsymbol{\Delta}\mathbf{u}_i(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_i(k+l|k), \mathbf{x}_j(k+l|k), \boldsymbol{\Delta}\mathbf{u}_j(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_j(k+l|k)\right) \\
&\qquad - \mathbf{A}^d_{ii}\mathbf{x}_i(k+l|k) - \mathbf{B}^d_{ii}\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k)
\end{aligned}
\tag{4.14b}
$$

where all the variables are still in their original form. It should be noted that, $\mathbf{B}^d_{ii}$ and $\mathbf{u}_i(k-1)$ are known. Thus, the term $\mathbf{B}^d_{ii}\mathbf{u}_i(k-1)$ in (4.12b), is a *known term* and it is cancelled by its negative form in (4.13).

Remark 4.2.1 *In order to be consistent with the terminology used throughout this thesis, the set of equations (4.14b) shall still be referred to as* ***interaction equations****; however, unlike the linear case, where the system of interaction equations only contains the interaction terms, here these equations contain local linear and the complete local nonlinear models. The effects of interactions are taken into account by nonlinear portion of the interaction equations.*

Remark 4.2.2 *If the discrete-time prediction model (4.11) is available, following the same manipulations and re-arrangements, it is equivalently expressed by the following sets of equations:*

$$\mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}^d \mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}^d \sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{v}_i(k+l|k) \tag{4.15a}$$

$$\begin{aligned}&\mathbf{v}_i(k+l|k) = \\ &\mathbf{f}_i\left(\mathbf{x}_i(k+l|k), \boldsymbol{\Delta}\mathbf{u}_i(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_i(k+l|k), \mathbf{x}_j(k+l|k), \boldsymbol{\Delta}\mathbf{u}_j(k|k), \ldots, \boldsymbol{\Delta}\mathbf{u}_j(k+l|k)\right) \\ &\qquad - \mathbf{A}_{ii}^d \mathbf{x}_i(k+l|k) - \mathbf{B}_{ii}^d \sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k)\end{aligned} \tag{4.15b}$$

4.2.2 Pseudo-Variables and Exact Linearized Prediction Models

As was discussed in the previous chapters, the key idea in the MPMC method is the use of pseudo-variables for the unknown interacting variables. For the MPC problem, the pseudo-variables are defined as:

$$\tilde{\mathbf{x}}_i(k+l|k) \triangleq \mathbf{x}_i(k+l|k) \tag{4.16a}$$

$$\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+b|k) \triangleq \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \tag{4.16b}$$

where $l = 1, \ldots, H_p - 1$ and $b = 0, \ldots, H_u - 1$. The interaction variables are replaced by the above pseudo-variables. It is desired that the value of $\tilde{\mathbf{x}}_i(k+l|k)$ and $\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+b|k)$ be equal to the value of $\mathbf{x}_i(k+l|k)$ and $\boldsymbol{\Delta}\mathbf{u}_i(k+b|k)$, respectively, which implies:

$$\tilde{\mathbf{x}}_i(k+l|k) = \mathbf{x}_i(k+l|k) \tag{4.17a}$$

$$\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+b|k) = \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \tag{4.17b}$$

where equations (4.17) are referred to as the transformed interaction equations and are included as equality constraints in the benchmark centralized MPC problem.

In MPC with linear dynamics, as in sections 2.5 and 3.5, the variables on the right-hand side of the interaction equations (i.e., the interaction variables) are replaced by the pseudo-variables. Similarly, here, the variables on the right-hand side of the

interaction equations (i.e., both local and interaction variables), are replaced by the pseudo-variables, which yields:

$$\begin{aligned}\mathbf{v}_i(k+l|k) &= \tilde{\mathbf{x}}_i(k+l|k)\\ &+ \Delta t\, \mathbf{f}_i(\tilde{\mathbf{x}}_i(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+l|k), \tilde{\mathbf{x}}_j(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+l|k))\\ &\quad - \mathbf{A}_{ii}^d \tilde{\mathbf{x}}_i(k+l|k) - \mathbf{B}_{ii}^d \sum_{a=0}^{l} \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+a|k)\end{aligned} \tag{4.18}$$

Now by replacing local process model (4.12b), by equations (4.14a) and (4.18), and including the transformed interaction equations (4.17) in the equality constraints, the first series of required modifications for converting the decentralized MPC to CDMPC, are performed. These adjustments transform the decentralized linear MPC problem (4.12) into the following nonlinear MPC problem:

$$\min_{\mathbf{X},\boldsymbol{\Delta}\mathbf{U}} \quad (\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i)^T \mathbb{Q}_{ii} (\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i \tag{4.19a}$$

subject to

$$\mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}^d \mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}^d \sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{v}_i(k+l|k) \tag{4.19b}$$

$$\begin{aligned}\mathbf{v}_i(k+l|k) &= \tilde{\mathbf{x}}_i(k+l|k)\\ &+ \Delta t\, \mathbf{f}_i(\tilde{\mathbf{x}}_i(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+l|k), \tilde{\mathbf{x}}_j(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+l|k))\\ &\quad - \mathbf{A}_{ii}^d \tilde{\mathbf{x}}_i(k+l|k) - \mathbf{B}_{ii}^d \sum_{a=0}^{l} \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+a|k)\end{aligned} \tag{4.19c}$$

$$\tilde{\mathbf{x}}_i(k+l|k) = \mathbf{x}_i(k+l|k) \tag{4.19d}$$

$$\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+b|k) = \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \tag{4.19e}$$

$$\begin{aligned}&\mathbf{y}_i^{min}(k+l+1) \le \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \le \mathbf{y}_i^{max}(k+l+1)\\ &\mathbf{u}_i^{min}(k+b) \le \mathbf{u}_i(k+b|k) \le \mathbf{u}_i^{max}(k+b)\\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \le \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \le \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b)\end{aligned} \tag{4.19f}$$

where the aggregate of (4.19) is equivalent to the plant-wide nonlinear MPC benchmark problem (4.10), in which the equality constraints have been replaced by (4.14a) and (4.18).

Remark 4.2.3 *If the discrete-time prediction model (4.15) is available, the local process model (4.12b) will be replaced by (4.15a) and the following form of (4.15b):*

$$\begin{aligned}&\mathbf{v}_i(k+l|k) = \\ &\mathbf{f}_i(\tilde{\mathbf{x}}_i(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+l|k), \tilde{\mathbf{x}}_j(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+l|k)) \\ &\qquad - \mathbf{A}^d_{ii}\tilde{\mathbf{x}}_i(k+l|k) - \mathbf{B}^d_{ii}\sum_{a=0}^{l}\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+a|k)\end{aligned} \tag{4.20}$$

4.2.3 Distributed Controllers

Following the same procedure and discussion as in section 2.5, it is possible to perform the second series of modifications by applying the MPMC method to problem (4.19). Ultimately, the following formulation is obtained for the CDMPC:

$$\begin{aligned}\min_{\mathbf{X}_i, \boldsymbol{\Delta}\mathbf{U}_i} \quad & \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right)^T \mathbb{Q}_{ii}\left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i\right) + \boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i + \\ & \frac{\varepsilon}{2}\left(||\mathbf{X}_i - \tilde{\mathbf{X}}_i||^2_{\bar{\mathbb{Q}}_i} + ||\boldsymbol{\Delta}\mathbf{U}_i - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\bar{\mathbb{R}}_i}\right) + \\ & \boldsymbol{\Gamma}_i^T(\mathbf{X}_i - \tilde{\mathbf{X}}_i) + \boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i)\end{aligned} \tag{4.21a}$$

subject to

$$\mathbf{x}_i(k+l+1|k) = \mathbf{A}^d_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}^d_{ii}\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{v}_i(k+l|k) \tag{4.21b}$$

$$\begin{aligned}&\mathbf{y}_i^{min}(k+l+1) \le \mathbf{C}_{ii}\mathbf{x}_i(k+l+1|k) \le \mathbf{y}_i^{max}(k+l+1) \\ &\mathbf{u}_i^{min}(k+b) \le \mathbf{u}_i(k+b|k) \le \mathbf{u}_i^{max}(k+b) \\ &\boldsymbol{\Delta}\mathbf{u}_i^{min}(k+b) \le \boldsymbol{\Delta}\mathbf{u}_i(k+b|k) \le \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+b)\end{aligned} \tag{4.21c}$$

where the prices $\boldsymbol{\Gamma}_i$ and $\boldsymbol{\Pi}_i$, the pseudo-variables $\tilde{\mathbf{X}}_i$ and $\boldsymbol{\Delta}\tilde{\mathbf{U}}_i$ and the variables $\mathbf{v}_i$ which contain predicted values of nonlinear and linear terms, are coordinating variables. At each communication cycle, $\mathbf{v}_i$ is fixed by the coordinator. Consequently, all constraints in the local controllers are linear. This is an important feature of the proposed CDMPC, as with linear constraints and quadratic objective function, they become QP problems. It should be noted that, in (4.19), using interaction equations (4.20), instead of (4.19c), will result in the same CDMPC, because local dynamic equations (4.19b) remain the same; however, coordinator's equations will be different.

4.2.4 Coordinator Design

The coordinator is designed by numerically solving portions of the first-order optimality conditions resulting form the aggregate of CDMPC (4.21). Before proceeding further, in order to simplify coordinator design, the aggregate of CDMPC (4.21) and the interaction equations (4.19c) are written in the following compact form:

$$\begin{aligned}\min_{\mathbf{X}_i,\boldsymbol{\Delta}\mathbf{U}_i} \quad & \sum_{i=1}^{N}\Big((\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)^T\,\mathbb{Q}_{ii}\,(\mathbb{C}_{ii}\mathbf{X}_i-\mathbf{r}_i)+\boldsymbol{\Delta}\mathbf{U}_i^T\mathbb{R}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\Big)+\\ & \sum_{i=1}^{N}\frac{\varepsilon}{2}\Big(||\mathbf{X}_i-\tilde{\mathbf{X}}_i||^2_{\mathbb{Q}_i}+||\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\mathbb{R}_i}\Big)+ \qquad (4.22a)\\ & \sum_{i=1}^{N}\Big(\boldsymbol{\Gamma}_i^T(\mathbf{X}_i-\tilde{\mathbf{X}}_i)+\boldsymbol{\Pi}_i^T(\boldsymbol{\Delta}\mathbf{U}_i-\boldsymbol{\Delta}\tilde{\mathbf{U}}_i)\Big)\end{aligned}$$

subject to

$$\hat{\mathbb{A}}_{ii}^d\mathbf{X}_i+\hat{\mathbb{B}}_{ii}^d\boldsymbol{\Delta}\mathbf{U}_i+\tilde{\mathbb{A}}_{ii}^d\tilde{\mathbf{X}}_i-\hat{\mathbb{B}}_{ii}^d\boldsymbol{\Delta}\tilde{\mathbf{U}}_i-\Delta t\mathbf{F}_i=\mathcal{Z}_i \qquad (4.22b)$$

$$\begin{aligned}&\mathbf{Y}_i^{min}\le\mathbb{C}_{ii}\mathbf{X}_i\le\mathbf{Y}_i^{max}\\ &\mathfrak{U}_i^{min}\le\mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i\le\mathfrak{U}_i^{max}\\ &i=1,\ldots,N\end{aligned} \qquad (4.22c)$$

where details on obtaining equality constraints (4.22b) and inequality constraints (4.22c) have been provided in Appendices E and B.3.2, respectively.

Remark 4.2.4 *If the aggregate of CDMPC (4.21) and the interaction equations (4.20) are used, instead of (4.22b), the following equations will be used in the compact form (4.22):*

$$\hat{\mathbb{A}}_{ii}^d\mathbf{X}_i+\hat{\mathbb{B}}_{ii}^d\boldsymbol{\Delta}\mathbf{U}_i+\tilde{\mathbb{A}}_{ii}^d\tilde{\mathbf{X}}_i-\hat{\mathbb{B}}_{ii}^d\boldsymbol{\Delta}\tilde{\mathbf{U}}_i-\mathbf{F}_i=\mathbf{0} \qquad (4.23)$$

where all the matrices and vectors, except for the matrix $\tilde{\mathbb{A}}_{ii}^d$ and vector $\mathcal{Z}_i$, are the same as those in (4.22b). More details on obtaining (4.23), can be found in Appendix E, section E.2.

To obtain the optimality conditions the Lagrangian for the optimization problem (4.22) is formed as:

$$
\begin{aligned}
&L(\mathbf{X}, \boldsymbol{\Delta}\mathbf{U}, \tilde{\mathbf{X}}, \boldsymbol{\Delta}\tilde{\mathbf{U}}, \boldsymbol{\Lambda}, \boldsymbol{\Omega}, \boldsymbol{\Upsilon}, \boldsymbol{\Gamma}, \boldsymbol{\Pi}) = \sum_{i=1}^{N} L_i(\mathbf{X}_i, \boldsymbol{\Delta}\mathbf{U}_i, \tilde{\mathbf{X}}_i, \boldsymbol{\Delta}\tilde{\mathbf{U}}_i, \boldsymbol{\Lambda}_i, \boldsymbol{\Omega}_i, \boldsymbol{\Upsilon}_i, \boldsymbol{\Gamma}_i, \boldsymbol{\Pi}_i) \\
&= \sum_{i=1}^{N} \Big((\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i)^T \mathbb{Q}_{ii} (\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{r}_i) + \boldsymbol{\Delta}\mathbf{U}_i^T \mathbb{R}_{ii} \boldsymbol{\Delta}\mathbf{U}_i \Big) \\
&+ \sum_{i=1}^{N} \frac{\varepsilon}{2} \Big(||\mathbf{X}_i - \tilde{\mathbf{X}}_i||^2_{\mathbb{Q}_i} + ||\boldsymbol{\Delta}\mathbf{U}_i - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i||^2_{\mathbb{R}_i} \Big) \\
&+ \sum_{i=1}^{N} \Big(\boldsymbol{\Gamma}_i^T (\mathbf{X}_i - \tilde{\mathbf{X}}_i) + \boldsymbol{\Pi}_i^T (\boldsymbol{\Delta}\mathbf{U}_i - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i) \Big) \\
&+ \sum_{i=1}^{N} \boldsymbol{\Lambda}_i^T \Big(\hat{\mathbb{A}}_{ii}^d \mathbf{X}_i + \hat{\mathbb{B}}_{ii}^d \boldsymbol{\Delta}\mathbf{U}_i + \tilde{\mathbb{A}}_{ii}^d \tilde{\mathbf{X}}_i - \hat{\mathbb{B}}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{U}}_i - \Delta t \mathbf{F}_i - \mathcal{Z}_i \Big) \\
&+ \sum_{i=1}^{N} \boldsymbol{\Omega}_{i,min}^T \left(\mathbf{Y}_i^{min} - \mathbb{C}_{ii}\mathbf{X}_i \right) + \boldsymbol{\Omega}_{i,max}^T \left(\mathbb{C}_{ii}\mathbf{X}_i - \mathbf{Y}_i^{max} \right) \\
&+ \sum_{i=1}^{N} \boldsymbol{\Upsilon}_{i,min}^T \left(\mathfrak{U}_i^{min} - \mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i \right) + \boldsymbol{\Upsilon}_{i,max}^T \left(\mathbb{U}_{ii}\boldsymbol{\Delta}\mathbf{U}_i - \mathfrak{U}_i^{max} \right)
\end{aligned}
\tag{4.24}
$$

The optimality conditions[5] are obtained by taking derivatives of the Lagrangian (4.24) with respect to the decision variables and Lagrange multipliers, as below:

$$
\begin{aligned}
&\nabla_\eta L(\mathbf{X}, \boldsymbol{\Delta}\mathbf{U}, \tilde{\mathbf{X}}, \boldsymbol{\Delta}\tilde{\mathbf{U}}, \boldsymbol{\Lambda}, \boldsymbol{\Omega}, \boldsymbol{\Upsilon}, \boldsymbol{\Gamma}, \boldsymbol{\Pi}) = \mathbf{0} \\
&\eta = \{\mathbf{X}, \boldsymbol{\Delta}\mathbf{U}, \tilde{\mathbf{X}}, \boldsymbol{\Delta}\tilde{\mathbf{U}}, \boldsymbol{\Lambda}, \boldsymbol{\Gamma}, \boldsymbol{\Pi}\}
\end{aligned}
\tag{4.25}
$$

The CDMPCs (4.21), are responsible for satisfaction of the optimality conditions resulting from when $\eta = \{\mathbf{X}_i, \boldsymbol{\Delta}\mathbf{U}_i, \boldsymbol{\Lambda}_i\}$ (for $i = 1, \ldots, N$) along with the feasibility, complementary slackness and non-negativity conditions, resulting from local inequality constraints. The coordinator can use fixed-point iteration to numerically solve the remaining conditions i.e, $\nabla_{\tilde{\mathbf{X}}} L = \mathbf{0}$, $\nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}} L = \mathbf{0}$, $\nabla_{\boldsymbol{\Gamma}} L = \mathbf{0}$ and $\nabla_{\boldsymbol{\Pi}} L = \mathbf{0}$. The update equations for the pseudo-variables $\tilde{\mathbf{X}}_i$ and $\boldsymbol{\Delta}\tilde{\mathbf{U}}_i$, are the same as those in the previous chapters, and are obtained as follows:

[5]More explanation can be found in the section 2.5.2.

$$\begin{cases} \nabla_{\mathbf{\Gamma}} L = \sum_{i=1}^{N} \nabla_{\mathbf{\Gamma}_i} L = \mathbf{0} \\ \nabla_{\mathbf{\Pi}} L = \sum_{i=1}^{N} \nabla_{\mathbf{\Pi}_i} L = \mathbf{0} \end{cases} \tag{4.26a}$$

$$\Longrightarrow$$

for i=1,. . . ,N :

$$\begin{cases} \mathbf{X}_i^q - \tilde{\mathbf{X}}_i^{q+1} = \mathbf{0} \\ \mathbf{\Delta U}_i^q - \mathbf{\Delta}\tilde{\mathbf{U}}_i^{q+1} = \mathbf{0} \end{cases} \tag{4.26b}$$

$$\Longrightarrow$$

for i=1,. . . ,N :

$$\begin{cases} \tilde{\mathbf{X}}_i^{q+1} = \mathbf{X}_i^q \\ \mathbf{\Delta}\tilde{\mathbf{U}}_i^{q+1} = \mathbf{\Delta U}_i^q \end{cases} \tag{4.26c}$$

Similar to the previous chapters, the update equations for the prices, are obtained by numerically solving $\nabla_{\tilde{\mathbf{X}}} L = \mathbf{0}$ and $\nabla_{\mathbf{\Delta}\tilde{\mathbf{U}}} L = \mathbf{0}$. The key difference is that in the previous chapters, optimality conditions result in linear algebraic equations. But, in this chapter, because of the vector $\mathbf{F}_i$ which contains nonlinear functions, the optimality conditions may lead to nonlinear algebraic equations. Therefore, usually a system of nonlinear algebraic equations will be solved using numerical methods. In the case of using a fixed-point iteration method, the update equations are obtained as follows:

$$\begin{cases} \nabla_{\tilde{\mathbf{X}}} L = \sum_{i=1}^{N} \nabla_{\tilde{\mathbf{X}}_i} L = \mathbf{0} \\ \nabla_{\mathbf{\Delta}\tilde{\mathbf{U}}} L = \sum_{i=1}^{N} \nabla_{\mathbf{\Delta}\tilde{\mathbf{U}}_i} L = \mathbf{0} \end{cases} \tag{4.27a}$$

Therefore, for $i = 1, \ldots, N,$ the following update equations are achieved:

$$
\begin{cases}
-\varepsilon(\mathbf{X}_i^q - \tilde{\mathbf{X}}_i^{q+1})\bar{\mathbb{Q}}_i - \boldsymbol{\Gamma}_i^{q+1} + \left(\tilde{\mathbb{A}}_{ii}^d\right)^T \boldsymbol{\Lambda}_i^q - \Delta t \sum_{j=1}^{N} \left(\nabla_{\tilde{\mathbf{X}}_i} \mathbf{F}_j \Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}} \right)^T \boldsymbol{\Lambda}_j^q \\
\quad + \left(\mathbb{C}_{ii} \left[\hat{\mathbb{A}}_{ii}^d\right]^{-1} \tilde{\mathbb{A}}_{ii}^d \right)^T \boldsymbol{\Omega}_{i,min}^q - \Delta t \sum_{j=1}^{N} \left(\mathbb{C}_{jj} \left[\hat{\mathbb{A}}_{jj}^d\right]^{-1} \nabla_{\tilde{\mathbf{X}}_i} \mathbf{F}_j \Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}} \right)^T \boldsymbol{\Omega}_{j,min}^q \\
\quad - \left(\mathbb{C}_{ii} \left[\hat{\mathbb{A}}_{ii}^d\right]^{-1} \tilde{\mathbb{A}}_{ii}^d \right)^T \boldsymbol{\Omega}_{i,max}^q + \Delta t \sum_{j=1}^{N} \left(\mathbb{C}_{jj} \left[\hat{\mathbb{A}}_{jj}^d\right]^{-1} \nabla_{\tilde{\mathbf{X}}_i} \mathbf{F}_j \Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}} \right)^T \boldsymbol{\Omega}_{j,max}^q = \mathbf{0} \\
-\varepsilon(\boldsymbol{\Delta}\mathbf{U}_i^q - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^{q+1})\bar{\mathbb{R}}_i - \boldsymbol{\Pi}_i^{q+1} - \left(\hat{\mathbb{B}}_{ii}^d\right)^T \boldsymbol{\Lambda}_i^q - \Delta t \sum_{j=1}^{N} \left(\nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_i} \mathbf{F}_j \Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}} \right)^T \boldsymbol{\Lambda}_j^q \\
\quad - \left(\mathbb{C}_{ii} \left[\hat{\mathbb{A}}_{ii}^d\right]^{-1} \hat{\mathbb{B}}_{ii}^d \right)^T \boldsymbol{\Omega}_{i,min}^q - \Delta t \sum_{j=1}^{N} \left(\mathbb{C}_{jj} \left[\hat{\mathbb{A}}_{jj}^d\right]^{-1} \nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_i} \mathbf{F}_j \Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}} \right)^T \boldsymbol{\Omega}_{j,min}^q \\
\quad + \left(\mathbb{C}_{ii} \left[\hat{\mathbb{A}}_{ii}^d\right]^{-1} \hat{\mathbb{B}}_{ii}^d \right)^T \boldsymbol{\Omega}_{i,max}^q + \Delta t \sum_{j=1}^{N} \left(\mathbb{C}_{jj} \left[\hat{\mathbb{A}}_{jj}^d\right]^{-1} \nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_i} \mathbf{F}_j \Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}} \right)^T \boldsymbol{\Omega}_{j,max}^q = \mathbf{0}
\end{cases}
\tag{4.27b}
$$

The derivative terms in (4.27b) can be replaced by the following matrices:

$$
\nabla_{\tilde{\mathbf{X}}_i} \mathbf{F}_j \Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}} = \mathbb{A}_{ji}^{c,q+1} \tag{4.28a}
$$

$$
\nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_i} \mathbf{F}_j \Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}} = \mathbb{B}_{ji}^{c,q+1} \tag{4.28b}
$$

where details on obtaining $\mathbb{A}_{ji}^c$ and $\mathbb{B}_{ji}^c$, have been provided in Appendix E.3. After replacing the derivative terms in (4.27b) by the relevant matrices in (4.28), the following re-arranged update equations are obtained for $i = 1, \ldots, N$:

$$
\begin{cases}
\boldsymbol{\Gamma}_i^{q+1} = -\varepsilon(\mathbf{X}_i^q - \tilde{\mathbf{X}}_i^{q+1})\bar{\mathbb{Q}}_i + \left(\tilde{\mathbb{A}}_{ii}^d\right)^T \boldsymbol{\Lambda}_i^q - \Delta t \left(\mathbb{A}_{ii}^{c,q+1}\right)^T \boldsymbol{\Lambda}_i^q - \Delta t \sum_{\substack{j=1 \\ j\neq i}}^{N} \left(\mathbb{A}_{ji}^{c,q+1}\right)^T \boldsymbol{\Lambda}_j^q \\
\qquad + \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\tilde{\mathbb{A}}_{ii}^d\right)^T \boldsymbol{\Omega}_{i,min}^q - \Delta t \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\mathbb{A}_{ii}^{c,q+1}\right)^T \boldsymbol{\Omega}_{i,min}^q \\
\qquad\qquad -\Delta t \sum_{\substack{j=1 \\ j\neq i}}^{N} \left(\mathbb{C}_{jj}\left[\hat{\mathbb{A}}_{jj}^d\right]^{-1}\mathbb{A}_{ji}^{c,q+1}\right)^T \boldsymbol{\Omega}_{j,min}^q \\
\qquad - \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\tilde{\mathbb{A}}_{ii}^d\right)^T \boldsymbol{\Omega}_{i,max}^q + \Delta t \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\mathbb{A}_{ii}^{c,q+1}\right)^T \boldsymbol{\Omega}_{i,max}^q \\
\qquad\qquad +\Delta t \sum_{\substack{j=1 \\ j\neq i}}^{N} \left(\mathbb{C}_{jj}\left[\hat{\mathbb{A}}_{jj}^d\right]^{-1}\mathbb{A}_{ji}^{c,q+1}\right)^T \boldsymbol{\Omega}_{j,max}^q \\
\boldsymbol{\Pi}_i^{q+1} = -\varepsilon(\boldsymbol{\Delta}\mathbf{U}_i^q - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^{q+1})\bar{\mathbb{R}}_i - \left(\hat{\mathbb{B}}_{ii}^d\right)^T \boldsymbol{\Lambda}_i^q - \Delta t \left(\mathbb{B}_{ii}^{c,q+1}\right)^T \boldsymbol{\Lambda}_i^q - \Delta t \sum_{\substack{j=1 \\ j\neq i}}^{N} \left(\mathbb{B}_{ji}^{c,q+1}\right)^T \boldsymbol{\Lambda}_j^q \\
\qquad - \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\hat{\mathbb{B}}_{ii}^d\right)^T \boldsymbol{\Omega}_{i,min}^q - \Delta t \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\mathbb{B}_{ii}^{c,q+1}\right)^T \boldsymbol{\Omega}_{i,min}^q \\
\qquad\qquad -\Delta t \sum_{\substack{j=1 \\ j\neq i}}^{N} \left(\mathbb{C}_{jj}\left[\hat{\mathbb{A}}_{jj}^d\right]^{-1}\mathbb{B}_{ji}^{c,q+1}\right)^T \boldsymbol{\Omega}_{j,min}^q \\
\qquad + \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\hat{\mathbb{B}}_{ii}^d\right)^T \boldsymbol{\Omega}_{i,max}^q + \Delta t \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\mathbb{B}_{ii}^{c,q+1}\right)^T \boldsymbol{\Omega}_{i,max}^q \\
\qquad\qquad +\Delta t \sum_{\substack{j=1 \\ j\neq i}}^{N} \left(\mathbb{C}_{jj}\left[\hat{\mathbb{A}}_{jj}^d\right]^{-1}\mathbb{B}_{ji}^{c,q+1}\right)^T \boldsymbol{\Omega}_{j,max}^q
\end{cases}
\tag{4.29}
$$

where in order to show how the effects of interactions are taken into account, the summation terms have been broken into two pieces, one for $j = i$ and one for $j \neq i$.

Thus far, the update equations have been obtained for all the coordinating variables except for $\mathbf{v}_i$. Since in writing the compact form (4.22), the predicted interaction terms $\mathbf{v}_i$, are replaced by (4.18), these variables are no longer explicitly present in the Lagrangian (4.24). Consequently, they have not been listed among the variables that are used to obtain the optimality conditions. Using updated pseudo-variables,

the vector $\mathbf{v}_i$, is calculated by (4.18):

$$\begin{aligned}\mathbf{v}_i^{q+1}(k+l|k) &= \tilde{\mathbf{x}}_i^{q+1}(k+l|k)+ \\ \Delta t\, \mathbf{f}_i(&\tilde{\mathbf{x}}_i^{q+1}(k+l|k),\boldsymbol{\Delta}\tilde{\mathbf{u}}_i^{q+1}(k|k),\ldots,\boldsymbol{\Delta}\tilde{\mathbf{u}}_i^{q+1}(k+l|k),\tilde{\mathbf{x}}_j^{q+1}(k+l|k),\boldsymbol{\Delta}\tilde{\mathbf{u}}_j^{q+1}(k|k),\ldots,\boldsymbol{\Delta}\tilde{\mathbf{u}}_j^{q+1}(k+l|k)) \\ &- \mathbf{A}_{ii}^d\tilde{\mathbf{x}}_i^{q+1}(k+l|k) - \mathbf{B}_{ii}^d\sum_{a=0}^{l}\boldsymbol{\Delta}\tilde{\mathbf{u}}_i^{q+1}(k+a|k)\end{aligned} \tag{4.30}$$

The sets of equations (4.26c), (4.29) and (4.30) form the coordinator for the CDMPC (4.21).

Remark 4.2.5 *If a gradient-based method is used to numerically solve* $\nabla_{\tilde{\mathbf{X}}}L = \mathbf{0}$, $\nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}}L = \mathbf{0}$, $\nabla_{\boldsymbol{\Gamma}}L = \mathbf{0}$ *and* $\nabla_{\boldsymbol{\Pi}}L = \mathbf{0}$, *the following coordinator will result:*

$$\boldsymbol{\Gamma}_i^{q+1} = \boldsymbol{\Gamma}_i^q + \epsilon_1(\mathbf{X}_i^q - \tilde{\mathbf{X}}_i^q) \tag{4.31}$$

$$\boldsymbol{\Pi}_i^{q+1} = \boldsymbol{\Pi}_i^q + \epsilon_2(\boldsymbol{\Delta}\mathbf{U}_i^q - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^q) \tag{4.32}$$

$$\tilde{\mathbf{X}}_i^{q+1} = \tilde{\mathbf{X}}_i^q - \epsilon_3(\nabla_{\tilde{\mathbf{X}}_i}\sum_{i=1}^{N}L_i)^q \tag{4.33}$$

$$\boldsymbol{\Delta}\tilde{\mathbf{U}}_i^{q+1} = \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^q - \epsilon_4(\nabla_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_i}\sum_{i=1}^{N}L_i)^q \tag{4.34}$$

and Equation (4.30)

where ϵ_1, ϵ_2, ϵ_3 *and* ϵ_4 *are tuning parameters.*

Remark 4.2.6 *If instead of (4.22b), (4.23) is used in the compact form (4.22), sampling time* Δt *will not explicitly appear in the coordinator's equations. In this case, the following update equations are obtained for the coordinator:*

Coordinator based on a fixed-point iteration method:

1) Update equations (4.26c)

2)

$$\begin{aligned}\boldsymbol{\Gamma}_i^{q+1} = &-\varepsilon(\mathbf{X}_i^q - \tilde{\mathbf{X}}_i^{q+1})\bar{\mathbb{Q}}_i + \left(\tilde{\mathbb{A}}_{ii}^d\right)^T \boldsymbol{\Lambda}_i^q - \left(\mathbb{A}_{ii}^{c,q+1}\right)^T \boldsymbol{\Lambda}_i^q - \sum_{\substack{j=1\\ j\neq i}}^{N} \left(\mathbb{A}_{ji}^{c,q+1}\right)^T \boldsymbol{\Lambda}_j^q \\ &+ \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\tilde{\mathbb{A}}_{ii}^d\right)^T \boldsymbol{\Omega}_{i,min}^q - \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\mathbb{A}_{ii}^{c,q+1}\right)^T \boldsymbol{\Omega}_{i,min}^q \\ &- \sum_{\substack{j=1\\ j\neq i}}^{N} \left(\mathbb{C}_{jj}\left[\hat{\mathbb{A}}_{jj}^d\right]^{-1}\mathbb{A}_{ji}^{c,q+1}\right)^T \boldsymbol{\Omega}_{j,min}^q \\ &- \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\tilde{\mathbb{A}}_{ii}^d\right)^T \boldsymbol{\Omega}_{i,max}^q + \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\mathbb{A}_{ii}^{c,q+1}\right)^T \boldsymbol{\Omega}_{i,max}^q \\ &+ \sum_{\substack{j=1\\ j\neq i}}^{N} \left(\mathbb{C}_{jj}\left[\hat{\mathbb{A}}_{jj}^d\right]^{-1}\mathbb{A}_{ji}^{c,q+1}\right)^T \boldsymbol{\Omega}_{j,max}^q\end{aligned} \tag{4.35}$$

3)

$$\begin{aligned}\boldsymbol{\Pi}_i^{q+1} = &-\varepsilon(\boldsymbol{\Delta}\mathbf{U}_i^q - \boldsymbol{\Delta}\tilde{\mathbf{U}}_i^{q+1})\bar{\mathbb{R}}_i - \left(\hat{\mathbb{B}}_{ii}^d\right)^T \boldsymbol{\Lambda}_i^q - \left(\mathbb{B}_{ii}^{c,q+1}\right)^T \boldsymbol{\Lambda}_i^q - \sum_{\substack{j=1\\ j\neq i}}^{N} \left(\mathbb{B}_{ji}^{c,q+1}\right)^T \boldsymbol{\Lambda}_j^q \\ &- \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\hat{\mathbb{B}}_{ii}^d\right)^T \boldsymbol{\Omega}_{i,min}^q - \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\mathbb{B}_{ii}^{c,q+1}\right)^T \boldsymbol{\Omega}_{i,min}^q \\ &- \sum_{\substack{j=1\\ j\neq i}}^{N} \left(\mathbb{C}_{jj}\left[\hat{\mathbb{A}}_{jj}^d\right]^{-1}\mathbb{B}_{ji}^{c,q+1}\right)^T \boldsymbol{\Omega}_{j,min}^q \\ &+ \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\hat{\mathbb{B}}_{ii}^d\right)^T \boldsymbol{\Omega}_{i,max}^q + \left(\mathbb{C}_{ii}\left[\hat{\mathbb{A}}_{ii}^d\right]^{-1}\mathbb{B}_{ii}^{c,q+1}\right)^T \boldsymbol{\Omega}_{i,max}^q \\ &+ \sum_{\substack{j=1\\ j\neq i}}^{N} \left(\mathbb{C}_{jj}\left[\hat{\mathbb{A}}_{jj}^d\right]^{-1}\mathbb{B}_{ji}^{c,q+1}\right)^T \boldsymbol{\Omega}_{j,max}^q\end{aligned} \tag{4.36}$$

4)

$$\begin{aligned}&\mathbf{v}_i^{q+1}(k+l|k) = \\ &\mathbf{f}_i(\tilde{\mathbf{x}}_i^{q+1}(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i^{q+1}(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i^{q+1}(k+l|k), \tilde{\mathbf{x}}_j^{q+1}(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j^{q+1}(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j^{q+1}(k+l|k)) \\ &\qquad - \mathbf{A}_{ii}^d\tilde{\mathbf{x}}_i^{q+1}(k+l|k) - \mathbf{B}_{ii}^d \sum_{a=0}^{l} \boldsymbol{\Delta}\tilde{\mathbf{u}}_i^{q+1}(k+a|k)\end{aligned} \tag{4.37}$$

where $i = 1, \ldots, N$.

Coordinator using a gradient-based method:

$$\mathbf{\Gamma}_i^{q+1} = \mathbf{\Gamma}_i^q + \epsilon_1(\mathbf{X}_i^q - \tilde{\mathbf{X}}_i^q) \tag{4.38}$$

$$\mathbf{\Pi}_i^{q+1} = \mathbf{\Pi}_i^q + \epsilon_2(\mathbf{\Delta U}_i^q - \mathbf{\Delta}\tilde{\mathbf{U}}_i^q) \tag{4.39}$$

$$\tilde{\mathbf{X}}_i^{q+1} = \tilde{\mathbf{X}}_i^q - \epsilon_3(\nabla_{\tilde{\mathbf{X}}_i}\sum_{i=1}^{N} L_i)^q \tag{4.40}$$

$$\mathbf{\Delta}\tilde{\mathbf{U}}_i^{q+1} = \mathbf{\Delta}\tilde{\mathbf{U}}_i^q - \epsilon_4(\nabla_{\mathbf{\Delta}\tilde{\mathbf{U}}_i}\sum_{i=1}^{N} L_i)^q \tag{4.41}$$

and Equation (4.37)

where $i = 1, \ldots, N$; L_i is the Lagrangian formed by using compact form (4.22), with (4.23) as its equality constraints.

During each sampling interval, the coordinator and CDMPC exchange information until the coordinator converges. Upon convergence, the following conditions are met:

1. The values of the price vectors equal the value of the corresponding Lagrange multipliers, associated with transformed interaction equations.

2. The transformed interaction equations are satisfied.

3. The interaction equations that contain the nonlinearities, are satisfied.

4. The vector $\mathbf{v}_i$, predicts the correct values for the nonlinear and interaction effects.

Implementation of the proposed CDMPC network can be carried out systematically, according to Algorithm 5.

Algorithm 5 : Implementation of MPMC-DMPC Network for Optimal Plant-Wide Nonlinear Control of Interconnected Dynamical Processes

1. Coordinator: Iteration counter q is set to 1.
2. Coordinator: Coordinating variables $\boldsymbol{\Gamma}_i$, $\boldsymbol{\Pi}_i$, $\tilde{\mathbf{X}}_\mathbf{i}$ and $\boldsymbol{\Delta}\tilde{\mathbf{U}}_i$ are arbitrarily initialized.
3. Coordinator: $\mathbf{v}_i$ is calculated using (4.30).
4. Coordinator: Coordinating variables are sent to local controllers.
5. Local Controllers: Local optimization problems (4.21) are solved.
6. Local Controllers: Local optimal solutions $\mathbf{X}_i$ and $\boldsymbol{\Delta}\mathbf{U}_i$, and Lagrange multipliers $\boldsymbol{\Lambda}_i$ and/or $\boldsymbol{\Omega}_i$, are sent to the coordinator.
7. Coordinator: If $\left\| \begin{bmatrix} \mathbf{X} - \tilde{\mathbf{X}} \\ \boldsymbol{\Delta}\mathbf{U} - \boldsymbol{\Delta}\tilde{\mathbf{U}} \end{bmatrix} \right\| \leq \mathfrak{e}$, algorithm stops. Otherwise, next step is taken.
8. Coordinator: Pseudo-variables are updated using (4.26c).
9. Coordinator: Gradient matrices (4.28) are updated according to (E.7) and (E.8).
10. Coordinator: Local prices are updated using (4.29).
11. Coordinator: Iteration counter is increased by 1.
12. Steps 3 to 7 are repeated.

4.3 Case Study

In this section, the suggested CDMPC approach is applied to a case study borrowed from Sun and El-Farra (2008). In this case study, the plant is composed of two interconnected continuous stirred-tank reactors (CSTRs) with recycle. The schematic diagram of the plant is shown in Figure 4.1. Three parallel exothermic irreversible reactions $A \xrightarrow{k_1} B$, $A \xrightarrow{k_2} U$ and $A \xrightarrow{k_3} R$ take place in the reactors. A and B are the reactant and desired product, respectively. R and U are the undesired byproducts.

A stream containing fresh A at flow rate F_0, molar concentration C_{A0} and temperature T_0 along with a stream of recycled A from the second reactor at flow rate F_r, concentration C_{A2} and temperature T_2, are the feed stream to the first reactor.

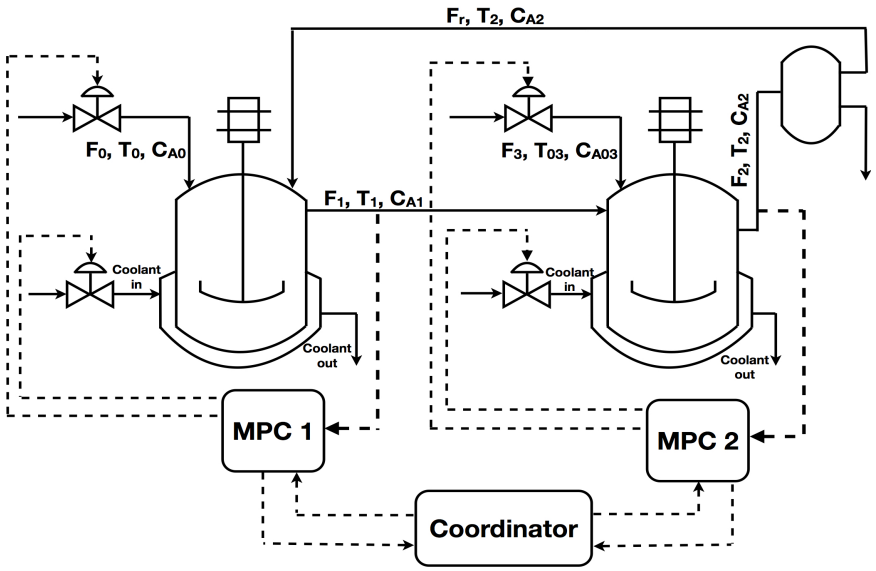


Figure 4.1: Interconnected CSTRs

The second reactor has also two feed streams, the output of the first reactor and an additional pure stream of A, at flow rate F_3, concentration C_{A03} and temperature T_{03}. The output of the second reactor enters a separator, that separates the unreacted A from the products and recycles it to the first reactor. For each of the reactors, a jacket is used to remove/provide heat. The following continuous nonlinear model has been derived for the plant:

$$\begin{aligned}
\frac{dT_1}{dt} &= \frac{F_0}{V_1}(T_0 - T_1) + \frac{F_r}{V_1}(T_2 - T_1) + \sum_{i=1}^{3} G_i(T_1)C_{A1} + \frac{\mathcal{Q}_1}{\rho c_p V_1} \\
\frac{dC_{A1}}{dt} &= \frac{F_0}{V_1}(C_{A0} - C_{A1}) + \frac{F_r}{V_1}(C_{A2} - C_{A1}) - \sum_{i=1}^{3} \mathcal{R}_i(T_1)C_{A1} \\
\frac{dT_2}{dt} &= \frac{F_1}{V_1}(T_1 - T_2) + \frac{F_3}{V_2}(T_{03} - T_2) + \sum_{i=1}^{3} G_i(T_2)C_{A2} + \frac{\mathcal{Q}_2}{\rho c_p V_2} \\
\frac{dC_{A2}}{dt} &= \frac{F_1}{V_2}(C_{A1} - C_{A2}) + \frac{F_3}{V_2}(C_{A03} - C_{A2}) - \sum_{i=1}^{3} \mathcal{R}_i(T_2)C_{A2}
\end{aligned} \tag{4.42}$$

where for $j = 1, 2$, $\mathcal{R}_i(T_j) = k_{i0}e^{-\frac{E_i}{RT_j}}$ and $G_i(T_j) = \frac{-\Delta H_i}{\rho c_p}\mathcal{R}_i(T_j)$. Also, T_j, C_{Aj}, $\mathcal{Q}_j$ and V_j are the reactor's temperature, concentration of A, the rate of heat input to the reactor and the reactor volume, respectively. For $i = 1, 2, 3$, ΔH_i, k_{i0} and E_i are the

$F_0 = 4.998\frac{m^3}{h}$	$T_0 = 300.0K$	$k_{10} = 3.0 \times 10^6 h^{-1}$	$c_p = 0.231\frac{kJ}{kgK}$
$F_1 = 39.996\frac{m^3}{h}$	$T_{03} = 300K$	$k_{20} = 3.0 \times 10^5 h^{-1}$	
$F_3 = 30.0\frac{m^3}{h}$	$C^s_{A0} = 4.0\frac{kmol}{m^3}$	$k_{30} = 3.0 \times 10^5 h^{-1}$	
$F_r = 34.998\frac{m^3}{h}$	$C^s_{A03} = 2.0\frac{kmol}{m^3}$	$E_1 = 5.0 \times 10^4 \frac{kJ}{kmol}$	
$V_1 = 1.0m^3$	$-\Delta H_1 = -5.0 \times 10^4 \frac{kJ}{kmol}$	$E_2 = 7.53 \times 10^4 \frac{kJ}{kmol}$	
$V_2 = 3.0m^3$	$-\Delta H_2 = -5.2 \times 10^4 \frac{kJ}{kmol}$	$E_3 = 7.53 \times 10^4 \frac{kJ}{kmol}$	
$R = 8.314\frac{kJ}{kmolK}$	$-\Delta H_3 = -5.4 \times 10^4 \frac{kJ}{kmol}$	$\rho = 1000.0\frac{kg}{m^3}$	

Table 4.1: Process parameters and steady-state values for the plant model (4.42)

enthalpies, rate constants and activation energies of the three reactions, respectively. c_p is the heat capacity and ρ is the fluid density. Parameter values for the nonlinear model (4.42), are given in Table 4.1.

For the given process parameters, the plant with $\mathcal{Q}_1 = \mathcal{Q}_2 = 0$, $C_{A0} = C^s_{A0}$, $C_{A03} = C^s_{A03}$ and a recycle ratio of $r = 0.5$, has two locally asymptotically stable steady-states and an unstable one at $(T^s_1, C^s_{A1}, T_2, C^s_{A2}) = (457.9K, 1.77\frac{kmol}{m^3}, 415.5K, 1.75\frac{kmol}{m^3})$. The control objective is to stabilize the plant around its open-loop unstable steady-state, to avoid high temperatures while achieving reasonable conversion. The manipulated variables are $\mathcal{Q}_1$ and C_{A0} for the first reactor and $\mathcal{Q}_2$ and C_{A03} for the second reactor.

The nonlinear continuous model 4.42, is discretized using Euler method and a sampling time of $0.003hr$, to obtain the following discrete-time nonlinear model:

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \Delta t\, \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)) \tag{4.43}$$

where $\mathbf{x}(k) = \Big[\mathbf{x}_1(k)^T,\ \mathbf{x}_2(k)^T\Big]^T \triangleq \Big[[T_1(k),\ C_{A1}(k)],\ [T_2(k),\ C_{A2}(k)]\Big]^T$, $\mathbf{u}(k) = \Big[\mathbf{u}_1(k)^T,\ \mathbf{u}_2(k)^T\Big]^T \triangleq \Big[[\mathcal{Q}_1(k),\ C_{A0}(k)],\ [\mathcal{Q}_2(k),\ C_{A03}(k)]\Big]^T$ and $\mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)) \triangleq \Big[[f_{T_1}(\mathbf{x}(k), \mathbf{u}(k)),\ f_{C_{A1}}(\mathbf{x}(k), \mathbf{u}(k))],\ [f_{T_2}(\mathbf{x}(k), \mathbf{u}(k)),\ f_{C_{A2}}(\mathbf{x}(k), \mathbf{u}(k))]\Big]^T$.

It is assumed that a decentralized control network, consisting of two MPC, controls the plant. The models used in the decentralized controllers have been obtained by first, linearizing the nonlinear model (4.42), around the steady-sate point $(T^s_1, C^s_{A1}, T^s_2, C^s_{A2}) = (457.9K, 1.77\frac{kmol}{m^3}, 415.5K, 1.75\frac{kmol}{m^3})$. The result is the following

$\mathbf{A}^c$ and $\mathbf{B}^c$ matrices:

$$\mathbf{A}^c = \left[\begin{array}{cc:cc} 25.2 & 1284.3 & 35 & 0 \\ -0.3 & -45.9 & 0 & 35 \\ \hdashline 13.3 & 0 & -2.8 & 336.2 \\ 0 & 13.3 & -0.1 & -24.9 \end{array}\right] \text{ and } \mathbf{B}^c = \left[\begin{array}{cc:cc} 0.0043 & 0 & 0 & 0 \\ 0 & 4.998 & 0 & 0 \\ \hdashline 0 & 0 & 0.0014 & 0 \\ 0 & 0 & 0 & 10 \end{array}\right] \tag{4.44}$$

Then, the continuous model (4.44) is discretized with the chosen sampling time of $0.003hr$. Ultimately, the following matrices for the discrete-time local linear models have been obtained:

$$\begin{aligned} \mathbf{A}^d_{11} &= \begin{bmatrix} 1.0790 & 3.7459 \\ -0.0009 & 0.8716 \end{bmatrix} \text{ and } \mathbf{B}^d_{11} = \begin{bmatrix} 0.0000 & 0.0283 \\ -0.0000 & 0.0140 \end{bmatrix} \\ \mathbf{A}^d_{22} &= \begin{bmatrix} 0.9936 & 0.9716 \\ -0.0003 & 0.9298 \end{bmatrix} \text{ and } \mathbf{B}^d_{22} = \begin{bmatrix} 0.0000 & 0.0147 \\ -0.0000 & 0.0289 \end{bmatrix} \end{aligned} \tag{4.45}$$

Simulations have been performed for the centralized nonlinear MPC case, in which a centralized MPC with the nonlinear prediction model (4.43), is designed to control the two reactors. To find the global minimum at each sampling time, MATLAB's *fmincon* function with *active-set* algorithm has been used in the *GlobalSearch* class. The maximum function evaluations and iterations have been set to 20000 and 15000, respectively. Also, in order to avoid any approximation in the centralized optimal solution and thus, make it comparable to the optimal solution of the CDMPC network, rather than estimating the gradient by finite difference, the exact gradient of the nonlinear constraints has been provided for the *fmincon* function.

For the CDMPC case, the network of decentralized MPC with linear model (4.45), has been transformed into a linear constrained CDMPC network. MATLAB's *quadprog* function with the *active-set* algorithm, has been used to solve the local optimization problems in the CDMPC scheme. The update equations (4.26c), (4.29) and (4.30), have been used for the coordinator. At each iteration, the coordinator uses local information sent by the controllers along with the derivatives of the nonlinear function $\mathbf{f}(\tilde{\mathbf{x}}(t), \boldsymbol{\Delta}\tilde{\mathbf{u}}(t))$ with respect to the pseudo-variables, to update the coordinating variables. The stopping criteria in the coordinator is chosen to be $\mathfrak{e} = 10^{-6}$. The rest of parameters used in the simulations, are listed in Table 4.2.

	MPC 1	MPC 2
Initial Conditions	$\mathbf{x}_1(0) = \begin{bmatrix} 462.9K & 0.27\frac{kmol}{m^3} \end{bmatrix}^T$	$\mathbf{x}_2(0) = \begin{bmatrix} 410.5K & 3.45\frac{kmol}{m^3} \end{bmatrix}^T$
Weighting Matrices	$\mathbf{Q}_{11} = 5\mathbf{I}$ $\mathbf{R}_{11} = 15\mathbf{I}$	$\mathbf{Q}_{22} = 5\mathbf{I}$ $\mathbf{R}_{22} = 15\mathbf{I}$
Upper Bounds	$\mathbf{u}_1^{max} = \begin{bmatrix} 5\frac{KJ}{hr} & 8\frac{kmol}{m^3} \end{bmatrix}^T$	$\mathbf{u}_2^{max} = \begin{bmatrix} 5\frac{KJ}{hr} & 4\frac{kmol}{m^3} \end{bmatrix}^T$
Lower Bounds	$\mathbf{u}_1^{min} = \begin{bmatrix} -5\frac{KJ}{hr} & 0\frac{kmol}{m^3} \end{bmatrix}^T$	$\mathbf{u}_2^{min} = \begin{bmatrix} -5\frac{KJ}{hr} & 0\frac{kmol}{m^3} \end{bmatrix}^T$
Prediction Horizon	10	10
Control Horizon	5	5

Table 4.2: Parameters used in the proposed CDMPC

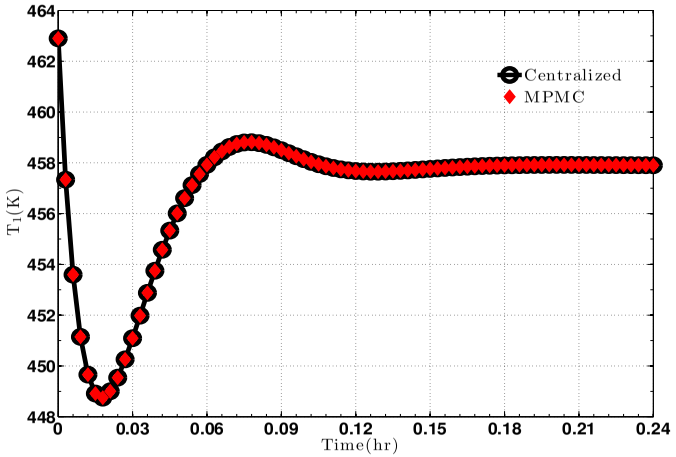
4.3.1 Simulation Results

Figure 4.2, in which output trajectories for both the centralized nonlinear MPC and CDMPC have been plotted, shows that the process outputs (states) resulting from the coordinated distributed network matched the states obtained from the centralized problem. Similarly, in Figure 4.3, trajectories for the optimal control inputs for the centralized nonlinear MPC and the proposed CDMPC coincided well with each other.

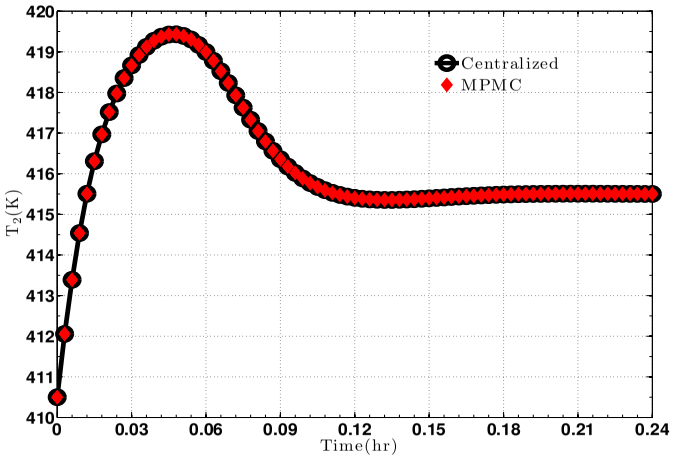
Simulation results show that the CDMPC network successfully produced the optimal solution to the centralized nonlinear MPC problem because: 1) the *active-set* algorithm chosen to solve the QP problems of CDMPC, terminated successfully at every communication cycle; and 2) as Figure 4.4 shows, during all sampling intervals, the coordinator converged within the chosen error tolerance of $\mathfrak{e} = 10^{-6}$. Therefore, at the end of the communication cycles, optimality conditions of the overall nonlinear control system were met and the distributed controllers calculated the optimal centralized nonlinear MPC solution.

At each sampling interval, the number of communications between the CDMPCs and the coordinator, until convergence, is shown in Figure 4.5. For this case study, the coordinator found the correct values of the coordinating variables in finite and reasonably low number of iterations.

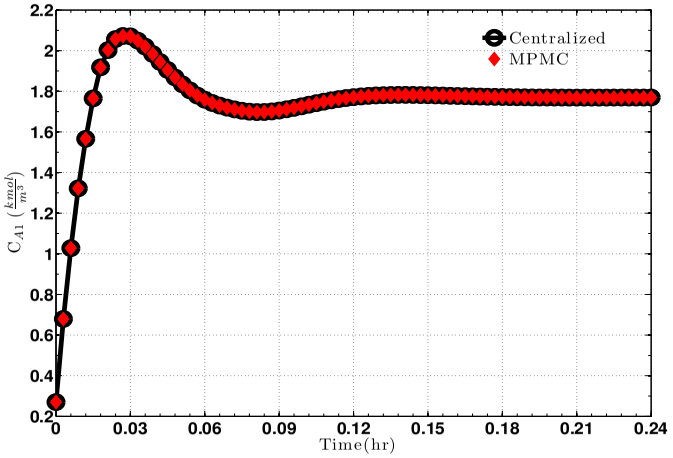
Figure 4.2: Output trajectories resulted from centralized and CDMPC



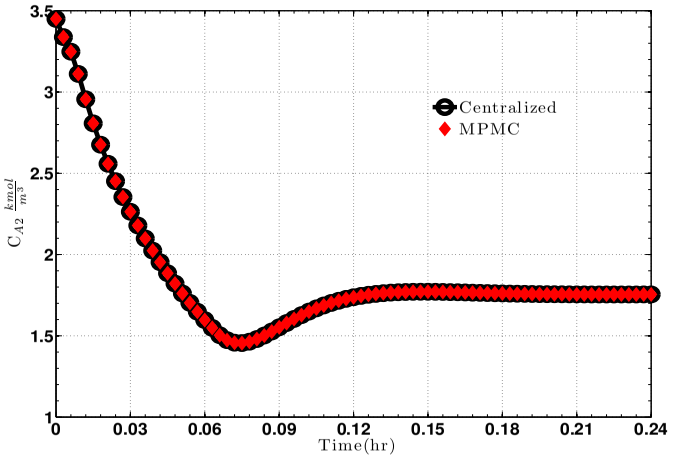
(a) Temperature of unit 1



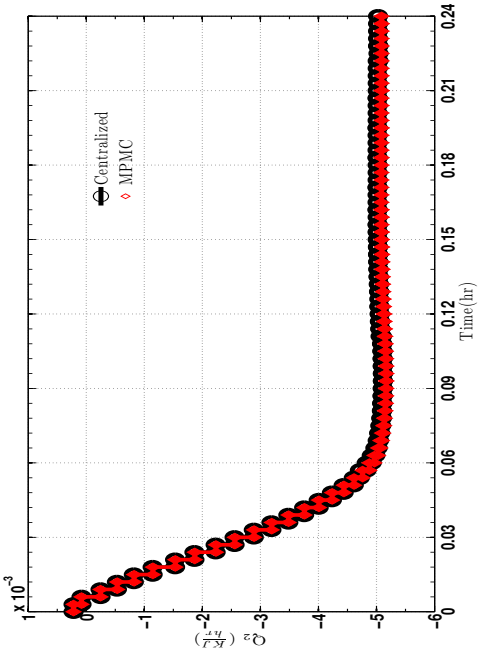
(b) Temperature of unit 2



(c) Concentration of unit 1

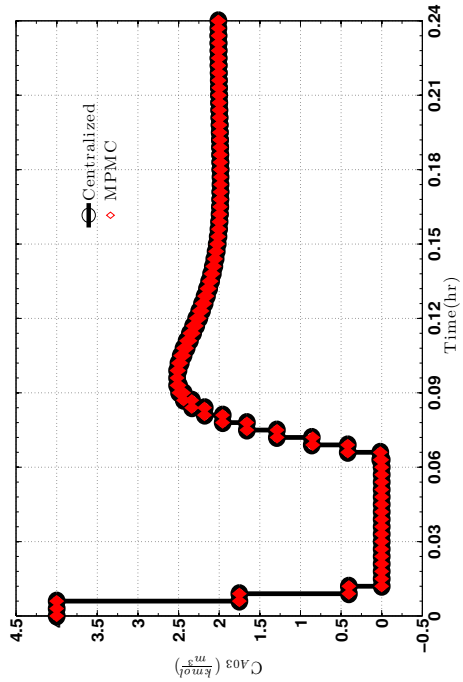


(d) Concentration of unit 2



(a) Heat input rate for unit 1

(b) Heat input rate for unit 2



(c) Inlet reactant concentration for unit 1

(d) Inlet reactant concentration for unit 2

Figure 4.3: Control input trajectories resulted from centralized and CDMPC

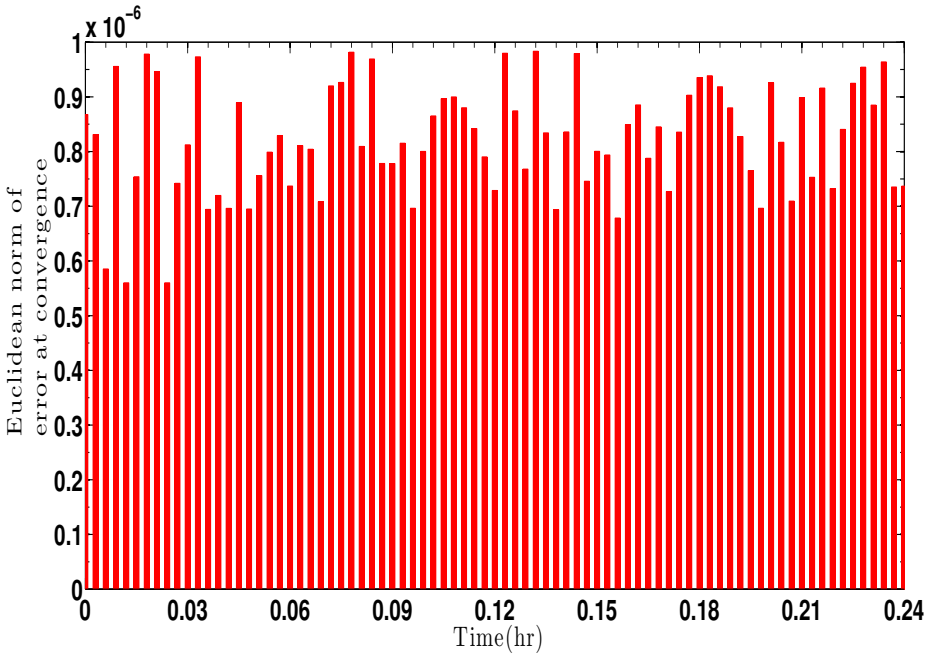


Figure 4.4: Norm of the error vector $\mathfrak{e}$, at the end of communication cycles

On the other hand, to perform the centralized nonlinear MPC simulation, large numbers were chosen for the maximum number of function evaluations and iterations, so that the optimization algorithm for each of the assigned solvers in the *GlobalSearch* toolbox, could terminate successfully; however, choosing large numbers for the tuning parameters of the optimization algorithm significantly prolonged the computation time. Under the same computing conditions, the simulation for the centralized nonlinear MPC problem took roughly three hours while the simulation for the proposed CDMPC network completed in about 15 minutes.

The computational burdens faced in solving the centralized nonlinear MPC problem for this fairly small case-study is indicative of how difficult and maybe impossible solving centralized nonlinear MPC problem could be, in practice. On the contrary, the proposed linear CDMPC approach smoothly and successfully obtained the optimal solution to the plant-wide nonlinear MPC problem.

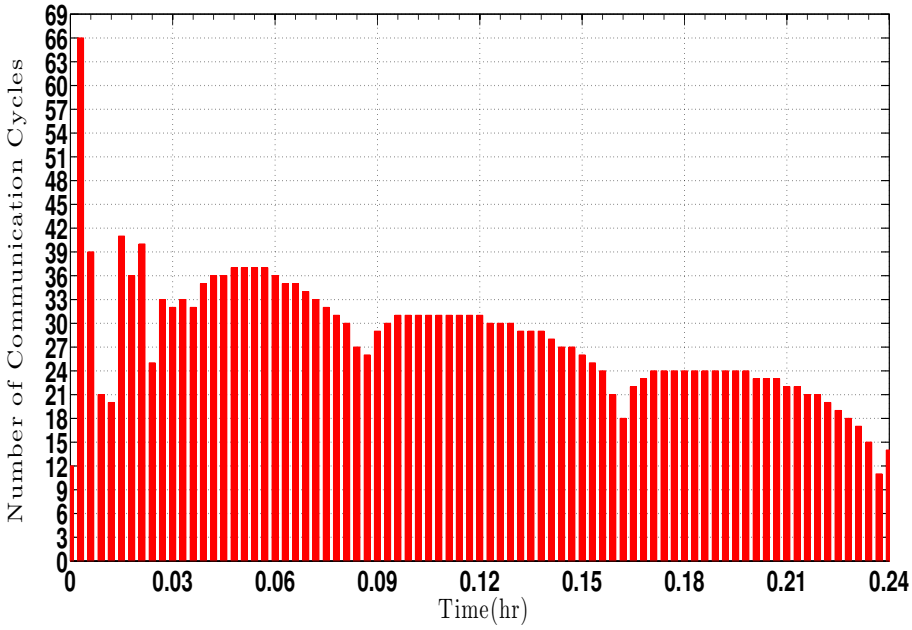


Figure 4.5: Number of communication cycles

4.4 Summary

This chapter intends to contribute to the development of CDMPC networks for plants with nonlinear dynamics. To this end, a convex MPMC-DMPC network is proposed for a class of plant-wide (non-convex) nonlinear MPC problems, in which nonlinear process models are at least once continuously differentiable.

Inspired by an exact linearization technique, the process models are re-stated in terms of two sets of equations, 1) a system of linear equations that forms the local dynamic equations; and 2) a set of nonlinear equations that creates the interaction equations. Once the two systems of equations are formed, the MPMC method is applied to convert the existing linear decentralized MPC network into a CDMPC network, that has linear CDMPCs and a (nonlinear) coordinator.

The coordinator is synthesized by the numerical method used to solve portions of first-order optimality conditions of the aggregate of CDMPCs and interaction equations. Since the optimality conditions usually lead to nonlinear algebraic equations, the coordinator often needs to find the solution to a system of nonlinear equations, using an appropriate numerical method. The main difference between the

MPMC-DMPC network for plants with linear dynamics and plants with nonlinear dynamics is that in the former, the coordinator always solves a system of linear algebraic equations.

Compared with the centralized and other distributed schemes, in which (non-convex) nonlinear optimization problems are solved, the proposed CDMPC scheme, solves a group of QP problems and a system of nonlinear equations. Considering the availability of efficient numerical methods, solving nonlinear equations is less challenging than solving non-convex optimization problems. Therefore, in addition to the benefits of CDMPC networks, significant computational benefits can be expected from the proposed scheme.

Chapter 5

Conclusions and Future Work

The focus of this thesis is on the development of CDMPC networks by using multilevel optimization-based coordination methods, in order to upgrade the control performance of existing decentralized MPC network to the control performance of plant-wide MPC. Goal Coordination, Interaction Prediction Coordination and Modified Pseudo-Model Coordination are the coordination approaches applied in this work. Among the wide range of MPC applications, this thesis considers systematic development of coordinated distributed networks for linear/convex MPC, individual chance-constrained MPC and nonlinear MPC.

Since there is a small literature on CDMPC, an important challenge is in providing a general framework wherein, regardless of type of the coordination method used, CDMPC and the coordinator can be systematically synthesized. This challenge is addressed in Chapter 2, where for the first time, a thorough study is carried out on applying the GC, IPC and MPMC methods to a network of decentralized linear constrained MPC. It is shown that the general procedure, which alters the decentralized MPC into a CDMPC network, involves five main steps; 1) interaction models are included in the local process models; 2) local process models are re-stated in terms a system of local dynamic equations and a system of interaction equations; 3) the aggregate of decentralized MPC with the modified process models, is formed; 4) in the aggregate problem, the system of interaction equations is relaxed, which results in formation of CDMPC; 5) the numerical strategy chosen to solve the

relaxed aggregate problem, leads to synthesis of the appropriate coordinator. The combination of CDMPC and coordinator always possesses an inherent hierarchical structure because of the involvement of iterative procedures in numerical methods. Following this standard procedure, three CDMPC schemes are proposed for plant-wide linear model predictive control of interconnected dynamical systems.

The coordinator design problem for CDMPC, is mathematically viewed as numerically solving a system of algebraic equations. In the GC-based DMPC network, the coordinator is designed based on the algorithm of the numerical method used for solving a dual optimization problem. The resulting coordinator implicitly accounts for the effects of local inequality constraints on the coordination algorithm. In the IPC-DMPC and MPMC-DMPC networks, solving portions of the optimality conditions of the composite CDMPC (relaxed aggregate problem), by either fixed-point iterations or gradient-based approaches, synthesizes the coordinator. The impacts of local inequality constraints can explicitly be accounted for in the coordinator update equations. Since in the GC method, a dual optimization problem is solved, the resulting coordinator has different convergence behaviour in comparison with the coordinators in the other two coordination approaches. The coordinators in the IPC-DMPC and MPMC-DMPC schemes, have similar convergence characteristics because of using same design strategy. For every coordination algorithm, performing a thorough convergence study is an important yet challenging task. In chapter 2, convergence accuracy of the three proposed coordination algorithms are studied by showing that the numerical algorithms converge to the solution of the system of algebraic equations used in designing the coordinators .

The proposed CDMPC networks have several important benefits. In addition to producing a flexible and maintainable control structure, the CDMPC networks have the performance of centralized linear MPC. Also, the coordinated distributed networks are designed using the existing network of decentralized MPC. Therefore, there is no need for major changes in the control structure of the plant. One limitation of the proposed CDMPC networks is that they can only guarantee converging to optimal plant-wide performance, if local controllers have convex structures. According to literature, another limitation of the coordination methods used in this work, is that

they are in the class of non-feasible multilevel optimization methods, meaning that interaction equality constraints are satisfied only when the coordination algorithm has converged. Therefore, solutions generated in an intermediate iteration, should not be used (Singh *et al.* (1975) and Mahmoud (1977)), as they are not feasible with respect to the *centralized* optimization problem. It should be noted that, if at any stage of the coordination algorithm, CDMPCs successfully solve their optimization problem the intermediate solutions will be feasible with respect to the distributed scheme. Thus, if it is guaranteed that the distributed controllers provide closed-loop stable solutions, then intermediate solutions can also be implemented; however, with the current level of knowledge, if for any reason, the coordination algorithm terminates prematurely, the safest action would be to switch the CDMPC network back to the decentralized mode.

The detailed study of coordinated distributed schemes for standard linear MPC in chapter 2, provides a general mathematical insight into developing CDMPC networks. Considering the inherent existence of uncertainty in processes, it is important that the controllers perform well despite the uncertainties affecting the plant. Chapter 3 intends to address the issue of how uncertainties affect CDMPC and coordinator design. To this end, among various design methods that explicitly incorporate uncertainty into MPC synthesis, individual chance-constrained MPC is chosen. Individual chance-constrained MPC is a simple yet informative approach, to account for the impacts of normally distributed uncertain disturbances in developing probabilistic CDMPC networks. In addition to their simplicity, single chance-constraints can be converted into equivalent linear inequalities. Therefore, the resulting MPC problem is convex and the proposed chance-constrained CDMPC can guarantee achieving the optimal plant-wide solution. By using the systematic procedure presented in chapter 2, coordination methods are applied to convert the existing decentralized chance-constrained MPC to CDMPC.

In producing probabilistic coordinated distributed networks, it is necessary to keep the prediction model as equality constraints. This way, the prediction models can be re-stated by local dynamic equations and interaction dynamic equations, and the application of coordination methods becomes straightforward. The process models

are uncertain, as they contain uncertain parameters and/or disturbances. Therefore, known stochastic information (expected values) is employed to convert the uncertain process models into deterministic models. The uncertainty effects are also taken into account by using the uncertain process models in the probabilistic constraints.

Another achievement in chapter 3, is regarding resolving convergence issues of coordinator in the GC method, specially in the presence of active local inequality constraints. Since inequality constraints are an integral part of chance-constrained MPC formulation, it is important to address the convergence issues by looking for more efficient numerical methods for solving the resulting dual optimization problem. Inspired by the *augmented Lagrangian methods* and by applying the *Separable Augmented Lagrangian Algorithms*, the CDMPC in the GC method is modified and a new coordinator formulation has been obtained. In the modified coordinator, the need for optimal step-size calculation is avoided. As expected, simulation results showed that the new coordinator has improved convergence performance in the sense that it can converge in finite number of iterations; however comparing with the other two coordinated distributed schemes, the number of required iterations are considerably higher. The proposed modified CDMPC and coordinator are applicable to deterministic GC-DMPC networks, as well.

The proposed probabilistic CDMPC schemes pave the way to develop more advanced robust CDMPC networks, in which other stochastic optimization methods such as joint chance-constraints and recourse problems, and Robust optimization methods are used. Similar to the deterministic case, the proposed probabilistic coordinated distributed networks can only guarantee producing the optimal plant-wide solution, if the CDMPCs have convex structure.

The final issue addressed in this work is regarding the performance limitation of the proposed coordination methods when the CDMPC has non-convex structure. In chapter 4, by using MPMC method, a novel CDMPC network is proposed for processes with nonlinear dynamics that result in non-convex MPC formulations. A useful characteristic of the MPMC approach is that instead of the interaction equations, the linear transformed equations are relaxed. Therefore, interaction equations which contain the nonlinearities, do not appear in the CDMPC formulation.

This is while, in the GC and IPC method, the interaction equations are relaxed, and portions of them appear explicitly in the objective function of CDMPC that form the coordinating terms. By using the idea of exact linearization technique, the nonlinear process models are re-stated by a system of linear local dynamic equations and a system of nonlinear interaction equations. The existing linear decentralized MPC network is then altered into a network of linear CDMPC that has the performance of plant-wide nonlinear MPC. The coordinator design problem is viewed as numerically solving a system of algebraic (nonlinear) equations. The system of algebraic equations are obtained from portions of the optimality conditions for the optimization problem resulting from aggregate of CDMPC and interaction equations.

Other than the common advantages of coordinated distributed networks, the proposed CDMPC scheme in chapter 4, involves solving QP problems for the CDMPC and a system of algebraic equations for the coordinator. Therefore, if an efficient numerical method is used in solving the system of algebraic equations, significant computational benefits will be achieved in comparison with nonlinear DMPC where nonlinear and often non-convex optimizations must be solved. The limitation of the proposed scheme is that it is applicable to plant-wide optimal control of systems with interconnected nonlinear dynamics, where the nonlinear process models are at least once continuously differentiable.

In addition to the aforementioned contributions, the studies and results obtained in this thesis, provide grounds for more insightful comparison between coordinated and non-coordinated DMPC methods. In both approaches, interaction models are needed. In the CDMPC networks, complete interaction models are needed; however, considering the amount of performance improvement achieved by coordinated distributed methods, the effort for identifying interaction models may be completely paid off. One of the criticisms of the CDMPC methods, is that an additional piece, the coordinator, has to be designed and its synthesis and operation is thought to be complicated. This is not true; as is shown throughout this thesis, the involved mathematical and optimization concepts, make coordinator design a straightforward and fairly simple process. The coordinator is essentially a computer program that contains a set of algebraic equations, not a physical addition to the plant equipment.

5.1 Directions for Future Research

Coordination methods and CDMPC networks studied in this thesis, provide an insightful guide for further theoretical developments, addressing open issues and exploring new applications.

An important theoretical study for the proposed CDMPC networks, concerns convergence properties. Convergence accuracy studies in chapter 2, indicate that the suggested coordination algorithms are convergent; however, more in-depth studies should be carried out, to obtain detailed properties such as convergence rates and conditions that affect convergence behaviour of the coordinator. The well-studied convergence proofs of the numerical methods can provide a powerful tool for detailed convergence study of coordination algorithms; for instance, contraction mapping theorem can be used to obtain convergence properties for the coordinators in the IPC and MPMC methods that are obtained based on fixed-point iteration technique.

Considering that coordinator is designed by the numerical algorithm used either for solving a dual optimization problem or portions of optimality conditions, further research is needed for possible improvements in convergence characteristics of coordination algorithms, by using more efficient numerical methods. Solving the dual optimization problem by SALA, to modify the CDMPC and coordinator in the GC method, in chapter 3, is an example of such possible enhancements.

Throughout this thesis, state-space models are used in the MPC formulations. It is assumed that full state measurements are available; however, measurements for some variables may not be available at every sampling time and/or some states may not be measured. Therefore, state estimation and observer design problems are needed be addressed in developing CDMPC networks.

The use of Input-Output process models in the MPC formulations should also be considered. It is expected that this extended application be fairly straightforward to accomplish, since the concepts and procedures in establishing coordinated distributed networks will remain unchanged.

The proposed individual chance-constrained CDMPC networks under uncertain disturbances in chapter 3, have the capability of extension to the case in which

uncertainties exist in both disturbances and model parameters. Also, joint chance-constrained CDMPC, as a more realistic approach, should be considered. Depending on the type of distribution function used for the probabilistic constraints, the resulting problem may fall in the category of non-convex optimization problems. In order to successfully apply coordination algorithms discussed in this thesis, one approach would be to use convex approximations for the non-convex joint chance-constrained MPC.

Another application of the studied CDMPC networks, would be in the context of Robust CDMPC, where the known bounds on the uncertainties result in a worst-case optimization problem for the CDMPC.

In chapters 2 and 4, an invariant terminal region can be calculated for the plant-wide MPC problem, so that it can be included in developing the CDMPC networks. In doing so, at each sampling time, when the coordinator converges, the obtained open-loop solution can guarantee a closed-loop stable system. If the invariant terminal region includes a terminal penalty term, incorporating the penalty term in the CDMPC formulations will be challenging because this term is usually non-separable. In addition to the approach adopted in this thesis, which is using a long enough prediction horizon, three possible solutions to overcome the difficulty of dealing with a non-separable terminal weighting matrix could be: 1) considering approaches for optimization of non-separable objective function subject to independent constraints, 2) using MPC design methods that only use terminal constraints, 3) seeking for alternative approaches for synthesizing stabilizing MPC.

Regarding the CDMPC networks proposed in this thesis, an important open issue to be addressed concerns stability of the intermediate solutions, and studying if these solutions can provide better performance than decentralized MPC. If these properties are guaranteed, in the case of early termination of the coordination algorithm, the amount of performance loss using the intermediate solutions will be less compared with the case that the control network is switched back to the decentralized MPC mode.

Throughout this thesis, it is assumed that all processes have similar time scales. Consequently, one sampling period is used for sampling all the variables, and the

CDMPCs operate simultaneously. In practice, the processes in a plant usually have different time scales. Therefore, depending on the process dynamics, different sampling times are used to sample different variables. Also, local controllers execute their control calculations at different control rates and/or sampling instants. It is important to address the issues of asynchronous CDMPC. The existence of different time scales in a plant suggests that multilayer hierarchical structures are embedded within the CDMPC network, which itself has a multiechelon hierarchical structure (Mesarovic *et al.* (1970), Mahmoud (1977) and Scattolini (2009)). Possible solutions to solve such problems would incorporate optimization and control theories of multilayer systems into the coordinated distributed approaches.

Bibliography

Bauman, E. J. (1966). Multi-Level Optimization Techniques with Application to Trajectory Decomposition. PhD thesis. University of California.

Bazaraa, M. S. and C. M. Shetty (1979). *Nonlinear Programming Theory and Algorithms*. John Wiley & Sons.

Beasley, J. E. (2013). Operations Research Notes: Stochastic Programming. `http://people.brunel.ac.uk/~mastjjb/jeb/or/sp.html`.

Bemporad, A. and M. Morari (1999). Robust model predictive control: A survey. *Robustness in Identification and Control* **245**, 207–226.

Bertsekas, D. P. (1999). *Nonlinear Programming*. 2nd ed.. Athena Scientific.

Blackmore, L. and M. Ono (2009). Convex chance constrained predictive control without sampling. In: *Proceedings of the AIAA Guidance, Navigation and Control Conference*.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.

Camacho, E. F. and C. Bordons (2007). Nonlinear Model Predictive Control: An Introductory Review. In: *Assessment and future directions of nonlinear model predictive control*. Vol. 358 of *Lecture notes in control and information sciences*. pp. 1–16. Springer Berlin/ Heidelberg.

Cannon, M. (2004). Efficient nonlinear model predictive control algorithms. *Annual Reviews in Control* pp. 229–237.

Chen, H. and F. Allgower (1998). A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability. *Automatica* **34**(10), 1205–1217.

Chen, X., M. Heidarinejad, J. Liu and P. D. Christofides (2012). Distributed economic MPC: Application to a nonlinear process network. *Journal of Process Control* **22**, 689–699.

Cheng, R. (2007). Decomposition and Coordination of Large-Scale Operations Optimization. PhD thesis. University of Alberta.

Cheng, R., J. F. Forbes and W. S. Yip (2007). Price-driven coordination method for solving plant-wide mpc problems. *Journal of Process Control* **17**, 429–438.

Christofides, P. D., R. Scattolini, D. Munoz de la Pena and J. Liu (2013). Distributed model predictive control: A tutorial review and future research directions. *Computers & Chemical Engineering* **51**, 21–41.

Cohen, G. (1977). On an algorithm of decentralized optimal control. *Journal of mathematical analysis and applications* **59**, 242–259.

Cohen, G. (1978). Comments on the paper "Multilevel control and optimization using general gradients technique". *International Journal of Control* **27**, 327.

Cohen, G., A. Beneviste and P. Bernhard (1974). Coordination algorithms for optimal control problems: Part I.. Technical Report Report No. A/57. Ecole de Mines, Paris.

Cohen, G. and B. Miara (1990). Optimization with an auxiliary constraint and decomposition. *SIAM J. Control and Optimization* **28**(1), 137–157.

COSP, The Committee on Stochastic Programming (2012). Stochastic programming community home page. `http://stoprog.org/`.

Cuzzola, F. A., J. C. Geromel and M. Morari (2002). An improved approach for constrained robust model predictive control. *Automatica* **38**(7), 1183–1189.

Doan, M. D., T. Keviczky and B. De Schutter (2011). An iterative scheme for distributed model predictive control using Fenchel's duality. *Journal of Process Control* **21**, 746–755.

Dussault, J. P., O. Mandione Gueye and P. Mahey (2005). Separable augmented Lagrangian algorithm with multidimensional scaling for monotropic programming. *Journal of Optimization Theory and applications* **127**(2), 329–345.

Findeisen, R. and F. Allgower (2002). An introduction to Nonlinear Model Predictive Control. In: *Control, 21st Benelux Meeting on Systems and Control, Veidhoven*. pp. 1–23.

Geoffrion, A. M. (1970). Elements of large-scale mathematical programming, Parts I and II. *Management Science* pp. 652–675 and 676–691.

Geoffrion, A. M. (1971). Duality in nonlinear programming: A simplified applications-oriented development. *SIAM rev.* **13**, 1–37.

Haimes, Y. Y. (1977). *Hierarchical Analyses of Water Resources Systems*. McGraw-Hill Series in Water Resources and Environmental Engineering. McGraw-Hill.

Hamdi, A. and S. K. Mishra (2011). *Topics in Nonconvex Optimization*. Chap. Decomposition methods based on augmented Lagrangians: A Survey. Vol. 50 of *Springer optimization and its applications*. 1st ed.. Springer New York.

Hamdi, A., P. Mahey and J. P. Dussault (1997). *A new decomposition method in non convex programming via separable augmented Lagrangians*. pp. 90–104. Vol. 452 of *Lecture notes in economics and mathematical systems*. Springer New York.

Hassan, M. and M. Singh (1976). The Optimization of Nonlinear Systems Using a New Two Level Method. *Automatica* **12**, 359–363.

Jamshidi, M. (1983). *Large-Scale Systems Modeling and Control*. Vol. 9 of *North-Holland Series in System Science and Engineering*. Elsevier Science.

Jose, R. A. and L. H. Ungar (1998). Auction-driven coordination for plantwide optimization. In: *The Foundations of Computer Aided Process Operations Conference (FOCAPO)*. Snowbird, Utah.

Jose, R. A. and L. H. Ungar (2000). Pricing interprocess streams using slack auctions. *AIChE Journal* **46**(3), 575–587.

Kall, P. and S. W. Wallace (1994). *Stochastic Programming*. Wiley.

Kothare, M. V., V. Balakrishnan and M. Morari (1996). Robust constrained model predictive control using linear matrix inequalities. *Automatica* **32**(10), 1361–1379.

Langson, W., I. Chryssochoos, S. V. Rakovic and D. Q. Mayne (2004). Robust model predictive control using tubes. *Automatica* **40**, 125–133.

Lasdon, L. S. (1970). *Optimization Theory for Large Systems*. Macmillan Series in Operation Research. The Macmillan Company.

Leunberger, D. G. (1984). *Linear and Nonlinear Programming*. second ed.. Addison-Wesley.

Li, P., H. Arellano-Garcia and G. Wozny (2008). Chance constrained programming approach to process optimization under uncertainty. *Computers & Chemical Engineering* **32**, 25–45.

Li, P., M. Wendt and G. Wozny (2000). Robust model predictive control under chance constraints. *Computers & Chemical Engineering* **24**, 829–834.

Li, P., M. Wendt and G. Wozny (2002). A probablistically constrained model predictive controller. *Automatica* **38**, 1171–1176.

Liu, J., D. Munoz de la Pena and P. D. Christofides (2009). Distributed model predictive control of nonlinear process systems. *AIChE Journal* **55**(5), 1171–1184.

Liu, J., D. Munoz de la Pena and P. D. Christofides (2010*a*). Distributed model predictive control of nonlinear systems subject to asynchronous and delayed measurements. *Automatica* **46**, 52–61.

Liu, J., X. Chen, D. Munoz de la Pena and P. D. Christofides (2010*b*). Sequential and iterative architectures for distributed model predictive control of nonlinear process systems. *AIChE Journal* **56**(8), 2137–2149.

Maciejowski, J. (2000). *Predictive Control with Constraints*. Prentice Hall.

Maestre, J. M., D. Munoz de la Pena, E. F. Camacho and T. Alamo (2011). Distributed model predictive control based on agent negotiation. *Journal of Process Control* **21**, 685–697.

Mahmoud, M. S. (1977). Multilevel systems control and applications: A survey. *IEEE Transactions on Systems, MAN and Cybernetics* **SMC-7**(3), 125–143.

Mahmoud, M. S. (1978). Optimal control of constrained problems by the costate coordination structure. *Automatica* **14**, 31–40.

Mahmoud, M. S., W. G. Vogt and M. H. Mickle (1977). Multilevel control and optimization using generalized gradients technique. *International Journal of Control* **25**, 525–543.

Marcos, N. I. (2012). Coordinated-Distributed Optimal Control of Large-Scale Linear Dynamic Systems. PhD thesis. University of Alberta.

Marcos, N. I., J. F. Forbes and M. Guay (2009). Coordination of distributed model predictive controllers for constrained dynamic processes.

Marcos, N. I., R. Cheng and J. F. Forbes (2008). Distributed model predictive control for large-scale dynamic systems.

Mayne, D. Q., J. B. Rawlings, C. V. Rao and P. O. M. Scokaert (2000). Constrained model predictive control: stability and optimality. *Automatica* **36**, 789–814.

Mesarovic, M. D., D. Macko and Y. Takahara (1970). *Theory of Hierarchical Multilevel Systems*. Academic Press. New York.

Mhaskar, P., N. H. El-Farra and P. D. Christofides (2006). Stabilization of nonlinear stabilization of nonlinear systems with state and control constraints using Lyapunov-based predictive control. *Systems and Control Letters* **55**, 650–659.

Nocedal, J. and S. Wright (2006). *Numerical Optimization*. Springer series in Operations research and financial. 2nd ed.. Springer New York.

Pearson, J. D. (1971). Dynamic decomposition techniques. In: *Optimization Methods for Large-Scale Systems with Application* (David A. Wismer, Ed.). McGraw-Hill.

Rao, S. S. (1996). *Engineering Optimization: Theory and Practice*. first ed.. John Wiley & Sons.

Rawlings, J. B. (1999). Tutorial: Model Predictive Control Technology. Proceedings of the American Control Conference. San Diego, California.

Rawlings, J. B. and B. T. Stewart (2008). Coordinating multiple optimization-based controllers: New opportunities and challenges. *Journal of Process Control* **18**, 839–845.

Scattolini, R. (2009). Architectures for distributed and hierarchical model predictive control- A review. *Journal of Process Control* **19**, 723–731.

Scheu, H. and W. Marquardt (2011). Sensitivity-based coordination in distributed modle predictive control. *Journal of Process Control* **21**, 715–728.

Schwarm, A. T. and M. Nikolaou (1999). Chance-constrained model predictive control. *AIChE Journal* **45**(8), 1743–1752.

Simmons, M. D. (1976). Comments on "a new algorithm for the on-line multi-level control of large interconnected systems with fast dynamics". *International Journal of Control* **24**(3), 441–442.

Singh, M. G. (1975). A new algorithm for the on-line multi-level control of large interconnected systems with fast dynamics. *International Journal of Control* **21**(4), 587–597.

Singh, M. G. (1980). *Dynamical Hierarchical Control*. second ed.. North-Holland.

Singh, M. G. and A. Titli (1978). *Systems Decomposition, Optimisation and Control*. Pergamon Press.

Singh, M. G., S. A. W. Drew and J. F. Coales (1975). Comparisons of practical hierarchical control methods for interconnected dynamical systems. *Automatica* **11**, 331–350.

Sorenson, H. M. and H. M. Koble (1984). The PMSS approach to large-scale optimal control problems. *Computers & Electrical Engineering* **11**(1), 29–51.

Stewart, B. T. (2010). Plantwide cooperative distributed model predictive control. PhD thesis. University of Wisconsin-Madison.

Stewart, B. T., S. J. Wright and J. B. Rawlings (2011). Cooperative distributed model predictive control for nonlinear systems. *Journal of Process Control* **21**, 698–704.

Sun, Y. and N. H. El-Farra (2008). Quasi-decentralized model-based networked control of process systems. *Computers & Chemical Engineering* **32**, 2016–2029.

Takahara, Y. (1965). Multilevel approach to dynamic optimization. Master's thesis. Case Institute of Technology.

WIKIPEDIA (2013). Nash equilibrium. `http://en.wikipedia.org/wiki/Nash_equilibrium`.

Venkat, A. N. (2006). Distributed Model Predictive Control: Theory and Applications. PhD thesis. University of Wisconsin-Madison.

Vert, J. P. (2006). Nonlinear optimization: Duality. `http://cbio.ensmp.fr/~jvert/teaching/2006insead/slides/4_duality/duality.pdf`.

Wan, Z. and M. V. Kothare (2003). An efficient off-line formulation of robust model predictive control using linear matrix inequalities. *Automatica* **39**(5), 837–846.

Wang, Y. J. and J. B. Rawlings (2004). A new robust model predictive control method I: theory and computation. *Journal of Process Control* **14**(3), 231–247.

Appendix A

The following flowcharts, show the information exchange between the CDMPC and the coordinator in the GC, IPC and MPMC methods, discussed in Chapter 2.

Before proceeding further, the *known interaction* portion, used in the local process models of the CDMPC (2.15), (2.31) and (2.77) are written in a summarized vector form, so that the obtained short form can be used in the flowcharts. To this end, the vector $\mathfrak{v}_i$ is defined as:

$$\mathfrak{v}_i(k+l) = \sum_{\substack{j=1 \\ j\neq i}}^{N} [\alpha \mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1)] \tag{A.1}$$

To have $\mathfrak{v}_i(k+l)$ be compatible with the notation used for other exchanged variables in the flowcharts, it is written over the entire prediction horizon as below:

$$l = 0: \quad \mathfrak{v}_i(k) = \sum_{\substack{j=1 \\ j\neq i}}^{N} [\mathbf{A}_{ij}\mathbf{x}_j(k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1)]$$

$$l = 1: \quad \mathfrak{v}_i(k+1) = \sum_{\substack{j=1 \\ j\neq i}}^{N} \mathbf{B}_{ij}\mathbf{u}_j(k-1)$$

$$\vdots$$

$$l = H_p - 1: \quad \mathfrak{v}_i(k+H_p-1) = \sum_{\substack{j=1 \\ j\neq i}}^{N} \mathbf{B}_{ij}\mathbf{u}_j(k-1)$$

$$\Longrightarrow$$

$$\mathfrak{V}_i \triangleq \begin{bmatrix} \mathfrak{v}_i(k) \\ \mathfrak{v}_i(k+1) \\ \vdots \\ \mathfrak{v}_i(k+H_p-1) \end{bmatrix} = \sum_{\substack{j=1 \\ j\neq i}}^{N} \begin{bmatrix} \mathbf{A}_{ij}\mathbf{x}_j(k) \\ \mathbf{B}_{ij}\mathbf{u}_j(k-1) \\ \vdots \\ \mathbf{B}_{ij}\mathbf{u}_j(k-1) \end{bmatrix} \tag{A.2}$$

A.1 Information Flow in the GC Method

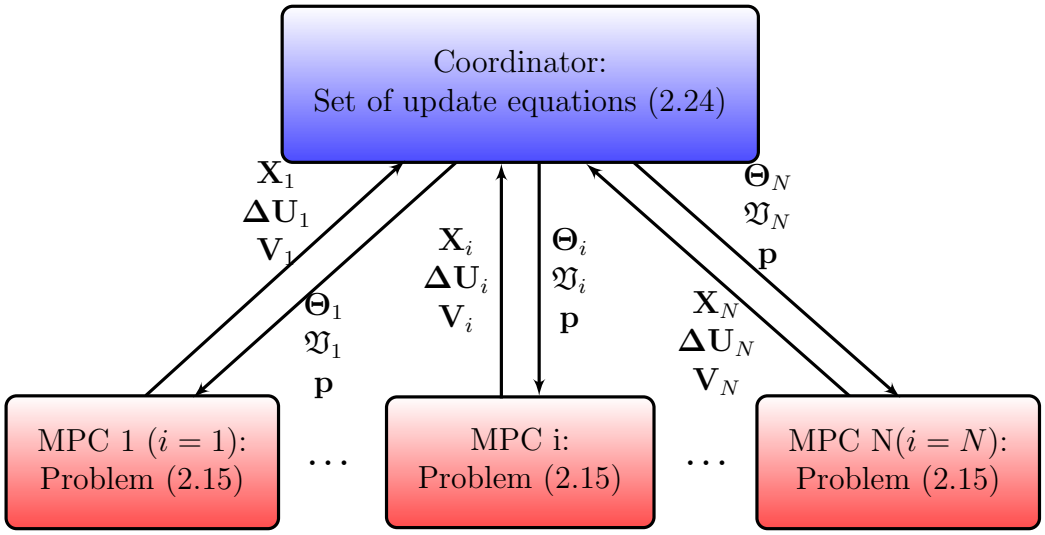


Figure A.1: Information flow in the GC-DMPC network

It should be noted that if Newton's method is used, along with the local solutions, local sensitivity information $\frac{d\mathbf{X}_i}{d\mathbf{p}}$, $\frac{d\mathbf{\Delta U}_i}{d\mathbf{p}}$ and $\frac{d\mathbf{V}_i}{d\mathbf{p}}$, has to be sent from local controllers to the coordinator, as well.

A.2 Information Flow in the IPC Method

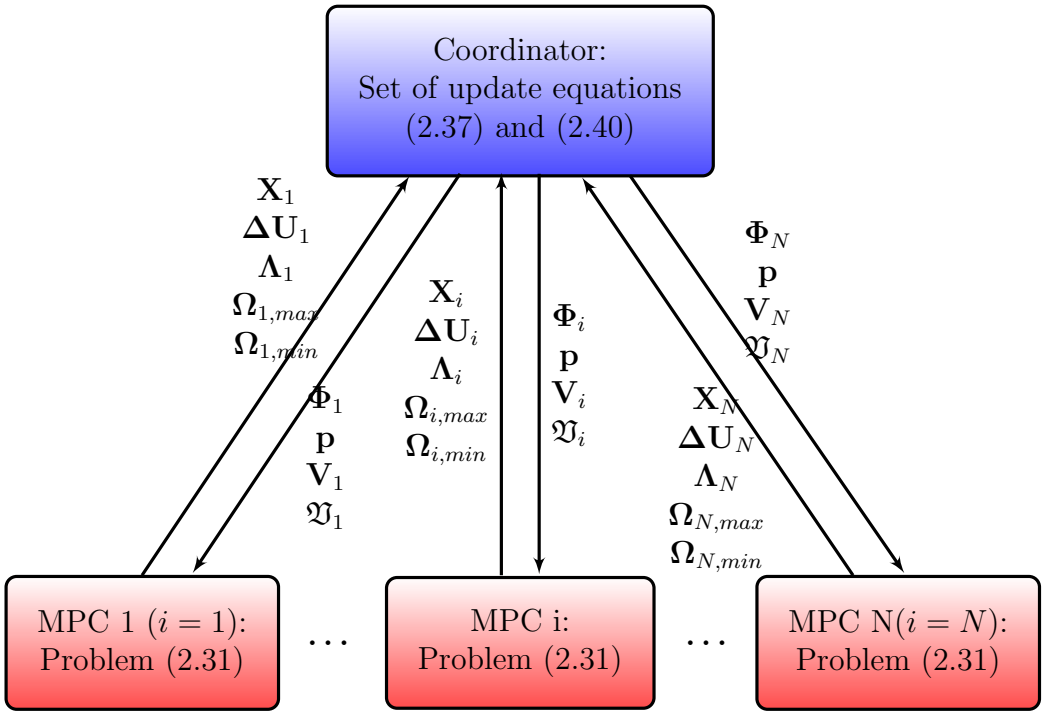


Figure A.2: Information flow in the IPC-DMPC network

A.3 Information Flow in the MPMC Method

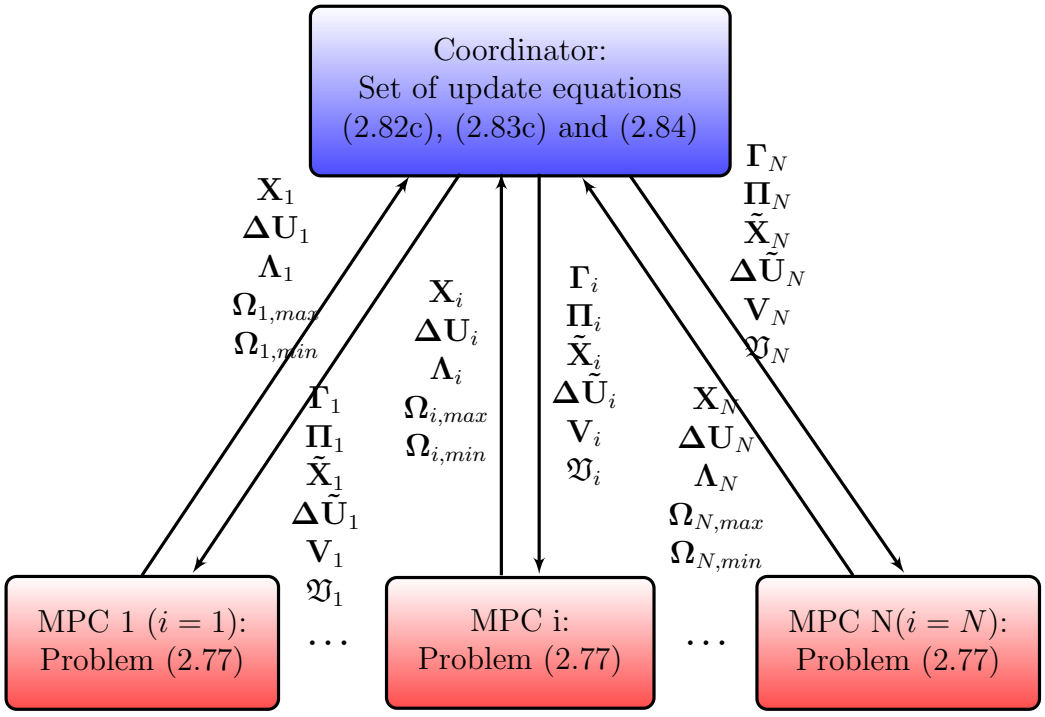


Figure A.3: Information flow in the MPMC-DMPC network

Appendix B

B.1 Coefficient Matrix in the GC Method

To show how the matrix $\mathbf{\Theta}_i$ is built, a simple example with three subsystems ($N = 3$) is used. Prediction and control horizons are chosen to be 5 and 2, respectively. Using (2.11a), the matrix $\mathbf{\Theta}_i$ can easily be constructed as:

$$\mathbf{\Theta_1} = \left[\begin{array}{ccccc:cc:ccccc}
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\
\hdashline
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
-\mathbf{A}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & -\mathbf{B}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & -\mathbf{A}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & -\mathbf{B}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & -\mathbf{A}_{21} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & -\mathbf{B}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{21} & \mathbf{0} & -\mathbf{B}_{21} & -\mathbf{B}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hdashline
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
-\mathbf{A}_{31} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{31} & -\mathbf{B}_{31} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & -\mathbf{A}_{31} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{31} & -\mathbf{B}_{31} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & -\mathbf{A}_{31} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{31} & -\mathbf{B}_{31} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{31} & \mathbf{0} & -\mathbf{B}_{31} & -\mathbf{B}_{31} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}
\end{array}\right] \tag{B.1}$$

$$\boldsymbol{\Theta_2} = \left[\begin{array}{ccccc:cc:ccccc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{A}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{12} & -\mathbf{B}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{A}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{12} & -\mathbf{B}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{A}_{12} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{12} & -\mathbf{B}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{12} & \mathbf{0} & -\mathbf{B}_{12} & -\mathbf{B}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \hdashline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{32} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{A}_{32} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{32} & -\mathbf{B}_{32} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{A}_{32} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{32} & -\mathbf{B}_{32} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{A}_{32} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{32} & -\mathbf{B}_{32} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{32} & \mathbf{0} & -\mathbf{B}_{32} & -\mathbf{B}_{32} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right] \tag{B.2}$$

$$\boldsymbol{\Theta_3} = \left[\begin{array}{ccccc:cc:ccccc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{A}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{13} & -\mathbf{B}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{A}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{13} & -\mathbf{B}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{A}_{13} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{13} & -\mathbf{B}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{13} & \mathbf{0} & -\mathbf{B}_{13} & -\mathbf{B}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{A}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{23} & -\mathbf{B}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{A}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{23} & -\mathbf{B}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{A}_{23} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{23} & -\mathbf{B}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{23} & \mathbf{0} & -\mathbf{B}_{23} & -\mathbf{B}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{array}\right] \tag{B.3}$$

B.2 Coefficient Matrix in the IPC Method

As is discussed in section 2.4, the interaction coefficient matrix used in the distributed controllers, are the same as $\boldsymbol{\Theta}_i$ used by the CDMPC in the GC method; however, for implementation purposes, lower dimensional matrix $\boldsymbol{\Phi}_i$ can be used instead. The

only difference in the structure of this matrix in comparison with $\mathbf{\Theta}_i$, is that $\mathbf{\Phi}_i$ does not have the columns corresponding to $\mathbf{V}_i$. Therefore, using the same example as in section B.1, the $\mathbf{\Phi}_i$ is written as:

$$\mathbf{\Phi_1} = \left[\begin{array}{ccccc:cc}
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hdashline
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & \mathbf{0} \\
-\mathbf{A}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & -\mathbf{B}_{21} \\
\mathbf{0} & -\mathbf{A}_{21} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & -\mathbf{B}_{21} \\
\mathbf{0} & \mathbf{0} & -\mathbf{A}_{21} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & -\mathbf{B}_{21} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{21} & \mathbf{0} & -\mathbf{B}_{21} & -\mathbf{B}_{21} \\
\hdashline
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{21} & \mathbf{0} \\
-\mathbf{A}_{31} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{31} & -\mathbf{B}_{31} \\
\mathbf{0} & -\mathbf{A}_{31} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{31} & -\mathbf{B}_{31} \\
\mathbf{0} & \mathbf{0} & -\mathbf{A}_{31} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{31} & -\mathbf{B}_{31} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{31} & \mathbf{0} & -\mathbf{B}_{31} & -\mathbf{B}_{31}
\end{array}\right] \tag{B.4}$$

$$\mathbf{\Phi_2} = \left[\begin{array}{ccccc:cc}
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{12} & \mathbf{0} \\
-\mathbf{A}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{12} & -\mathbf{B}_{12} \\
\mathbf{0} & -\mathbf{A}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{12} & -\mathbf{B}_{12} \\
\mathbf{0} & \mathbf{0} & -\mathbf{A}_{12} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{12} & -\mathbf{B}_{12} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{12} & \mathbf{0} & -\mathbf{B}_{12} & -\mathbf{B}_{12} \\
\hdashline
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hdashline
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{32} & \mathbf{0} \\
-\mathbf{A}_{32} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{32} & -\mathbf{B}_{32} \\
\mathbf{0} & -\mathbf{A}_{32} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{32} & -\mathbf{B}_{32} \\
\mathbf{0} & \mathbf{0} & -\mathbf{A}_{32} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{32} & -\mathbf{B}_{32} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{32} & \mathbf{0} & -\mathbf{B}_{32} & -\mathbf{B}_{32}
\end{array}\right] \tag{B.5}$$

$$\mathbf{\Phi_3} = \left[\begin{array}{ccccc:cc} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{13} & \mathbf{0} \\ -\mathbf{A}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{13} & -\mathbf{B}_{13} \\ \mathbf{0} & -\mathbf{A}_{13} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{13} & -\mathbf{B}_{13} \\ \mathbf{0} & \mathbf{0} & -\mathbf{A}_{13} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{13} & -\mathbf{B}_{13} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{13} & \mathbf{0} & -\mathbf{B}_{13} & -\mathbf{B}_{13} \\ \hdashline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{23} & \mathbf{0} \\ -\mathbf{A}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{23} & -\mathbf{B}_{23} \\ \mathbf{0} & -\mathbf{A}_{23} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{23} & -\mathbf{B}_{23} \\ \mathbf{0} & \mathbf{0} & -\mathbf{A}_{23} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{23} & -\mathbf{B}_{23} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{23} & \mathbf{0} & -\mathbf{B}_{23} & -\mathbf{B}_{23} \\ \hdashline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{array}\right] \tag{B.6}$$

B.3 Compact Form of Problem (2.32)

B.3.1 Equality Constraints (2.31b)

First the local process model (2.31b) is written over the prediction and control horizons H_p and H_u:

$$\mathbf{x}_i(k+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{\Delta u}_i(k|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \sum_{\substack{j=1 \\ j\neq i}}^{N} \mathbf{A}_{ij}\mathbf{x}_j(k) + \sum_{\substack{j=1 \\ j\neq i}}^{N} \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \mathbf{v}_i(k|k)$$

$$\mathbf{x}_i(k+2|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+1|k) + \mathbf{B}_{ii}\mathbf{\Delta u}_i(k+1|k) + \mathbf{B}_{ii}\mathbf{\Delta u}_i(k|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \sum_{\substack{j=1 \\ j\neq i}}^{N} \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \mathbf{v}_i(k+1|k)$$

$$\vdots$$

$$\begin{aligned}
\mathbf{x}_i(k+H_u|k) &= \mathbf{A}_{ii}\mathbf{x}_i(k+H_u-1|k) + \sum_{a=0}^{H_u-1}\mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \\
&\quad \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1) + \\
&\quad \mathbf{v}_i(k+H_u-1|k) \\
&\vdots \\
\mathbf{x}_i(k+H_p|k) &= \mathbf{A}_{ii}\mathbf{x}_i(k+H_p-1|k) + \sum_{a=0}^{H_u-1}\mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \\
&\quad \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1) + \\
&\quad \mathbf{v}_i(k+H_p-1|k)
\end{aligned} \tag{B.7}$$

In the next step, the terms in (B.7) containing the *unknown variables*, are moved to the left hand-side and then written in the following matrix form:

$$\underbrace{\begin{bmatrix}
\mathbf{I} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
-\mathbf{A}_{ii} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \dots & \mathbf{0} & -\mathbf{A}_{ii} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{ii} & \mathbf{I}
\end{bmatrix}}_{\hat{\mathbb{A}}_{ii}}
\underbrace{\begin{bmatrix}
\mathbf{x}_i(k+1|k) \\ \mathbf{x}_i(k+2|k) \\ \vdots \\ \mathbf{x}_i(k+H_u|k) \\ \vdots \\ \mathbf{x}_i(k+H_p|k)
\end{bmatrix}}_{\mathbf{X}_i} +$$

$$\underbrace{\begin{bmatrix}
-\mathbf{B}_{ii} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
-\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
\vdots & & \dots & & & \vdots \\
-\mathbf{B}_{ii} & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \dots & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} \\
\vdots & & \dots & & & \vdots \\
-\mathbf{B}_{ii} & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \dots & -\mathbf{B}_{ii} & -\mathbf{B}_{ii}
\end{bmatrix}}_{\hat{\mathbb{B}}_{ii}}
\underbrace{\begin{bmatrix}
\boldsymbol{\Delta}\mathbf{u}_i(k|k) \\ \boldsymbol{\Delta}\mathbf{u}_i(k+1|k) \\ \vdots \\ \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k)
\end{bmatrix}}_{\boldsymbol{\Delta}\mathbf{U}_i} -
\underbrace{\begin{bmatrix}
\mathbf{v}_i(k|k) \\ \mathbf{v}_i(k+1|k) \\ \vdots \\ \mathbf{v}_i(k+H_u-1|k) \\ \vdots \\ \mathbf{v}_i(k+H_p-1|k)
\end{bmatrix}}_{\mathbf{V}_i} =$$

$$\underbrace{\begin{bmatrix}\mathbf{A}_{ii}\\ \mathbf{0}\\ \mathbf{0}\\ \vdots\\ \\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}}_{\bar{\mathbb{A}}_{ii}}\mathbf{x}_i(k)+\underbrace{\begin{bmatrix}\mathbf{B}_{ii}\\ \mathbf{B}_{ii}\\ \mathbf{B}_{ii}\\ \vdots\\ \\ \mathbf{B}_{ii}\\ \mathbf{B}_{ii}\end{bmatrix}}_{\bar{\mathbb{B}}_{ii}}\mathbf{u}_i(k-1)+\sum_{\substack{j=1\\ j\neq i}}^{N}\underbrace{\begin{bmatrix}\mathbf{A}_{ij}\\ \mathbf{0}\\ \mathbf{0}\\ \vdots\\ \\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}}_{\bar{\mathbb{A}}_{ij}}\mathbf{x}_j(k)+\sum_{\substack{j=1\\ j\neq i}}^{N}\underbrace{\begin{bmatrix}\mathbf{B}_{ij}\\ \mathbf{B}_{ij}\\ \mathbf{B}_{ij}\\ \vdots\\ \\ \mathbf{B}_{ij}\\ \mathbf{B}_{ij}\end{bmatrix}}_{\bar{\mathbb{B}}_{ij}}\mathbf{u}_j(k-1) \tag{B.8}$$

B.3.2 Inequality Constraints (2.31c)

Writing the inequality constraints on the outputs (2.31c), over the control and prediction horizons, results in:

$$\underbrace{\begin{bmatrix}\mathbf{y}_i^{min}(k+1)\\ \vdots\\ \mathbf{y}_i^{min}(k+H_p)\end{bmatrix}}_{\mathbf{Y}_i^{min}}\leq\underbrace{\begin{bmatrix}\mathbf{C}_{ii} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0}\\ \mathbf{0} & \mathbf{C}_{ii} & \mathbf{0} & \dots & \mathbf{0}\\ \vdots & & \ddots & & \vdots\\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{C}_{ii}\end{bmatrix}}_{\mathbb{C}_{ii}}\underbrace{\begin{bmatrix}\mathbf{x}_i(k+1|k)\\ \vdots\\ \mathbf{x}_i(k+H_p|k)\end{bmatrix}}_{\mathbf{X}_i}\leq\underbrace{\begin{bmatrix}\mathbf{y}_i^{max}(k+1)\\ \vdots\\ \mathbf{y}_i^{max}(k+H_p)\end{bmatrix}}_{\mathbf{Y}_i^{max}} \tag{B.9}$$

For the inequality constraints on the manipulated variables, first $\mathbf{u}_i(k+b|k)$ is written in terms of $\mathbf{\Delta u}_i(k+b|k)$ using (2.5):

$$\begin{bmatrix}\mathbf{u}_i^{min}(k)\\ \mathbf{u}_i^{min}(k+1)\\ \vdots\\ \mathbf{u}_i^{min}(k+H_u-1)\end{bmatrix}\leq$$
$$\begin{bmatrix}\mathbf{u}_i(k-1)+\mathbf{\Delta u}_i(k|k)\\ \mathbf{u}_i(k-1)+\mathbf{\Delta u}_i(k|k)+\mathbf{\Delta u}_i(k+1|k)\\ \vdots\\ \mathbf{u}_i(k-1)+\mathbf{\Delta u}_i(k|k)+\mathbf{\Delta u}_i(k+1|k)+\dots+\mathbf{\Delta u}_i(k+H_u-1|k)\end{bmatrix}\leq \tag{B.10}$$
$$\begin{bmatrix}\mathbf{u}_i^{max}(k)\\ \mathbf{u}_i^{max}(k+1)\\ \vdots\\ \mathbf{u}_i^{max}(k+H_u-1)\end{bmatrix}$$

The set of inequalities (B.10) are re-arranged as below:

$$\underbrace{\begin{bmatrix} \mathbf{u}_i^{min}(k) - \mathbf{u}_i(k-1) \\ \mathbf{u}_i^{min}(k+1) - \mathbf{u}_i(k-1) \\ \vdots \\ \mathbf{u}_i^{min}(k+H_u-1) - \mathbf{u}_i(k-1) \end{bmatrix}}_{\mathfrak{u}_{i*}^{min}} \leq \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{0} & & \dots & \mathbf{0} \\ \mathbf{I} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & & \ddots & & \vdots \\ \mathbf{I} & \mathbf{I} & \mathbf{I} & \dots & \mathbf{I} \end{bmatrix}}_{\mathbb{U}_*} \underbrace{\begin{bmatrix} \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\ \boldsymbol{\Delta}\mathbf{u}_i(k+1|k) \\ \vdots \\ \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k) \end{bmatrix}}_{\boldsymbol{\Delta}\mathbf{U}_i} \leq \underbrace{\begin{bmatrix} \mathbf{u}_i^{max}(k) - \mathbf{u}_i(k-1) \\ \mathbf{u}_i^{max}(k+1) - \mathbf{u}_i(k-1) \\ \vdots \\ \mathbf{u}_i^{max}(k+H_u-1) - \mathbf{u}_i(k-1) \end{bmatrix}}_{\mathfrak{u}_{i*}^{max}} \tag{B.11}$$

Also, the inequality constraints on the control input changes in (2.31c), are written over the control horizon H_u:

$$\underbrace{\begin{bmatrix} \boldsymbol{\Delta}\mathbf{u}_i^{min}(k) \\ \vdots \\ \boldsymbol{\Delta}\mathbf{u}_i^{min}(k+H_u-1) \end{bmatrix}}_{\mathfrak{u}_{i**}^{min}} \leq \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{0} & & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & & \ddots & & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{I} \end{bmatrix}}_{\mathbb{U}_i^{**}} \underbrace{\begin{bmatrix} \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\ \vdots \\ \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k) \end{bmatrix}}_{\boldsymbol{\Delta}\mathbf{U}_i} \leq \underbrace{\begin{bmatrix} \boldsymbol{\Delta}\mathbf{u}_i^{max}(k) \\ \vdots \\ \boldsymbol{\Delta}\mathbf{u}_i^{max}(k+H_u-1) \end{bmatrix}}_{\mathfrak{u}_{i**}^{max}} \tag{B.12}$$

Finally aggregate of (B.11) and (B.12), results in the following compact form:

$$\underbrace{\begin{bmatrix} \mathbf{u}_{i*}^{min} \\ \mathbf{u}_{i**}^{min} \end{bmatrix}}_{\mathfrak{U}_i^{min}} \leq \underbrace{\begin{bmatrix} \mathbb{U}_i^{*} & \mathbf{0} \\ \mathbf{0} & \mathbb{U}_i^{**} \end{bmatrix}}_{\mathbb{U}_i} \boldsymbol{\Delta}\mathbf{U}_i \leq \underbrace{\begin{bmatrix} \mathbf{u}_{i*}^{max} \\ \mathbf{u}_{i**}^{max} \end{bmatrix}}_{\mathfrak{U}_i^{max}} \tag{B.13}$$

B.4 Compact Form of Problem (2.78)

B.4.1 Equality Constraints (2.78b)

First the equality constraints (2.75b) and (2.75c), are combined:

$$\begin{aligned}\mathbf{x}_i(k+l+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}\left[\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{u}_i(k-1)\right] + \\ \sum_{\substack{j=1\\ j\neq i}}^{N}\left[\alpha\mathbf{A}_{ij}\mathbf{x}_j(k+l|k) + \mathbf{B}_{ij}\mathbf{u}_j(k-1)\right] + \\ \sum_{\substack{j=1\\ j\neq i}}^{N}\left[\beta\mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+l|k) + \mathbf{B}_{ij}\sum_{a=0}^{l}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+a|k)\right]\end{aligned} \tag{B.14}$$

Then the local process model (B.14), is expanded over the prediction and control horizons:

$$\begin{aligned}\mathbf{x}_i(k+1|k) &= \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \\ &\quad \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{A}_{ij}\mathbf{x}_j(k) + \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1) + \\ &\quad \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k) \\ \mathbf{x}_i(k+2|k) &= \mathbf{A}_{ii}\mathbf{x}_i(k+1|k) + \mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+1|k) + \mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \\ &\quad \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1) + \\ &\quad \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+1|k) + \sum_{\substack{j=1\\ j\neq i}}^{N}[\mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+1|k) + \mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k)] \\ &\vdots\end{aligned}$$

$$\begin{aligned}
\mathbf{x}_i(k+H_u|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+H_u-1|k) + \sum_{a=0}^{H_u-1}\mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \\
\sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1) + \\
\sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+H_u-1|k) + \sum_{\substack{j=1\\ j\neq i}}^{N}\sum_{a=0}^{H_u-1}\mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+a|k) \\
\vdots \\
\mathbf{x}_i(k+H_p|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+H_p-1|k) + \sum_{a=0}^{H_u-1}\mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \\
\sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1) + \\
\sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+H_p-1|k) + \sum_{\substack{j=1\\ j\neq i}}^{N}\sum_{a=0}^{H_u-1}\mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+a|k)
\end{aligned} \tag{B.15}$$

Now the terms in (B.15), which contain the *unknown variables*, are moved to the left hand-side. The unknown variables are the local decision variables as well as the pseudo-variables that are determined by the coordinator. Thus:

$$\underbrace{\begin{bmatrix}
\mathbf{I} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
-\mathbf{A}_{ii} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \dots & \mathbf{0} & -\mathbf{A}_{ii} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{ii} & \mathbf{I}
\end{bmatrix}}_{\hat{\mathbb{A}}_{ii}}
\underbrace{\begin{bmatrix}
\mathbf{x}_i(k+1|k) \\ \mathbf{x}_i(k+2|k) \\ \vdots \\ \mathbf{x}_i(k+H_u|k) \\ \vdots \\ \mathbf{x}_i(k+H_p|k)
\end{bmatrix}}_{\mathbf{x}_i} +$$

$$\underbrace{\begin{bmatrix}
-\mathbf{B}_{ii} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
-\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
\vdots & & \dots & & & \vdots \\
-\mathbf{B}_{ii} & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \dots & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} \\
\vdots & & \dots & & & \vdots \\
-\mathbf{B}_{ii} & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \dots & -\mathbf{B}_{ii} & -\mathbf{B}_{ii}
\end{bmatrix}}_{\hat{\mathbb{B}}_{ii}}
\underbrace{\begin{bmatrix}
\boldsymbol{\Delta}\mathbf{u}_i(k|k) \\ \boldsymbol{\Delta}\mathbf{u}_i(k+1|k) \\ \vdots \\ \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k)
\end{bmatrix}}_{\boldsymbol{\Delta}\mathbf{U}_i} +$$

$$\sum_{\substack{j=1\\j\neq i}}^{N} \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ -\mathbf{A}_{ij} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{A}_{ij} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & & & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{ij} \end{bmatrix}}_{\hat{\mathbb{A}}_{ij}} \underbrace{\begin{bmatrix} \tilde{\mathbf{x}}_j(k+1|k) \\ \tilde{\mathbf{x}}_j(k+2|k) \\ \tilde{\mathbf{x}}_j(k+3|k) \\ \vdots \\ \tilde{\mathbf{x}}_j(k+H_p-1|k) \end{bmatrix}}_{\tilde{\mathbf{x}}_j} +$$

$$\sum_{\substack{j=1\\j\neq i}}^{N} \underbrace{\begin{bmatrix} -\mathbf{B}_{ij} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \dots & & & \vdots \\ -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & \dots & -\mathbf{B}_{ij} & -\mathbf{B}_{ij} \\ \vdots & & \dots & & & \vdots \\ -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & \dots & -\mathbf{B}_{ij} & -\mathbf{B}_{ij} \end{bmatrix}}_{\hat{\mathbb{B}}_{ij}} \underbrace{\begin{bmatrix} \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k) \\ \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+1|k) \\ \vdots \\ \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k) \end{bmatrix}}_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_j} =$$

$$\underbrace{\begin{bmatrix} \mathbf{A}_{ii} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{\bar{\mathbb{A}}_{ii}} \mathbf{x}_i(k) + \underbrace{\begin{bmatrix} \mathbf{B}_{ii} \\ \mathbf{B}_{ii} \\ \mathbf{B}_{ii} \\ \vdots \\ \mathbf{B}_{ii} \\ \mathbf{B}_{ii} \end{bmatrix}}_{\bar{\mathbb{B}}_{ii}} \mathbf{u}_i(k-1) + \sum_{\substack{j=1\\j\neq i}}^{N} \underbrace{\begin{bmatrix} \mathbf{A}_{ij} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{\bar{\mathbb{A}}_{ij}} \mathbf{x}_j(k) + \sum_{\substack{j=1\\j\neq i}}^{N} \underbrace{\begin{bmatrix} \mathbf{B}_{ij} \\ \mathbf{B}_{ij} \\ \mathbf{B}_{ij} \\ \vdots \\ \mathbf{B}_{ij} \\ \mathbf{B}_{ij} \end{bmatrix}}_{\bar{\mathbb{B}}_{ij}} \mathbf{u}_j(k-1) \quad \text{(B.16)}$$

B.4.2 Inequality Constraints (2.78c)

The matrices used in the inequality constraints (2.78c), are the same as those in (2.32c). Therefore, they are obtained according to section B.3.2.

Appendix C

C.1 Compact Form of the Centralized Prediction Model (3.3)

In obtaining (3.3), process model (3.1) is used for the overall system, as below:

$$\mathbf{x}(k+l+1|k) = \mathbf{A}\mathbf{x}(k+l|k) + \mathbf{B}\sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}(k+a|k) + \mathbf{u}(k-1) + \mathbf{D}\mathbf{d}(k+l|k) \quad \text{(C.1)}$$

where $\mathbf{x} \triangleq \left[\mathbf{x}_1^T, \ \ldots \ , \mathbf{x}_N^T\right]^T$, $\boldsymbol{\Delta}\mathbf{u} \triangleq \left[\boldsymbol{\Delta}\mathbf{u}_1^T, \ \ldots \ , \boldsymbol{\Delta}\mathbf{u}_N^T\right]^T$ and $\mathbf{d} \triangleq \left[\mathbf{d}_1^T, \ \ldots \ , \mathbf{d}_N^T\right]^T$. The centralized matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{D}$, have the following block-wise structure:

$$\mathbf{A} = \left[\begin{array}{c:c:c:c:c} \mathbf{A}_{11} & \ldots & \mathbf{A}_{1j} & \ldots & \mathbf{A}_{1N} \\ \hdashline \vdots & & \vdots & & \vdots \\ \hdashline \mathbf{A}_{i1} & \ldots & \mathbf{A}_{ii} & \ldots & \mathbf{A}_{iN} \\ \hdashline \vdots & & \vdots & & \vdots \\ \hdashline \mathbf{A}_{N1} & \ldots & \mathbf{A}_{Nj} & \ldots & \mathbf{A}_{NN} \end{array}\right], \quad \mathbf{B} = \left[\begin{array}{c:c:c:c:c} \mathbf{B}_{11} & \ldots & \mathbf{B}_{1j} & \ldots & \mathbf{B}_{1N} \\ \hdashline \vdots & & \vdots & & \vdots \\ \hdashline \mathbf{B}_{i1} & \ldots & \mathbf{B}_{ii} & \ldots & \mathbf{B}_{iN} \\ \hdashline \vdots & & \vdots & & \vdots \\ \hdashline \mathbf{B}_{N1} & \ldots & \mathbf{B}_{Nj} & \ldots & \mathbf{B}_{NN} \end{array}\right]$$

and (C.2)

$$\mathbf{D} = \left[\begin{array}{c:c:c:c:c} \mathbf{D}_{11} & \ldots & \mathbf{D}_{1j} & \ldots & \mathbf{D}_{1N} \\ \hdashline \vdots & & \vdots & & \vdots \\ \hdashline \mathbf{D}_{i1} & \ldots & \mathbf{D}_{ii} & \ldots & \mathbf{D}_{iN} \\ \hdashline \vdots & & \vdots & & \vdots \\ \hdashline \mathbf{D}_{N1} & \ldots & \mathbf{D}_{Nj} & \ldots & \mathbf{D}_{NN} \end{array}\right]$$

Now the plant model (C.1), is written over the entire prediction and control horizons, as below:

$$
\begin{aligned}
&\mathbf{x}(k+1|k) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\boldsymbol{\Delta}\mathbf{u}(k|k) + \mathbf{B}\mathbf{u}(k-1) + \mathbf{D}\mathbf{d}(k|k)\\
&\mathbf{x}(k+2|k) = \mathbf{A}\mathbf{x}(k+1|k) + \mathbf{B}\boldsymbol{\Delta}\mathbf{u}(k+1|k) + \mathbf{B}\boldsymbol{\Delta}\mathbf{u}(k|k) + \mathbf{B}\mathbf{u}(k-1) + \mathbf{D}\mathbf{d}(k+1|k)\\
&\vdots\\
&\mathbf{x}(k+H_u|k) = \mathbf{A}\mathbf{x}(k+H_u-1|k) + \sum_{a=0}^{H_u-1} \mathbf{B}\boldsymbol{\Delta}\mathbf{u}(k+a|k) + \mathbf{B}\mathbf{u}(k-1) + \mathbf{D}\mathbf{d}(k+H_u-1|k)\\
&\vdots\\
&\mathbf{x}(k+H_p|k) = \mathbf{A}\mathbf{x}(k+H_p-1|k) + \sum_{a=0}^{H_u-1} \mathbf{B}\boldsymbol{\Delta}\mathbf{u}(k+a|k) + \mathbf{B}\mathbf{u}(k-1) + \mathbf{D}\mathbf{d}(k+H_p-1|k)
\end{aligned}
\tag{C.3}
$$

After moving the unknown terms to the left-hand side and the known terms to the right-hand side, (C.3) becomes:

$$
\underbrace{\begin{bmatrix}
\mathbf{I} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
-\mathbf{A} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\
\vdots & & & \ddots & & & & \vdots\\
\mathbf{0} & \dots & \mathbf{0} & -\mathbf{A} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0}\\
\vdots & & & \ddots & & & & \vdots\\
\mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A} & \mathbf{I}
\end{bmatrix}}_{\hat{\mathbb{A}}}
\underbrace{\begin{bmatrix}
\mathbf{x}(k+1|k)\\
\mathbf{x}(k+2|k)\\
\vdots\\
\mathbf{x}(k+H_u|k)\\
\vdots\\
\mathbf{x}(k+H_p|k)
\end{bmatrix}}_{\mathbf{X}} +
$$

$$
\underbrace{\begin{bmatrix}
-\mathbf{B} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0}\\
-\mathbf{B} & -\mathbf{B} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0}\\
\vdots & & \dots & & & \vdots\\
-\mathbf{B} & -\mathbf{B} & -\mathbf{B} & \dots & -\mathbf{B} & -\mathbf{B}\\
\vdots & & \dots & & & \vdots\\
-\mathbf{B} & -\mathbf{B} & -\mathbf{B} & \dots & -\mathbf{B} & -\mathbf{B}
\end{bmatrix}}_{\hat{\mathbb{B}}}
\underbrace{\begin{bmatrix}
\boldsymbol{\Delta}\mathbf{u}(k|k)\\
\boldsymbol{\Delta}\mathbf{u}(k+1|k)\\
\vdots\\
\boldsymbol{\Delta}\mathbf{u}(k+H_u-1|k)
\end{bmatrix}}_{\boldsymbol{\Delta}\mathbf{U}}
$$

$$
+ \underbrace{\begin{bmatrix}
-\mathbf{D} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0}\\
\mathbf{0} & -\mathbf{D} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0}\\
\vdots & & \ddots & & & \vdots\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & -\mathbf{D}
\end{bmatrix}}_{\hat{\mathbb{D}}}
\underbrace{\begin{bmatrix}
\mathbf{d}(k|k)\\
\mathbf{d}(k+1|k)\\
\vdots\\
\mathbf{d}(k+H_p-1|k)
\end{bmatrix}}_{\mathfrak{D}}
$$

$$\underbrace{\begin{bmatrix}\mathbf{A}\\ \mathbf{0}\\ \mathbf{0}\\ \vdots\\ \mathbf{0}\\ \mathbf{0}\end{bmatrix}}_{\bar{\mathbb{A}}}\mathbf{x}(k)+\underbrace{\begin{bmatrix}\mathbf{B}\\ \mathbf{B}\\ \mathbf{B}\\ \vdots\\ \mathbf{B}\\ \mathbf{B}\end{bmatrix}}_{\bar{\mathbb{B}}}\mathbf{u}(k-1) \tag{C.4}$$

In addition to (C.3), the centralized prediction model can also be expressed in terms of local process units, as follows:

$$\begin{aligned}\mathbf{x}_i(k+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k)+\mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k|k)+\mathbf{B}_{ii}\mathbf{u}_i(k-1)+\\ \mathbf{D}_{ii}\mathbf{d}_i(k|k)+\sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{D}_{ij}\mathbf{d}_j(k|k)\\ \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{A}_{ij}\mathbf{x}_j(k)+\sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1)+\\ \mathbf{v}_i(k|k)\end{aligned}$$

$$\begin{aligned}\mathbf{x}_i(k+2|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+1|k)+\mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+1|k)+\mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k|k)+\mathbf{B}_{ii}\mathbf{u}_i(k-1)+\\ \mathbf{D}_{ii}\mathbf{d}_i(k+1|k)+\sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{D}_{ij}\mathbf{d}_j(k+1|k)\\ \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1)+\\ \mathbf{v}_i(k+1|k)\end{aligned}$$

$\vdots$

$$\begin{aligned}
\mathbf{x}_i(k+H_u|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+H_u-1|k) + \sum_{a=0}^{H_u-1}\mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \\
\mathbf{D}_{ii}\mathbf{d}_i(k+H_u-1|k) + \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{D}_{ij}\mathbf{d}_j(k+H_u-1|k) \\
\sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1) + \\
\mathbf{v}_i(k+H_u-1|k)
\end{aligned}$$

$$\vdots$$

$$\begin{aligned}
\mathbf{x}_i(k+H_p|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+H_p-1|k) + \sum_{a=0}^{H_u-1}\mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) + \\
\mathbf{D}_{ii}\mathbf{d}_i(k+H_p-1|k) + \sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{D}_{ij}\mathbf{d}_j(k+H_p-1|k) \\
\sum_{\substack{j=1\\ j\neq i}}^{N}\mathbf{B}_{ij}\mathbf{u}_j(k-1) + \\
\mathbf{v}_i(k+H_p-1|k)
\end{aligned} \tag{C.5}$$

After re-arranging the terms in (C.5), the following compact form is obtained:

$$\underbrace{\begin{bmatrix}
\mathbf{I} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
-\mathbf{A}_{ii} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \dots & \mathbf{0} & -\mathbf{A}_{ii} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{ii} & \mathbf{I}
\end{bmatrix}}_{\hat{\mathbb{A}}_{ii}}
\underbrace{\begin{bmatrix}
\mathbf{x}_i(k+1|k) \\
\mathbf{x}_i(k+2|k) \\
\vdots \\
\mathbf{x}_i(k+H_u|k) \\
\vdots \\
\mathbf{x}_i(k+H_p|k)
\end{bmatrix}}_{\mathbf{x}_i} +$$

$$\underbrace{\begin{bmatrix}
-\mathbf{B}_{ii} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
-\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
\vdots & & \dots & & & \vdots \\
-\mathbf{B}_{ii} & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \dots & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} \\
\vdots & & \dots & & & \vdots \\
-\mathbf{B}_{ii} & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \dots & -\mathbf{B}_{ii} & -\mathbf{B}_{ii}
\end{bmatrix}}_{\hat{\mathbb{B}}_{ii}}
\underbrace{\begin{bmatrix}
\boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
\boldsymbol{\Delta}\mathbf{u}_i(k+1|k) \\
\vdots \\
\boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k)
\end{bmatrix}}_{\boldsymbol{\Delta}\mathbf{U}_i}$$

$$+\underbrace{\begin{bmatrix} -\mathbf{D}_{ii} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{D}_{ii} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & & & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & -\mathbf{D}_{ii} \end{bmatrix}}_{\hat{\mathbb{D}}_{ii}} \underbrace{\begin{bmatrix} \mathbf{d}_i(k|k) \\ \mathbf{d}_i(k+1|k) \\ \vdots \\ \mathbf{d}_i(k+H_p-1|k) \end{bmatrix}}_{\mathfrak{D}_i}$$

$$+\sum_{\substack{j=1 \\ j\neq i}}^{N} \underbrace{\begin{bmatrix} -\mathbf{D}_{ij} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{D}_{ij} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & & & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & -\mathbf{D}_{ij} \end{bmatrix}}_{\hat{\mathbb{D}}_{ij}} \underbrace{\begin{bmatrix} \mathbf{d}_j(k|k) \\ \mathbf{d}_j(k+1|k) \\ \vdots \\ \mathbf{d}_j(k+H_p-1|k) \end{bmatrix}}_{\mathfrak{D}_j}$$

$$-\underbrace{\begin{bmatrix} \mathbf{v}_i(k|k) \\ \mathbf{v}_i(k+1|k) \\ \vdots \\ \mathbf{v}_i(k+H_u-1|k) \\ \vdots \\ \mathbf{v}_i(k+H_p-1|k) \end{bmatrix}}_{\mathbf{v}_i} =$$

$$\underbrace{\begin{bmatrix} \mathbf{A}_{ii} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{\bar{\mathbb{A}}_{ii}} \mathbf{x}_i(k) + \underbrace{\begin{bmatrix} \mathbf{B}_{ii} \\ \mathbf{B}_{ii} \\ \mathbf{B}_{ii} \\ \vdots \\ \\ \mathbf{B}_{ii} \\ \mathbf{B}_{ii} \end{bmatrix}}_{\bar{\mathbb{B}}_{ii}} \mathbf{u}_i(k-1) + \sum_{\substack{j=1 \\ j\neq i}}^{N} \underbrace{\begin{bmatrix} \mathbf{A}_{ij} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{\bar{\mathbb{A}}_{ij}} \mathbf{x}_j(k) + \sum_{\substack{j=1 \\ j\neq i}}^{N} \underbrace{\begin{bmatrix} \mathbf{B}_{ij} \\ \mathbf{B}_{ij} \\ \mathbf{B}_{ij} \\ \vdots \\ \\ \mathbf{B}_{ij} \\ \mathbf{B}_{ij} \end{bmatrix}}_{\bar{\mathbb{B}}_{ij}} \mathbf{u}_j(k-1) \qquad \text{(C.6)}$$

C.2 Distributed Equivalence of the matrix $\mathbb{G}$

Through the following simple example, the elements needed to be taken into account in localizing vectors of the form $\bar{\mathbf{z}} = \mathbf{WVz}$, is explained. Suppose that $\mathbf{W}$, $\mathbf{V}$ and $\mathbf{z}$ are defined as below:

$$\mathbf{W} = \left[\begin{array}{c:c} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \hdashline \mathbf{W}_{21} & \mathbf{W}_{22} \end{array}\right] \quad , \quad \mathbf{V} = \left[\begin{array}{c:c} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \hdashline \mathbf{V}_{21} & \mathbf{V}_{22} \end{array}\right] \quad \text{and} \quad \mathbf{z} = \left[\begin{array}{c} \mathbf{z}_1 \\ \hdashline \mathbf{z}_2 \end{array}\right] \qquad \text{(C.7)}$$

The product **WVz** becomes:

$$\left[\begin{array}{c}\bar{\mathbf{z}}_1\\ \hdashline \bar{\mathbf{z}}_2\end{array}\right] = \left[\begin{array}{c:c}\mathbf{W}_{11} & \mathbf{W}_{12}\\ \hdashline \mathbf{W}_{21} & \mathbf{W}_{22}\end{array}\right]\left[\begin{array}{c:c}\mathbf{V}_{11} & \mathbf{V}_{21}\\ \hdashline \mathbf{V}_{12} & \mathbf{V}_{22}\end{array}\right]\left[\begin{array}{c}\mathbf{z}_1\\ \hdashline \mathbf{z}_2\end{array}\right]$$

$$= \left[\begin{array}{c:c}\mathbf{W}_{11}\mathbf{V}_{11}+\mathbf{W}_{12}\mathbf{V}_{12} & \mathbf{W}_{11}\mathbf{V}_{21}+\mathbf{W}_{12}\mathbf{V}_{22}\\ \hdashline \mathbf{W}_{21}\mathbf{V}_{11}+\mathbf{W}_{22}\mathbf{V}_{12} & \mathbf{W}_{21}\mathbf{V}_{21}+\mathbf{W}_{22}\mathbf{V}_{22}\end{array}\right]\left[\begin{array}{c}\mathbf{z}_1\\ \hdashline \mathbf{z}_2\end{array}\right] \tag{C.8a}$$

$$= \left[\begin{array}{c}(\mathbf{W}_{11}\mathbf{V}_{11}+\mathbf{W}_{12}\mathbf{V}_{12})\mathbf{z}_1+(\mathbf{W}_{11}\mathbf{V}_{21}+\mathbf{W}_{12}\mathbf{V}_{22})\mathbf{z}_2\\ \hdashline (\mathbf{W}_{21}\mathbf{V}_{11}+\mathbf{W}_{22}\mathbf{V}_{12})\mathbf{z}_1+(\mathbf{W}_{21}\mathbf{V}_{21}+\mathbf{W}_{22}\mathbf{V}_{22})\mathbf{z}_2\end{array}\right] \tag{C.8b}$$

Thus, in writing **WVz** in terms of block-martices, $\mathbf{W}_{ii}\mathbf{V}_{ii}+\mathbf{W}_{ij}\mathbf{V}_{ij}$ will only produce the diagonal matrices in (C.8b), while the effect of off-diagonal parts should also been taken into account. Therefore, as is shown in (C.8b), for this example where two matrices are multiplied, each local variable $\bar{\mathbf{z}}_i$ $(i = 1, 2)$ is expressed by the rows in **WV** corresponding to the i_{th} block times the vector $\mathbf{z}$. Similarly, for cases where N subsystems exist and/or when more than two matrices are multiplied, the end result of the multiplication should be decomposed.

C.3 Compact form of the plant model with pseudo-variables

The following local process model, in which interaction variables are replaced by the pseudo-variables, is used:

$$\begin{aligned}\bar{\mathbf{x}}_i(k+l+1|k) &= \mathbf{A}_{ii}\bar{\mathbf{x}}_i(k+l|k)+\mathbf{B}_{ii}\left[\sum_{a=0}^{l}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k)+\mathbf{u}_i(k-1)\right]+\mathbf{D}_{ii}\bar{\mathbf{d}}(k+l|k)\\ &\sum_{\substack{j=1\\ j\neq i}}^{N}\left[\alpha\mathbf{A}_{ij}\mathbf{x}_j(k+l|k)+\mathbf{B}_{ij}\mathbf{u}_j(k-1)+\mathbf{D}_{ij}\bar{\mathbf{d}}_j(k+l|k)\right]+\\ &\sum_{\substack{j=1\\ j\neq i}}^{N}\left[\beta\mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+l|k)+\mathbf{B}_{ij}\sum_{a=0}^{l}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+a|k)\right]\end{aligned} \tag{C.9}$$

Now, the local model (C.9) is written over the entire prediction and control horizons:

$$\mathbf{x}_i(k+1|k) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) +$$

$$\mathbf{D}_{ii}\bar{\mathbf{d}}_i(k|k) +$$

$$\sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{A}_{ij}\mathbf{x}_j(k) + \sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{D}_{ij}\bar{\mathbf{d}}_j(k|k) +$$

$$\sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k)$$

$$\mathbf{x}_i(k+2|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+1|k) + \mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+1|k) + \mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) +$$

$$\mathbf{D}_{ii}\bar{\mathbf{d}}_i(k+1|k) +$$

$$\sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{D}_{ij}\bar{\mathbf{d}}_j(k+1|k) +$$

$$\sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+1|k) + \sum_{\substack{j=1\\j\neq i}}^{N} [\mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+1|k) + \mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k)]$$

$$\vdots$$

$$\mathbf{x}_i(k+H_u|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+H_u-1|k) + \sum_{a=0}^{H_u-1} \mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) +$$

$$\mathbf{D}_{ii}\bar{\mathbf{d}}_i(k+H_u-1|k) +$$

$$\sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{D}_{ij}\bar{\mathbf{d}}_j(k+H_u-1|k) +$$

$$\sum_{\substack{j=1\\j\neq i}}^{N} \mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+H_u-1|k) + \sum_{\substack{j=1\\j\neq i}}^{N} \sum_{a=0}^{H_u-1} \mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+a|k)$$

$$\vdots$$

$$\mathbf{x}_i(k+H_p|k) = \mathbf{A}_{ii}\mathbf{x}_i(k+H_p-1|k) + \sum_{a=0}^{H_u-1} \mathbf{B}_{ii}\boldsymbol{\Delta}\mathbf{u}_i(k+a|k) + \mathbf{B}_{ii}\mathbf{u}_i(k-1) +$$
$$\mathbf{D}_{ii}\bar{\mathbf{d}}_i(k+H_p-1|k) +$$
$$\sum_{\substack{j=1\\ j\neq i}}^{N} \mathbf{B}_{ij}\mathbf{u}_j(k-1) + \sum_{\substack{j=1\\ j\neq i}}^{N} \mathbf{D}_{ij}\bar{\mathbf{d}}_j(k+H_p-1|k) +$$
$$\sum_{\substack{j=1\\ j\neq i}}^{N} \mathbf{A}_{ij}\tilde{\mathbf{x}}_j(k+H_p-1|k) + \sum_{\substack{j=1\\ j\neq i}}^{N} \sum_{a=0}^{H_u-1} \mathbf{B}_{ij}\boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+a|k) \tag{C.10}$$

Equations (C.10) are re-arranged and formed into the following compact version:

$$\underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\mathbf{A}_{ii} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & & & \ddots & & & & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & -\mathbf{A}_{ii} & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & & & \ddots & & & & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{ii} & \mathbf{I} \end{bmatrix}}_{\hat{\mathbb{A}}_{ii}} \underbrace{\begin{bmatrix} \mathbf{x}_i(k+1|k) \\ \mathbf{x}_i(k+2|k) \\ \vdots \\ \mathbf{x}_i(k+H_u|k) \\ \vdots \\ \mathbf{x}_i(k+H_p|k) \end{bmatrix}}_{\mathbf{x}_i} +$$

$$\underbrace{\begin{bmatrix} -\mathbf{B}_{ii} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \dots & & & \vdots \\ -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \dots & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} \\ \vdots & & \dots & & & \vdots \\ -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} & \dots & -\mathbf{B}_{ii} & -\mathbf{B}_{ii} \end{bmatrix}}_{\hat{\mathbb{B}}_{ii}} \underbrace{\begin{bmatrix} \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\ \boldsymbol{\Delta}\mathbf{u}_i(k+1|k) \\ \vdots \\ \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k) \end{bmatrix}}_{\boldsymbol{\Delta}\mathbf{U}_i} +$$

$$\underbrace{\begin{bmatrix} -\mathbf{D}_{ii} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{D}_{ii} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & & & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & -\mathbf{D}_{ii} \end{bmatrix}}_{\hat{\mathbb{D}}_{ii}} \underbrace{\begin{bmatrix} \mathbf{d}_i(k|k) \\ \mathbf{d}_i(k+1|k) \\ \vdots \\ \mathbf{d}_i(k+H_p-1|k) \end{bmatrix}}_{\mathfrak{D}_i}$$

$$\sum_{\substack{j=1\\ j\neq i}}^{N} \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ -\mathbf{A}_{ij} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{A}_{ij} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & & & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{ij} \end{bmatrix}}_{\hat{\mathbb{A}}_{ij}} \underbrace{\begin{bmatrix} \tilde{\mathbf{x}}_j(k+1|k) \\ \tilde{\mathbf{x}}_j(k+2|k) \\ \tilde{\mathbf{x}}_j(k+3|k) \\ \vdots \\ \tilde{\mathbf{x}}_j(k+H_p-1|k) \end{bmatrix}}_{\tilde{\mathbf{x}}_j} +$$

$$\sum_{\substack{j=1\\ j\neq i}}^{N} \underbrace{\begin{bmatrix} -\mathbf{B}_{ij} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \dots & & & \vdots \\ -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & \dots & -\mathbf{B}_{ij} & -\mathbf{B}_{ij} \\ \vdots & & \dots & & & \vdots \\ -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & -\mathbf{B}_{ij} & \dots & -\mathbf{B}_{ij} & -\mathbf{B}_{ij} \end{bmatrix}}_{\hat{\mathbb{B}}_{ij}} \underbrace{\begin{bmatrix} \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k) \\ \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+1|k) \\ \vdots \\ \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k) \end{bmatrix}}_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_j} +$$

$$\sum_{\substack{j=1\\ j\neq i}}^{N} \underbrace{\begin{bmatrix} -\mathbf{D}_{ij} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{D}_{ij} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & & & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & -\mathbf{D}_{ij} \end{bmatrix}}_{\hat{\mathbb{D}}_{ij}} \underbrace{\begin{bmatrix} \mathbf{d}_j(k|k) \\ \mathbf{d}_j(k+1|k) \\ \vdots \\ \mathbf{d}_j(k+H_p-1|k) \end{bmatrix}}_{\mathfrak{D}_j} =$$

$$\underbrace{\begin{bmatrix} \mathbf{A}_{ii} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{\bar{\mathbb{A}}_{ii}} \mathbf{x}_i(k) + \underbrace{\begin{bmatrix} \mathbf{B}_{ii} \\ \mathbf{B}_{ii} \\ \mathbf{B}_{ii} \\ \vdots \\ \mathbf{B}_{ii} \\ \mathbf{B}_{ii} \end{bmatrix}}_{\bar{\mathbb{B}}_{ii}} \mathbf{u}_i(k-1) + \sum_{\substack{j=1\\ j\neq i}}^{N} \underbrace{\begin{bmatrix} \mathbf{A}_{ij} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}}_{\bar{\mathbb{A}}_{ij}} \mathbf{x}_j(k) + \sum_{\substack{j=1\\ j\neq i}}^{N} \underbrace{\begin{bmatrix} \mathbf{B}_{ij} \\ \mathbf{B}_{ij} \\ \mathbf{B}_{ij} \\ \vdots \\ \mathbf{B}_{ij} \\ \mathbf{B}_{ij} \end{bmatrix}}_{\bar{\mathbb{B}}_{ij}} \mathbf{u}_j(k-1) \quad \text{(C.11)}$$

Appendix D

D.1 Deterministic CDMPC Simulation Results using GC(SALA) Method

In this appendix, simulation results using GC(SALA), IPC and MPMC methods are presented for both deterministic constrained and unconstrained CDMPC networks. Parameters used in the simulations are the same as those used in section 2.6. For the constrained CDMPC simulations, the constraints on the process inputs and outputs, shown in table 2.1, are used.

$$\begin{aligned} \mathbf{y}_1^{max} &= \begin{bmatrix} 5 & 1.5 \end{bmatrix}^T & \mathbf{y}_2^{max} &= \begin{bmatrix} 5 & 1.5 \end{bmatrix}^T \\ \mathbf{u}_1^{max} &= \begin{bmatrix} 5 & 4 \end{bmatrix}^T & \mathbf{u}_2^{max} &= \begin{bmatrix} 5 & 2 \end{bmatrix}^T \\ \mathbf{y}_1^{min} &= \begin{bmatrix} -15 & -1 \end{bmatrix}^T & \mathbf{y}_2^{min} &= \begin{bmatrix} -5 & -1 \end{bmatrix}^T \\ \mathbf{u}_1^{min} &= \begin{bmatrix} -5 & -4 \end{bmatrix}^T & \mathbf{u}_2^{min} &= \begin{bmatrix} -5 & -2 \end{bmatrix}^T \end{aligned} \tag{D.1}$$

Simulation Results for Unconstrained CDMPC: Process output and manipulated input profiles for the unconstrained case are plotted in Figures 2.1 and 2.2, for the GC(Newton-based), IPC, MPMC and centralized schemes. As expected, simulation results using GC(SALA) matches the results shown in these Figures.

The required number of communication cycles until convergence is plotted in D.1, for the four variations of coordination algorithms. The coordinator in GC(SALA) takes more iterations, in order to converge to the optimal centralized solution, because a first-order gradient-based algorithm solves the unconstrained dual optimization problem. This is while, the use of second order information (the Hessian) in the Newton-based coordinator in the GC method, leads to very fast convergence rate.

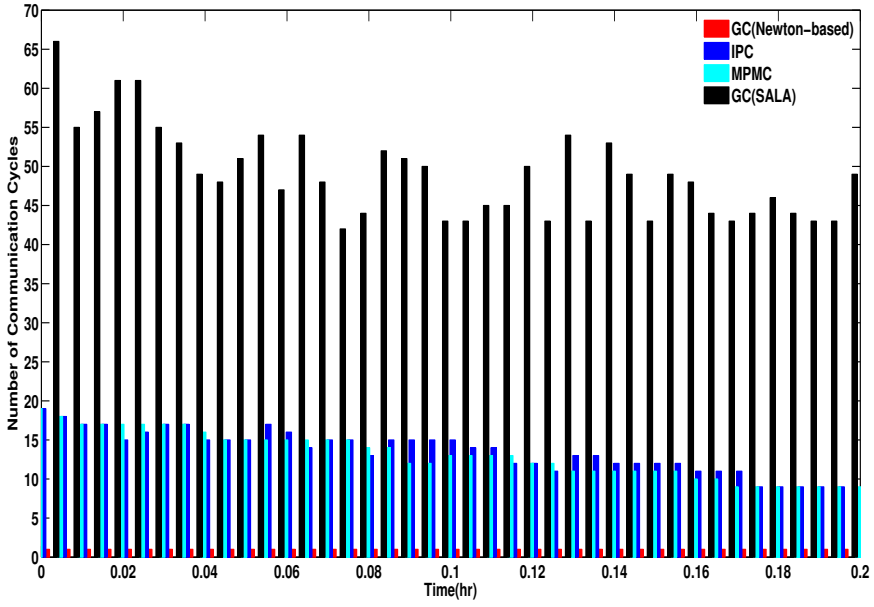


Figure D.1: Number of communication cycles at every sampling instant

On the other hand, the IPC and MPMC methods explicitly account for the effects of local inequalities, in the coordinator formulation. Therefore, in addition to showing similar convergence behaviour, number of iterations until convergence in these two methods, are lower than what is required by the GC(SALA); however, the Newton-based coordinator in the GC approach, still outperforms the coordinators in the IPC and MPMC methods, obtained using fixed-point iteration method.

Simulation Results for Constrained CDMPC: Before proceeding with the simulation results, it should be mentioned that neither steepest ascend, nor DFP and BFGS methods, with constant step-size ($0 < \epsilon \leq 1$), were able to solve the resulting dual optimization problem. In the steepest ascend method, in the first few communication cycles, second norm of error vector $\mathfrak{E}$ decreased. As the error vector got closer to zero, no further progress was observed. Quasi-Newton methods could not solve the convergence problem, either.

Process output and control input profiles in Figures D.2 and D.3, show that the three coordination methods successfully yield the optimal plant-wide MPC solution.

The number of communication cycles have been plotted in Figure D.4. As expected, the coordinator in the GC(SALA) approach, requires more computational effort to converge, compared with the coordinators of the IPC and MPMC methods.

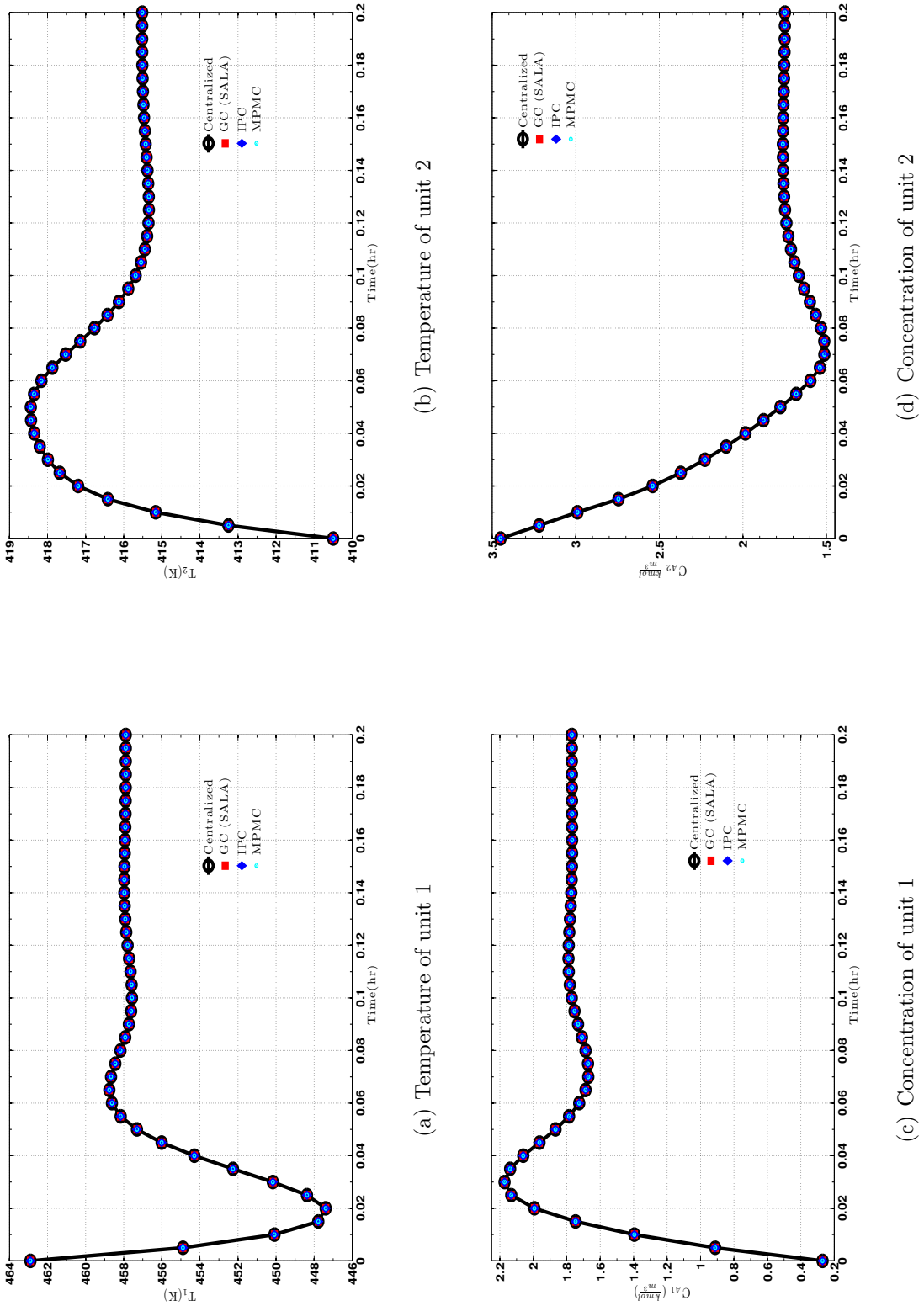


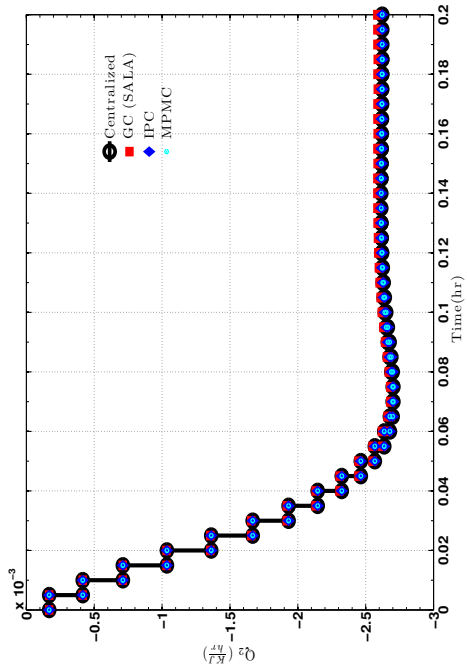
Figure D.2: Output trajectories resulting from CDMPC and centralized MPC

(a) Temperature of unit 1

(b) Temperature of unit 2

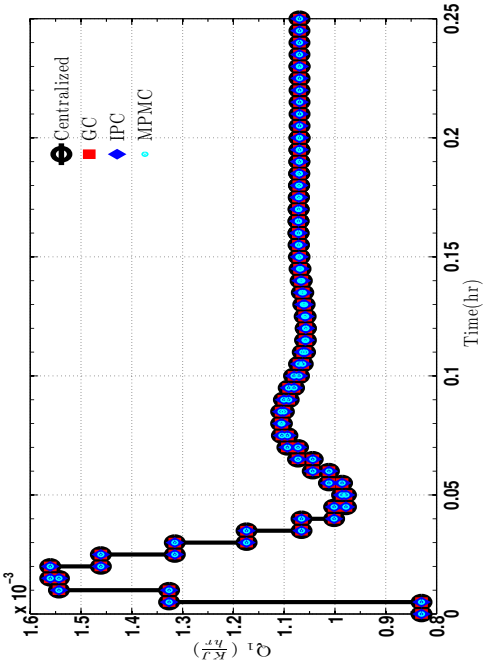
(c) Concentration of unit 1

(d) Concentration of unit 2



(a) Heat input rate for unit 1

(b) Heat input rate for unit 2



(c) Inlet reactant concentration for unit 1

(d) Inlet reactant concentration for unit 2

Figure D.3: Control input trajectories resulting from CDMPC and centralized MPC

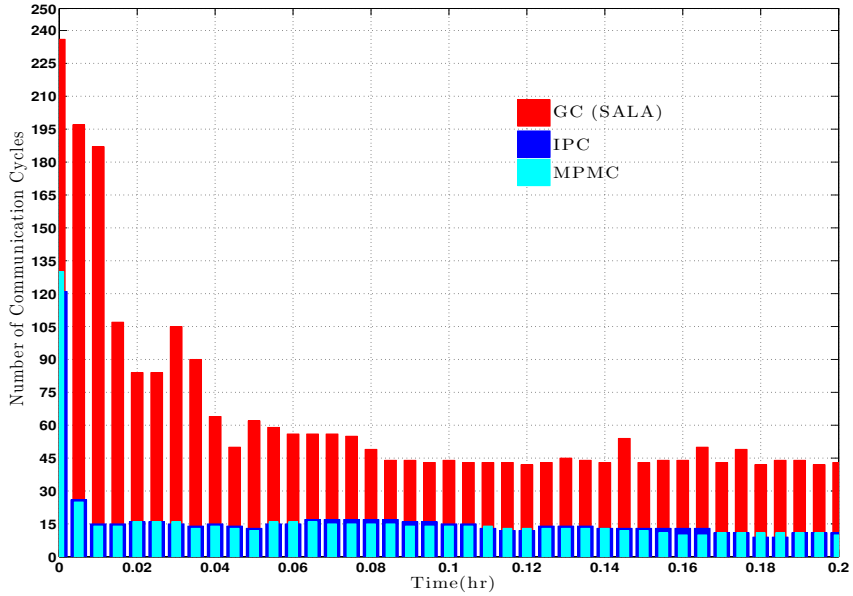


Figure D.4: Number of communication cycles at every sampling instant

Appendix E

E.1 Compact Form of Equality Constraints (4.22b)

The aggregate of CDMPC (4.21), along with the equality constraints (4.18) are equivalent to the plant-wide nonlinear MPC problem. To write the equality constraints of such a composite optimization problem in a compact form, first (4.18) and (4.21b) are combined as below:

$$
\begin{aligned}
&\mathbf{x}_i(k+l+1|k) = \\
&\mathbf{A}_{ii}^d \mathbf{x}_i(k+l|k) + \mathbf{B}_{ii}^d \sum_{a=0}^{l} \boldsymbol{\Delta}\mathbf{u}_i(k+a|k) \\
&+ \tilde{\mathbf{x}}_i(k+l|k) \\
&+ \Delta t \mathbf{f}_i \left(\tilde{\mathbf{x}}_i(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+a|k), \tilde{\mathbf{x}}_j(k+l|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+a|k)\right) \\
&- \mathbf{A}_{ii}^d \tilde{\mathbf{x}}_i(k+l|k) - \mathbf{B}_{ii}^d \sum_{a=0}^{l} \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+a|k)
\end{aligned}
\tag{E.1}
$$

Now, (E.1) is expanded over the prediction horizon H_p (i.e, $l = 0, \ldots, H_p - 1$) as

follows:

$$
\begin{aligned}
&\mathbf{x}_i(k+1|k) = \\
&\mathbf{A}_{ii}^d \mathbf{x}_i(k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
&+ \mathbf{x}_i(k)^1 \\
&+ \Delta t \mathbf{f}_i(\tilde{\mathbf{x}}_i(k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \tilde{\mathbf{x}}_j(k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k)) \\
&- \mathbf{A}_{ii}^d \mathbf{x}_i(k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k)
\end{aligned}
$$

$$
\begin{aligned}
&\mathbf{x}_i(k+2|k) = \\
&\mathbf{A}_{ii}^d \mathbf{x}_i(k+1|k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k+1|k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
&+ \tilde{\mathbf{x}}_i(k+1|k) \\
&+ \Delta t \mathbf{f}_i\left(\tilde{\mathbf{x}}_i(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+1|k), \tilde{\mathbf{x}}_j(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+1|k)\right) \\
&- \mathbf{A}_{ii}^d \tilde{\mathbf{x}}_i(k+1|k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+1|k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k)
\end{aligned}
$$

$$
\begin{aligned}
&\vdots \\
&\mathbf{x}_i(k+H_u|k) = \\
&\mathbf{A}_{ii}^d \mathbf{x}_i(k+H_u-1|k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k) + \cdots + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
&+ \tilde{\mathbf{x}}_i(k+H_u-1|k) \\
&+ \Delta t \mathbf{f}_i(\tilde{\mathbf{x}}_i(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)) \\
&- \mathbf{A}_{ii}^d \tilde{\mathbf{x}}_i(k+H_u-1|k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k) - \cdots - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k)
\end{aligned}
$$

$$
\begin{aligned}
&\vdots \\
&\mathbf{x}_i(k+H_p|k) = \\
&\mathbf{A}_{ii}^d \mathbf{x}_i(k+H_p-1|k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k) + \cdots + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
&+ \tilde{\mathbf{x}}_i(k+H_p-1|k) \\
&+ \Delta t \mathbf{f}_i(\tilde{\mathbf{x}}_i(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)) \\
&- \mathbf{A}_{ii}^d \tilde{\mathbf{x}}_i(k+H_p-1|k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k) - \cdots - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k)
\end{aligned}
\tag{E.2}
$$

Next, the equations in (E.2) are re-arranged and put together in the following

[1] States at time k are known $\implies \tilde{\mathbf{x}}_i(k|k) = \mathbf{x}_i(k)$.

matrix form:

$$
\underbrace{\begin{bmatrix}
\mathbf{I} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
-\mathbf{A}_{ii}^d & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \dots & \mathbf{0} & -\mathbf{A}_{ii}^d & \mathbf{I} & \mathbf{0} & \dots & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{ii}^d & \mathbf{I}
\end{bmatrix}}_{\hat{\mathbb{A}}_{ii}^d}
\underbrace{\begin{bmatrix}
\mathbf{x}_i(k+1|k) \\
\mathbf{x}_i(k+2|k) \\
\vdots \\
\mathbf{x}_i(k+H_u|k) \\
\vdots \\
\mathbf{x}_i(k+H_p|k)
\end{bmatrix}}_{\mathbf{x}_i} +
$$

$$
\underbrace{\begin{bmatrix}
-\mathbf{B}_{ii}^d & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
-\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
\vdots & & \dots & & & \vdots \\
-\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & \dots & -\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d \\
\vdots & & \dots & & & \vdots \\
-\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & \dots & -\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d
\end{bmatrix}}_{\hat{\mathbb{B}}_{ii}^d}
\underbrace{\begin{bmatrix}
\boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
\boldsymbol{\Delta}\mathbf{u}_i(k+1|k) \\
\vdots \\
\boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k)
\end{bmatrix}}_{\boldsymbol{\Delta}\mathbf{U}_i} +
$$

$$
\underbrace{\begin{bmatrix}
\mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
-\mathbf{I}+\mathbf{A}_{ii}^d & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & -\mathbf{I}+\mathbf{A}_{ii}^d & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
\vdots & & \ddots & & & \vdots \\
\mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & -\mathbf{I}+\mathbf{A}_{ii}^d
\end{bmatrix}}_{\tilde{\mathbb{A}}_{ii}^d}
\underbrace{\begin{bmatrix}
\tilde{\mathbf{x}}_i(k+1|k) \\
\tilde{\mathbf{x}}_i(k+2|k) \\
\tilde{\mathbf{x}}_i(k+3|k) \\
\vdots \\
\tilde{\mathbf{x}}_i(k+H_p-1|k)
\end{bmatrix}}_{\tilde{\mathbf{x}}_i} +
$$

$$
\underbrace{\begin{bmatrix}
\mathbf{B}_{ii}^d & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
\mathbf{B}_{ii}^d & \mathbf{B}_{ii}^d & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\
\vdots & & \dots & & & \vdots \\
\mathbf{B}_{ii}^d & \mathbf{B}_{ii}^d & \mathbf{B}_{ii}^d & \dots & \mathbf{B}_{ii}^d & \mathbf{B}_{ii}^d \\
\vdots & & \dots & & & \vdots \\
\mathbf{B}_{ii}^d & \mathbf{B}_{ii}^d & \mathbf{B}_{ii}^d & \dots & \mathbf{B}_{ii}^d & \mathbf{B}_{ii}^d
\end{bmatrix}}_{-\hat{\mathbb{B}}_{ii}^d}
\underbrace{\begin{bmatrix}
\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k) \\
\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+1|k) \\
\vdots \\
\boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k)
\end{bmatrix}}_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_i} -
$$

$$\Delta t \underbrace{\begin{bmatrix} \mathbf{f}_i\left(\tilde{\mathbf{x}}_i(k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \tilde{\mathbf{x}}_j(k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k)\right) \\ \mathbf{f}_i\left(\tilde{\mathbf{x}}_i(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+1|k), \tilde{\mathbf{x}}_j(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+1|k)\right) \\ \vdots \\ \mathbf{f}_i(\tilde{\mathbf{x}}_i(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)) \\ \vdots \\ \mathbf{f}_i(\tilde{\mathbf{x}}_i(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)) \end{bmatrix}}_{\mathbf{F}_i}$$

$$= \underbrace{\begin{bmatrix} \mathbf{x}_i(k) \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}}_{\mathcal{Z}_i} \tag{E.3}$$

E.2 Compact Form of Equality Constraints (4.23)

If (4.23) is used alongside the aggregate of CDMPC (4.21), the compact form of equality constraints in the composite optimization problem (4.22), will be obtained as follows:

$$\begin{aligned}
&\mathbf{x}_i(k+1|k) = \\
&\mathbf{A}_{ii}^d \mathbf{x}_i(k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
&+ \mathbf{f}_i(\tilde{\mathbf{x}}_i(k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \tilde{\mathbf{x}}_j(k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k)) \\
&- \mathbf{A}_{ii}^d \mathbf{x}_i(k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k)
\end{aligned}$$

$$\begin{aligned}
&\mathbf{x}_i(k+2|k) = \\
&\mathbf{A}_{ii}^d \mathbf{x}_i(k+1|k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k+1|k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
&+ \mathbf{f}_i\left(\tilde{\mathbf{x}}_i(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+1|k), \tilde{\mathbf{x}}_j(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+1|k)\right) \\
&- \mathbf{A}_{ii}^d \tilde{\mathbf{x}}_i(k+1|k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+1|k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k)
\end{aligned}$$

$$
\begin{aligned}
&\vdots \\
&\mathbf{x}_i(k+H_u|k) = \\
&\mathbf{A}_{ii}^d \mathbf{x}_i(k+H_u-1|k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k) + \cdots + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
&+ \mathbf{f}_i(\tilde{\mathbf{x}}_i(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)) \\
&- \mathbf{A}_{ii}^d \tilde{\mathbf{x}}_i(k+H_u-1|k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k) - \cdots - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k) \\
&\vdots \\
&\mathbf{x}_i(k+H_p|k) = \\
&\mathbf{A}_{ii}^d \mathbf{x}_i(k+H_p-1|k) + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k) + \cdots + \mathbf{B}_{ii}^d \boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
&+ \mathbf{f}_i(\tilde{\mathbf{x}}_i(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)) \\
&- \mathbf{A}_{ii}^d \tilde{\mathbf{x}}_i(k+H_p-1|k) - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k) - \cdots - \mathbf{B}_{ii}^d \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k)
\end{aligned}
\tag{E.4}
$$

After, re-arranging (E.4), the following matrix form, is obtained:

$$
\underbrace{\begin{bmatrix}
\mathbf{I} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
-\mathbf{A}_{ii}^d & \mathbf{I} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \ldots & \mathbf{0} & -\mathbf{A}_{ii}^d & \mathbf{I} & \mathbf{0} & \ldots & \mathbf{0} \\
\vdots & & & \ddots & & & & \vdots \\
\mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{A}_{ii}^d & \mathbf{I}
\end{bmatrix}}_{\hat{\mathbb{A}}_{ii}^d}
\underbrace{\begin{bmatrix}
\mathbf{x}_i(k+1|k) \\
\mathbf{x}_i(k+2|k) \\
\vdots \\
\mathbf{x}_i(k+H_u|k) \\
\vdots \\
\mathbf{x}_i(k+H_p|k)
\end{bmatrix}}_{\mathbf{x}_i} +
$$

$$
\underbrace{\begin{bmatrix}
-\mathbf{B}_{ii}^d & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0} \\
-\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0} \\
\vdots & & \ldots & & & \vdots \\
-\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & \ldots & -\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d \\
\vdots & & \ldots & & & \vdots \\
-\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d & \ldots & -\mathbf{B}_{ii}^d & -\mathbf{B}_{ii}^d
\end{bmatrix}}_{\hat{\mathbb{B}}_{ii}^d}
\underbrace{\begin{bmatrix}
\boldsymbol{\Delta}\mathbf{u}_i(k|k) \\
\boldsymbol{\Delta}\mathbf{u}_i(k+1|k) \\
\vdots \\
\boldsymbol{\Delta}\mathbf{u}_i(k+H_u-1|k)
\end{bmatrix}}_{\boldsymbol{\Delta}\mathbf{U}_i} +
$$

$$\underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{A}_{ii}^{d} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{ii}^{d} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \ddots & & & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{A}_{ii}^{d} \end{bmatrix}}_{\tilde{\mathbb{A}}_{ii}^{d}} \underbrace{\begin{bmatrix} \tilde{\mathbf{x}}_i(k+1|k) \\ \tilde{\mathbf{x}}_i(k+2|k) \\ \tilde{\mathbf{x}}_i(k+3|k) \\ \vdots \\ \tilde{\mathbf{x}}_i(k+H_p-1|k) \end{bmatrix}}_{\tilde{\mathbf{x}}_i} +$$

$$\underbrace{\begin{bmatrix} \mathbf{B}_{ii}^{d} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{B}_{ii}^{d} & \mathbf{B}_{ii}^{d} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & & \dots & & & \vdots \\ \mathbf{B}_{ii}^{d} & \mathbf{B}_{ii}^{d} & \mathbf{B}_{ii}^{d} & \dots & \mathbf{B}_{ii}^{d} & \mathbf{B}_{ii}^{d} \\ \vdots & & \dots & & & \vdots \\ \mathbf{B}_{ii}^{d} & \mathbf{B}_{ii}^{d} & \mathbf{B}_{ii}^{d} & \dots & \mathbf{B}_{ii}^{d} & \mathbf{B}_{ii}^{d} \end{bmatrix}}_{-\tilde{\mathbb{B}}_{ii}^{d}} \underbrace{\begin{bmatrix} \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k) \\ \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+1|k) \\ \vdots \\ \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k) \end{bmatrix}}_{\boldsymbol{\Delta}\tilde{\mathbf{U}}_i} -$$

$$\underbrace{\begin{bmatrix} \mathbf{f}_i\left(\tilde{\mathbf{x}}_i(k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \tilde{\mathbf{x}}_j(k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k)\right) \\ \mathbf{f}_i\left(\tilde{\mathbf{x}}_i(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+1|k), \tilde{\mathbf{x}}_j(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+1|k)\right) \\ \vdots \\ \mathbf{f}_i\left(\tilde{\mathbf{x}}_i(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \dots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \dots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)\right) \\ \vdots \\ \mathbf{f}_i\left(\tilde{\mathbf{x}}_i(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \dots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \dots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)\right) \end{bmatrix}}_{\mathbf{F}_i}$$

$$= \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}}_{\mathcal{Z}_i} \tag{E.5}$$

E.3 Derivative Calculations

At each communication cycle q, the coordinator uses updated values of the pseudo-variables to calculate the derivatives of nonlinear functions with respect to the pseudo-variables. In what follows, details on constructing the matrices $\mathbb{A}_{ji}^{c,q+1}$ and $\mathbb{B}_{ji}^{c,q+1}$ in (4.28) are provided.

The $\mathbf{F}_j$ vector, that contains the nonlinear functions over the prediction horizon, is written as:

$$\mathbf{F}_j =$$

$$\begin{bmatrix} \mathbf{f}_j(\tilde{\mathbf{x}}_i(k), \boldsymbol{\Delta}\mathbf{u}_i(k|k), \tilde{\mathbf{x}}_j(k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k)) \\ \mathbf{f}_j(\tilde{\mathbf{x}}_i(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+1|k), \mathbf{x}_j(k+1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+1|k)) \\ \vdots \\ \mathbf{f}_j(\tilde{\mathbf{x}}_i(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_u-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)) \\ \vdots \\ \mathbf{f}_j(\tilde{\mathbf{x}}_i(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_i(k+H_u-1|k), \tilde{\mathbf{x}}_j(k+H_p-1|k), \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k|k), \ldots, \boldsymbol{\Delta}\tilde{\mathbf{u}}_j(k+H_u-1|k)) \end{bmatrix}$$

$$\triangleq \begin{bmatrix} \mathbf{f}_j(k) \\ \mathbf{f}_j(k+1) \\ \vdots \\ \mathbf{f}_j(k+H_u-1) \\ \vdots \\ \mathbf{f}_j(k+H_p-1) \end{bmatrix} \tag{E.6}$$

where $j = 1, \ldots, N$. Thus, the gradient matrix $\nabla_{\tilde{\mathbf{x}}_i}\mathbf{F}_j\Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}}$ is constructed as:

$$\nabla_{\tilde{\mathbf{x}}_i}\mathbf{F}_j\Big|_{\substack{\tilde{\mathbf{X}}^{q+1} \\ \boldsymbol{\Delta}\tilde{\mathbf{U}}^{q+1}}} =$$

$$\begin{bmatrix}
\frac{\partial \mathbf{f}_j(k)}{\partial \tilde{\mathbf{x}}_i(k+1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \frac{\partial \mathbf{f}_j(k)}{\partial \tilde{\mathbf{x}}_i(k+2)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \frac{\partial \mathbf{f}_j(k)}{\partial \tilde{\mathbf{x}}_i(k+H_p-1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} \\
\frac{\partial \mathbf{f}_j(k+1)}{\partial \tilde{\mathbf{x}}_i(k+1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \frac{\partial \mathbf{f}_j(k+1)}{\partial \tilde{\mathbf{x}}_i(k+2)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \frac{\partial \mathbf{f}_j(k+1)}{\partial \tilde{\mathbf{x}}_i(k+H_p-1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} \\
\frac{\partial \mathbf{f}_j(k+2)}{\partial \tilde{\mathbf{x}}_i(k+1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \frac{\partial \mathbf{f}_j(k+2)}{\partial \tilde{\mathbf{x}}_i(k+2)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \frac{\partial \mathbf{f}_j(k+2)}{\partial \tilde{\mathbf{x}}_i(k+H_p-1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} \\
\vdots & \vdots & & \vdots \\
\frac{\partial \mathbf{f}_j(k+H_u-1)}{\partial \tilde{\mathbf{x}}_i(k+1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \frac{\partial \mathbf{f}_j(k+H_u-1)}{\partial \tilde{\mathbf{x}}_i(k+2)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \frac{\partial \mathbf{f}_j(k+H_u-1)}{\partial \tilde{\mathbf{x}}_i(k+H_p-1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} \\
\vdots & \vdots & & \vdots \\
\frac{\partial \mathbf{f}_j(k+H_p-1)}{\partial \tilde{\mathbf{x}}_i(k+1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \frac{\partial \mathbf{f}_j(k+H_p-1)}{\partial \tilde{\mathbf{x}}_i(k+2)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \frac{\partial \mathbf{f}_j(k+H_p-1)}{\partial \tilde{\mathbf{x}}_i(k+H_p-1)}\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}}
\end{bmatrix} =$$

$$\underbrace{\begin{bmatrix}
\mathbf{0} & \mathbf{0} & \cdots & & \mathbf{0} \\
\mathbf{A}_{ji}^{c,k+1} & \mathbf{0} & \cdots & & \mathbf{0} \\
\mathbf{0} & \mathbf{A}_{ji}^{c,k+2} & \cdots & & \mathbf{0} \\
\vdots & \vdots & & & \vdots \\
\mathbf{0} & \cdots & \mathbf{A}_{ji}^{c,k+Hu-1} & \cdots & \mathbf{0} \\
\vdots & \vdots & & & \vdots \\
\mathbf{0} & \mathbf{0} & \cdots & & \mathbf{A}_{ji}^{c,k+H_p-1}
\end{bmatrix}}_{\mathbb{A}_{ji}^{c,q+1}} \tag{E.7}$$

where the dimension of $\mathbb{A}_{ji}^{c,q+1}$ is $n_{xj}H_p \times n_{xi}H_p$. The superscript $k+l$ in $\mathbf{A}_{ji}^{c,k+l}$ indicates that the derivatives are calculated using the relevant updated pseudo-variable values at the $k+l$ time point. Similarly the gradient matrix $\nabla_{\Delta\tilde{\mathbf{U}}_i}\mathbf{F}_j\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}}$ is formed as:

$$\nabla_{\Delta\tilde{\mathbf{U}}_i}\mathbf{F}_j\Big|_{\substack{\tilde{\mathbf{X}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} =$$

$$
\begin{bmatrix}
\left.\frac{\partial \mathbf{f}_j(k)}{\partial \Delta\tilde{\mathbf{u}}_i(k)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \left.\frac{\partial \mathbf{f}_j(k)}{\partial \Delta\tilde{\mathbf{u}}_i(k+1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \left.\frac{\partial \mathbf{f}_j(k)}{\partial \Delta\tilde{\mathbf{u}}_i(k+H_u-1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} \\
\left.\frac{\partial \mathbf{f}_j(k+1)}{\partial \Delta\tilde{\mathbf{u}}_i(k)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \left.\frac{\partial \mathbf{f}_j(k+1)}{\partial \Delta\tilde{\mathbf{u}}_i(k+1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \left.\frac{\partial \mathbf{f}_j(k+1)}{\partial \Delta\tilde{\mathbf{u}}_i(k+H_u-1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} \\
\vdots & \vdots & & \vdots \\
\left.\frac{\partial \mathbf{f}_j(k+H_u-1)}{\partial \Delta\tilde{\mathbf{u}}_i(k)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \left.\frac{\partial \mathbf{f}_j(k+H_u-1)}{\partial \Delta\tilde{\mathbf{u}}_i(k+1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \left.\frac{\partial \mathbf{f}_j(k+H_u-1)}{\partial \Delta\tilde{\mathbf{u}}_i(k+H_u-1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} \\
\left.\frac{\partial \mathbf{f}_j(k+H_u)}{\partial \Delta\tilde{\mathbf{u}}_i(k)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \left.\frac{\partial \mathbf{f}_j(k+H_u)}{\partial \Delta\tilde{\mathbf{u}}_i(k+1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \left.\frac{\partial \mathbf{f}_j(k+H_u)}{\partial \Delta\tilde{\mathbf{u}}_i(k+H_u-1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} \\
\vdots & \vdots & & \vdots \\
\left.\frac{\partial \mathbf{f}_j(k+H_p-1)}{\partial \Delta\tilde{\mathbf{u}}_i(k)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \left.\frac{\partial \mathbf{f}_j(k+H_p-1)}{\partial \Delta\tilde{\mathbf{u}}_i(k+1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}} & \cdots & \left.\frac{\partial \mathbf{f}_j(k+H_p-1)}{\partial \Delta\tilde{\mathbf{u}}_i(k+H_u-1)}\right|_{\substack{\tilde{\mathbf{x}}^{q+1}\\ \Delta\tilde{\mathbf{U}}^{q+1}}}
\end{bmatrix} =
$$

$$
\underbrace{\begin{bmatrix}
\mathbf{B}_{ji}^{c,k} & \mathbf{0} & \cdots & \mathbf{0} \\
\mathbf{B}_{ji}^{c,k} & \mathbf{B}_{ji}^{c,k+1} & \cdots & \mathbf{0} \\
\vdots & \vdots & & \vdots \\
\mathbf{B}_{ji}^{c,k} & \mathbf{B}_{ji}^{c,k+1} & \cdots & \mathbf{B}_{ji}^{c,k+H_u-1} \\
\mathbf{B}_{ji}^{c,k} & \mathbf{B}_{ji}^{c,k+1} & \cdots & \mathbf{B}_{ji}^{c,k+H_u-1} \\
\vdots & \vdots & & \vdots \\
\mathbf{B}_{ji}^{c,k} & \mathbf{B}_{ji}^{c,k+1} & \cdots & \mathbf{B}_{ji}^{c,k+H_u-1}
\end{bmatrix}}_{\mathbb{B}_{ji}^{c,q+1}} \tag{E.8}
$$

where the dimension of $\mathbb{B}_{ji}^{c,q+1}$ is $n_{xj}H_p \times n_{ui}H_c$.